Cloud computing: survey on energy efficiency

Toni Mastelic, Ariel Oleksiak, Holger Claussen, Ivona Brandic, Jean-Marc Pierson, Athnasios V. Vasilakos

Cloud Computing: Survey on Energy Efficiency

TONI MASTELIC, Vienna University of Technology
ARIEL OLEKSIAK, Poznan Supercomputing and Networking Center and Poznan University of Technology
HOLGER CLAUSSEN, Bell Labs, Alcatel-Lucent
IVONA BRANDIC, Vienna University of Technology
JEAN-MARC PIERSON, University of Toulouse
ATHANASIOS V. VASILAKOS, University of Western Macedonia

Cloud computing is today's most emphasized Information and Communications Technology (ICT) paradigm that is directly or indirectly used by almost every online user. However, such great significance comes with the support of a great infrastructure that includes large data centers comprising thousands of server units and other supporting equipment. Their share in power consumption generates between 1.1% and 1.5% of the total electricity use worldwide and is projected to rise even more. Such alarming numbers demand rethinking the energy efficiency of such infrastructures. However, before making any changes to infrastructure, an analysis of the current status is required. In this article, we perform a comprehensive analysis of an infrastructure supporting the cloud computing paradigm with regards to energy efficiency. First, we define a systematic approach for analyzing the energy efficiency of most important data center domains, including server and network equipment, as well as cloud management systems and appliances consisting of a software utilized by end users. Second, we utilize this approach for analyzing available scientific and industrial literature on state-of-the-art practices in data centers and their equipment. Finally, we extract existing challenges and highlight future research directions.

1. INTRODUCTION

1.1. Motivation

New technological breakthroughs and massive production provide cheap and easy-to-use products that are more accessible to the average person, which leads to worldwide usage of emerging technologies. One of the main enablers of technological progress (and modern civilization more generally) is the energy that drives this machinery. However, due to its global usage, technological machinery creates an increasing demand for more energy. From 1990 until today, power consumption doubled from 10k TWh up to 20k TWh worldwide [Enerdata 2014]. Future projections estimate almost 40k TWh by 2040—a 2.2% increase per year [EIA 2013].

To enhance sustainability of the energy supply and to reduce emissions of greenhouse gases and other pollutants, the European Commission pointed out energy efficiency as the most cost effective way for achieving long-term energy and climate goals [EU 2011]. Among other solutions, Information and Communications Technology (ICT) has already been recognized as an important instrument for achieving these goals [EU 2008]. However, ICT is also recognized as one of the major energy consumers through equipment manufacture, use, and disposal [Advisory Group 2008], which also became one of the key issues of the Digital Agenda for Europe issued by the European Commission in 2010 [EU 2010].

Today, the majority of data centers spread over 300–4,500 square meters [Emerson 2008] and host up to several thousand server units. A typical 500-square-meter data center can consume 27,048 kilowatt-hours (kWh) per day [Emerson 2009], which is more than the consumption of more than 2,500 households in the EU [Enerdata 2011]. As reported by Koomey [2008], the power consumption of data centers doubled from 2000 to 2005 worldwide, going from 70.8 billion kWh to 152.5 billion kWh. Although the U.S. Environmental Protection Agency (EPA) estimated the same growth until 2010 [Fanara 2007], power consumption increased only by 56%, which corresponds to "between 1.1% and 1.5% of total electricity use worldwide" [Koomey 2011]. Although energy consumption did not double, this was mainly because of a lower installed server base due to the 2008 financial crisis and the use of virtualization instead of hardware efficiency improvements [Koomey 2011]. However, even at the start of the economic crisis in 2008, 63% of data center managers claimed that their operations and expansion plans would not be affected by the economic situation, and more than 80% of them had plans to renovate/expand existing facilities (47%) or build a new data center (38%) [Emerson 2008]. Moreover, only 13% anticipated that their capacity would be sufficient beyond 2014 [Emerson 2008].

The current situation is due to a greater focus on high availability than on energy efficiency [Emerson 2008]. However, in their 2010 Data Center Users' Group survey, Emerson [2010] identified heat density (cooling), energy efficiency, and power density among five major data center concerns. Taking into account that a global annual data

center construction size for 2020 is projected to be $78 billion, which is almost twice that in 2010, Belady [2011] stresses the importance of dealing with the energy efficiency and environmental impacts of ICT.

1.2. The Focus of the Survey

In this survey, we investigate the energy efficiency of an infrastructure that powers ICT machinery. As a representative of ICT technologies, we use cloud computing, the leading and most promising ICT approach and one that makes up a large portion of the total ICT energy consumption in providing elastic and on-demand ICT infrastructures [Koomey 2011]. A single cloud computing data center includes a data center building, power supply, and cooling as supporting equipment, as well as servers and networking ICT equipment. In this survey, we focus on energy efficiency of the ICT equipment separated into two domains: Server and Network. We also cover software solutions running on top of ICT equipment; these include the Cloud Management System (CMS) domain for managing a cloud infrastructure and the Appliance domain that represents a software for servicing users.

For the purpose of our survey, we define taxonomy and terminology used throughout the article describing energy efficiency in general. We apply it to the cloud computing infrastructure to create a systematic approach for analyzing the energy efficiency of ICT equipment within a data center.

1.3. The Goal of the Article

The main goals of this survey are to:

—Introduce a systematic analysis of cloud infrastructures by defining a taxonomy and terminology for energy efficiency.
—Provide an overview of existing technologies, research work, and projects for every domain of ICT equipment supporting the cloud computing concept.
—Discover and present correlations between different ICT domains with regard to energy efficiency.
—Highlight existing research areas and future challenges.

We describe our approach in Section 2, including goals for improving energy efficiency. Domains and their systems are described and analyzed in Sections 3, 4, 5, and 6 to provide a context for our energy efficiency goals, cover the state of the art and highlightresearch directions. Correlations between domains are given in Section 7. Finally, in Section 8, we conclude our survey.

2. TAXONOMY AND TERMINOLOGY

2.1. Cloud Computing

Cloud computing represents a novel and promising paradigm for managing and providing ICT resources to remote users. As the most cited definition of cloud computing, the U.S. National Institute of Standards and Technology (NIST) [Mell and Grance 2009] defines it as “a model that enables ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources, e.g., networks, servers, storage, applications, and services.” It utilizes technologies such as virtualization, distributed computing, Service Oriented Architecture (SOA), and Service Level Agreements (SLAs) [Foster et al. 2008], based on which different service types are offered. As defined by NIST [Mell and Grance 2009], cloud computing recognizes three service models: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). The service models are offered by providers, which can be public, private, or community providers, as well as hybrids. Regardless of its deployment or service model,

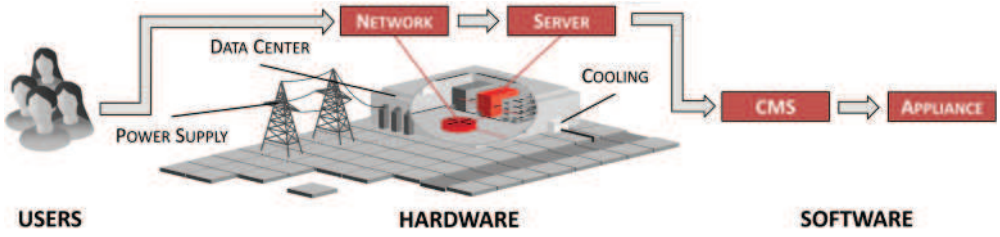

Fig. 1. Cloud computing data center domains.

cloud computing services are powered by large data centers comprised of numerous virtualized server instances and high-bandwidth networks, as well as of supporting systems such as cooling and power supplies. The listed equipment can be classified into two types, as shown in Figure 1; namely, hardware and software equipment [Hoelzle and Barroso 2013].

Hardware includes both ICT equipment and supporting equipment within a data center, as defined in Avelar et al. [2012]. ICT equipment includes *Network* and *Server* domains because they perform the main task of the data center and are the main focus of this survey. Domains such as Power supply, Cooling, and the Data center building itself are considered supporting equipment and are covered only briefly in this survey. Network and Server domains are described and analyzed in Sections 3 and 4, respectively.

Software equipment within a data center includes everything that runs on top of the ICT equipment. It includes two domains that are covered in this survey: *Cloud Management Systems (CMS)* that are used to manage the entire data center and *Appliances*, which include software used by a user. The CMS and Appliances are described and analyzed in Sections 5 and 6, respectively.

In this survey, the energy efficiency of both hardware and software equipment listed above is analyzed through a literature review of existing and emerging technologies and approaches. However, prior to our analysis, we first define the terminology used in the context of energy efficiency. Furthermore, because most of the domains overlap in some aspects and influence one another, we cover these correlations in Section 7. However, each domain is still analyzed separately in each section.

2.2. Energy Efficiency

Energy efficiency refers to a reduction of energy used for a given service or level of activity, as defined by the World Energy Council [Moisan and Bosseboeuf 2010]. However, defining the energy efficiency of data center equipment is extremely difficult [Fanara 2007] because it represents a complex system with a large number of components from various research areas such as computing, networking, management, and the like. The provided service of such a system is too diverse to be covered in detail.

On the one hand, surveys such as that found in Beloglazov et al. [2011] define an energy model through static and dynamic power consumption, which deals only with energy waste while running idle. On the other hand, Avelar et al. [2012] define a difference between energy used by ICT and auxiliary equipment in order to measure energy losses by the latter. However, we are interested in energy efficiency in general, and thus we combine these two in order to define energy efficiency from a more general perspective.

Figure 2 shows an arbitrary system as a set of interconnected components, in which each component can be observed as a different (sub)system. Therefore, every (sub)system can be optimized for itself, which can affect the energy efficiency of other related systems. Furthermore, each system requires an input energy for performing a

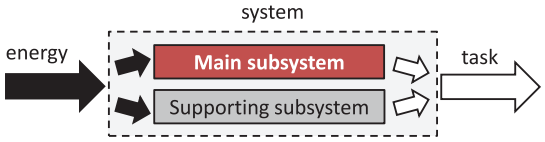

Fig. 2. A system and (sub)systems.

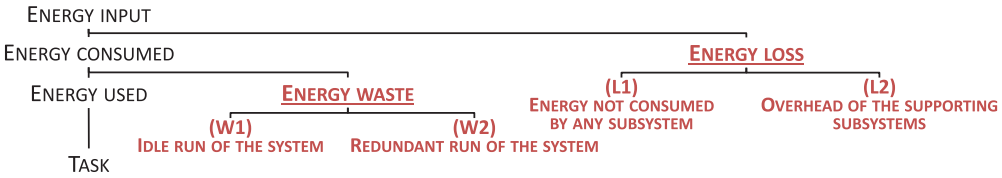

Fig. 3. Critical points within a system where energy is *lost* or *wasted*.

certain task, where a task is an abstract assignment that the system has to perform to fulfil its purpose. To improve the energy efficiency of a system, first it is necessary to identify problems that degrade efficiency.

Therefore, we identify two critical points where energy is not used in an efficient way but is instead *lost* or *wasted*. Both terms define inefficient use of energy from an agnostic point of view, where *energy loss* refers to energy brought to the system but not consumed for its main task (e.g., energy lost due to transport and conversion). This also includes energy used by supporting subsystems, such as cooling or lighting within a data center whose main task is the provision of cloud services. *Energy waste* refers to energy used by the system's main task but without useful output (e.g., energy used while running in idle mode). Additionally, useless work by the system is also considered energy waste; for example, for a cooling subsystem, this would mean keeping the cooling at maximum during the night when temperatures are lower. Both critical points are shown in Figure 3.

Based on these definitions, two goals are defined for reducing energy loss and two goals for reducing energy waste, thus improving the energy efficiency:

—L1. The first goal is minimizing a percentage of input **energy that is not consumed by a subsystem**. This can be done by implementing more efficient components (e.g., using more efficient power supply units for servers that leak less energy).
—L2. The second goal is to reduce the **overhead of supporting systems** (i.e., systems that do not perform the main task of the system), for example, by implementing a single cooling unit for the entire cabinet instead of cooling each rack server separately.
—W1. The third goal is to reduce **idle run of the system** and increase utilization or achieve zero energy consumption when no output is produced (i.e., during idle time). This also implies achieving a proportional increase of energy consumption with system output (e.g., to provide twice as much bandwidth, a network router requires twice the amount of energy or less).
—W2. The fourth goal is to minimize energy consumption where the **system performs redundant operations**. This can be done by implementing smart functions and subsystems, such as implementing an optimized algorithm that does not require redundant steps to perform the same task.

The listed goals are taken as a basis for the literature review in our search to find current as well as future research directions that focus on improving the energy efficiency of cloud computing infrastructure. Figure 4 shows data center domains and their energy cascades as they are covered in this article, starting with Network and

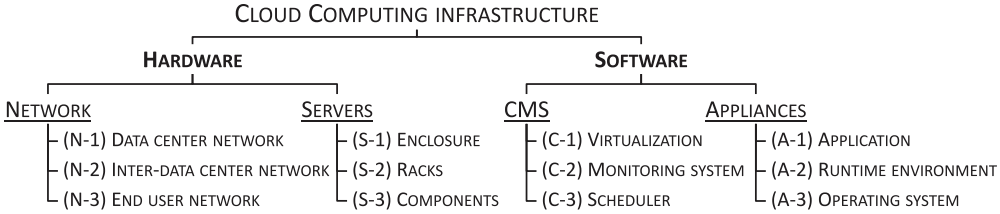

Fig. 4. Cloud computing infrastructure domains and related systems.

Table I. Research Areas in Wireless Networks and Relevant Literature

Research area	References
Energy efficiency in general	[GreenTouch 2012] [EARTH 2011]
Network architectures	[Claussen et al. 2009] [Razavi and Claussen 2012] [Yang and Marzetta 2013]
Scaling of energy consumption with load	[Grebennikov and Bulja 2012] [Claussen et al. 2010]
Low-complexity processing	[Mesleh et al. 2008] [Hochwald and Ten Brink 2003] [Claussen et al. 2005]

Table II. Research Areas in Wired Networks and Relevant Literature

Research area	References
Energy efficiency in general	[Gupta and Singh 2003] [Bolla et al. 2011] [Bianzino et al. 2012] [Ge et al. 2013] [Bari et al. 2013]
Scaling of energy consumption with load	[Gupta and Singh 2007b] [Nedevschi et al. 2008]
Traffic engineering and routing	[Zhang et al. 2010] [Vasic and Kostic 2010] [Vasic et al. 2011] [Cianfrani et al. 2012]

Server domains and moving on to the CMS and Appliance domains. Finally, relevant papers are summarized for each domain in Tables I, II, III, IV and V.

The following section covers the Network, the first hardware domain in this article.

3. NETWORK

The network is a key enabling component for cloud computing since it allows communication between computing and storage resources and allows the end user to access them. Recent traffic predictions for North America until 2020 indicate an exponential increase of network traffic within this period [Kilper et al. 2011].

3.1. Context

The energy consumption of the Network domain consists of three main systems: the connections inside of a data center, the fixed network between data centers, and the end user network that increasingly provides the wireless last hop to end users who access services via smartphones, tablets, and laptops. Based on this breakdown, each system brings its own energy wastes and losses, as shown in Figure 5.

—**DCN Data center network (N-1):** Within a data center, the energy consumption of the network currently accounts for up to 5% of its total energy consumption [Fanara 2007]. As shown by Abts et al. [2010], network power accounts for approximately 20% of the total power when the servers are utilized at 100%. However, it goes up to 50% when utilization of servers decreases to 15%. This is due to the fact that many new cloud applications heavily rely on fast connectivity within a data center. As a result, this leads to an increasing share in energy consumption for these networks.

Table III. Research Areas in Server Domain and Relevant Literature

Research area	References
Energy efficiency in general	[Hoelzle and Barroso 2013] [Greenberg et al. 2006]
Server cooling	[Snyder et al. 2006] [Park and Yang 2013] [Haywood et al. 2012] [Ayoub et al. 2012]
Processor architecture and design	[Zer et al. 2010] [Berge et al. 2012] [Vor dem Berge et al. 2014] [Dreslinski et al. 2009] [Mudge and Holzle 2010]
DVFS and alternatives	[Anghel et al. 2011] [Cioara et al. 2011a] [Chetsa et al. 2012] [Chen et al. 2012a] [Kim et al. 2012] [Kahng et al. 2013] [Megalingam et al. 2009] [Hankendi et al. 2013]
Cache management	[Powell et al. 2001] [Kim et al. 2013] [Tavarageri and Sadayappan 2013] [de Langen and Juurlink 2009] [Sundararajan et al. 2011] [Dreslinski et al. 2008]
Storage systems	[Chen et al. 2012b] [Felter et al. 2011] [Lee and Koh 2009] [Ge et al. 2011] [Chen and Zhang 2008] [Ruan et al. 2009b] [Manzanares et al. 2008] [Nijim et al. 2009] [Wang et al. 2008] [Bostoen et al. 2013] [Zhou and Mandagere 2012] [Scarfo 2013] [Shiroishi et al. 2009]

Table IV. Research Areas in CMS Domain and Relevant Literature

Research area	References
Energy efficiency in general	[Borgetto 2013], [Feller 2012] [Treutner 2012] [Beloglazov et al. 2011]
VM Reconfiguration and hardware management	[Zhang et al. 2005] [Borgetto et al. 2012b] [Cardosa et al. 2009] [Kim et al. 2011] [Nathuji and Schwan 2007] [Stoess et al. 2007]
VM placement	[Beloglazov and Buyya 2010] [Borgetto et al. 2012a] [Barbagallo et al. 2010] [Mazzucco et al. 2010] [Kamitsos et al. 2010] [Petrucci et al. 2010] [Borgetto et al. 2012a] [Hoyer et al. 2010] [Hoyer et al. 2010]
VM Migration and consolidation	[Liu et al. 2011a] [Banerjee et al. 2010] [Zhao and Huang 2009] [Nurmi et al. 2009] [Choi et al. 2008] [Cioara et al. 2011b] [Berral et al. 2010] [Hermenier et al. 2009] [Kumar et al. 2009] [Verma et al. 2008] [Feller et al. 2010]
VM scheduling	[Burge et al. 2007] [Steinder et al. 2007] [Beloglazov et al. 2012] [Berral et al. 2010] [Polverini et al. 2014]

Table V. Research Areas in Appliance Domain and Relevant Literature

Research area	References
Design and development	[Agrawal and Sabharwal 2012] [Saxe 2010] [Smith and Sommerville 2010] [Smith and Sommerville 2010]
Compilers	[Fakhar et al. 2012] [Falk and Lokuciejewski 2010] [Raghavan et al. 2008]
Application profiling and metrics	[Kansal et al. 2010] [Berge et al. 2012] [Witkowski et al. 2013] [Hemera 2013] [Magellan 2013] [Beik 2012]
Application platforms	[Smith and Sommerville 2010] [Pianese et al. 2010] [Smets-Solanes et al. 2011]

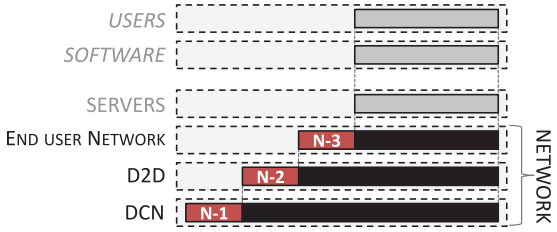

Fig. 5. Lc

Moreover, poor network architectures not suited for cloud applications can increase energy waste by unnecessarily rerouting traffic or keeping some parts of the network underutilized. Finally, not all energy is used for networking since communication equipment shows the highest heat load footprint [ASHRAE 2012], accounting for lost energy not used by the system. This also results in additional stress for the cooling system within a data center.

—**D2D Inter-data center network (N-2):** Connections between data centers are important for applications that run on a global scale, where instances that serve individual users are located in the data center closest to the end user but still need to communicate between each other. Another application of these networks is to migrate applications or data between data centers depending on the time of day, to minimize delay, energy consumption, or costs. As observed by Wang et al. [2014b], this communication includes background, noninteractive, and bulk data transfers.

—**End user network (N-3):** A connection to an end user who is accessing cloud services is usually made through a combination of wired and wireless networks. Since an increasing number of users access these services via mobile devices, the last hop of the connection is increasingly made through a wireless connection. Recent traffic predictions show that, compared to other kinds of network traffic, wireless traffic is increasing at the highest rate [Kilper et al. 2011], indicating that this trend will continue in the future. The wireless connection is significantly less energy efficient due to the high path loss, interference, and high processing involved in detection and error correction [Feeney and Nilsson 2001], which all represent energy consumption overhead created by supporting tasks rather than the main task (i.e., data delivery).

To reduce energy loss and waste, a number of actions can be taken to achieve the goals defined in the Section 2. These actions include:

—**L1.** Reducing the heat load of network equipment inside a data center (N-1) would reduce its energy consumption and the consumption of its cooling subsystem as well. This can be achieved by adapting the network equipment design suggested by ASHRAE [2012] implementing front to rear air flow. This would also increase its reliability by a factor of 2.

—**L2.** Goal L1 also brings benefits to goal L2: by reducing heat load, a smaller cooling subsystem can be installed, which consumes less energy. Although it comprises basic network equipment, failure handling supported by redundant equipment can also be considered a subsystem because it does not perform the main task of the system. Therefore, moving away from a traditional 2N tree topology toward the more flexible topologies currently being adopted by new data centers, such as Fat-Tree [Al-Fares et al. 2008], BCube [Guo et al. 2009], and DCell [Guo et al. 2008], can provide benefits in terms of improved energy-efficient traffic management.

—**W1.** Today's network equipment is not energy proportional, and simply turning on a switch can consume over 80% of its max power [Mahadevan et al. 2009]. By implementing power saving modes [Gupta and Singh 2007a; Claussen et al. 2010; Razavi and Claussen 2012], rate adaptation [Lopez-Perez et al. 2014; Gunaratne et al. 2008], or simply turning off unused ports, links, and switches inside a data center (N-1) [Heller et al. 2010] would reduce idle energy consumption and therefore achieve this goal. In addition to tweaking communication equipment, utilizing more energy-efficient network topologies can also reduce power consumption [Abts et al. 2010; Huang et al. 2011; Claussen et al. 2009]. For D2D networks (N-2), solutions such as NetStitcher [Laoutaris et al. 2011] can reduce network idle time by using unutilized network bandwidth for bulk transfers between data centers or exploiting the benefits of different data rates [Mahimkar et al. 2011].

—**W2.** Achieving this goal depends mostly on how a network is used by the servers, as well as the Software and User domains. However, some optimization can still be done by observing communication patterns and reducing unnecessary traffic. Such an approach combines network traffic engineering and VM assignment [Wang et al. 2014a], as well as application profiling for network traffic [Xie et al. 2012].

3.2. State of the Art

Energy efficiency of both wireless and wired access networks has been the focus of several international initiatives and projects. Some prominent ones are the Bell Labs-led GreenTouch™ consortium and the EU funded projects EARTH and ECONET.

GreenTouch™ [2012] is a consortium of more than 50 leading ICT organizations, from both industry and the academy, dedicated to reducing the carbon footprint of communication networks. The goal of GreenTouch is to identify key components that can increase network energy efficiency by a factor of 1,000 by year 2015 compared to 2010 levels. This goal will be achieved by delivering improved architectures, specifications, and technologies.

EARTH [2011] is a EU-funded IP research project in FP7 focused on mobile communication systems and their energy efficiency. The target of the project is a reduction of energy consumption by at least 50% focusing on LTE and LTE-A and existing 3G networks. The project ended in 2012 with tangible results in the areas of energy-efficient network architectures, deployment strategies, and optimization.

ECONET [2013] is a EU-funded IP research project in FP7 investigating "dynamic adaptive technologies for wired network devices that allow saving energy when a device, or part of it, is not used." The objective of the project is "reducing the energy requirements of wired network equipment by 50% in the short to mid-term and by 80% in the long run."

For wireless networks, recent research has focused on network architectures, scaling of energy consumption with load, and low-complexity processing.

3.2.1. Wireless Network Architectures. From a wireless architecture perspective, moving from traditional macrocellular networks to a HetNet architecture is one of the most impactful changes, one with a high potential for reducing energy consumption [Claussen et al. 2009]. In macrocellular networks, due to the fact that energy is transmitted in a relatively unfocused way and the distance between base station and mobile device is typically large, a high amount of transmit power is required. Serving users with small cells can reduce this path loss by several orders of magnitude, to an extent at which the transmit power is no longer the limiting factor. In Razavi and Claussen [2012], the authors showed that, for an urban area, energy consumption can be reduced by a factor of 46 by moving to a HetNet architecture with efficient idle modes. For macrocells, moving to remote radio heads can reduce cable losses and improve efficiency on the order of 3dB. Furthermore, increasing the number of antennas at the base station to large-scale antenna systems in combination with beamforming, which focuses energy to the user and also reduces interference, can significantly improve energy efficiency [Yang and Marzetta 2013].

3.2.2. Scaling of Energy Consumption with Load in Wireless Networks. A second important aspect for wireless network equipment is the ability of network equipment to scale energy consumption linearly with load and to switch off components into an idle state while not in use. Scalability with load is a big issue with macrocellular networks, which are dimensioned for peak load, but often operate at a fraction of their capacity. A major contributor to the power consumption of macrocells is their power amplifiers, which are currently relatively inefficient and consume a large amount of power even when the cell is only lightly loaded. One approach for addressing this problem is presented

in Grebennikov and Bulja [2012] using multistage Doherty power amplifiers. A further important area is the use of idle modes that allow network equipment to be switched off while not required and switched quickly back on when users need to be served. This is particularly important for heterogeneous networks where many small cells are deployed, since with reducing coverage, the fraction of time when the cell is not serving users is increasing. In Claussen et al. [2010], an efficient idle mode control mechanism was proposed that enables small cells to switch off all components, except for a low power uplink power detector, while not serving active connections.

3.2.3. Low-Complexity Processing in Wireless Networks. Finally, processing for wireless communications becomes more complex to maximize capacity within limited frequency resources. Examples for this trend are Multiple Input Multiple Output (MIMO) transmission, turbo coding, and base station coordination. In Mesleh et al. [2008], the authors showed that receiver complexity can be reduced significantly using the concept of spatial modulation. Examples of earlier work have focused on low-complexity detection algorithms [Hochwald and Ten Brink 2003] and on new ways of combining modulation and coding to reduce the complexity of the detection process [Claussen et al. 2005]. Reducing processing complexity is becoming increasingly important since when moving to small cells processing becomes the limiting factor for the energy consumption of wireless networks.

The topic of reducing energy consumption in fixed-access networks has been well-studied, and the concept of greening the Internet was proposed in Gupta and Singh [2003]. Since then, a significant amount of work has been carried out. Two comprehensive surveys have been recently published [Bolla et al. 2011; Bianzino et al. 2012]. Additional surveys include those by Ge et al. [2013] on power-saving techniques and more specifically Bari et al. [2013] on network virtualization. Most of the work can be categorized into two main directions: designing energy-aware network devices and exploring energy-efficient traffic engineering and routing.

3.2.4. Scaling of Energy Consumption with Load in Wired Networks. Initial attention has been focused on designing energy-aware network devices in which power consumption is manageable according to traffic load. Among them, sleeping and rate adaptation are two representative approaches. In Gupta and Singh [2007b], the authors proposed taking advantage of the low-power modes of Ethernet interfaces and discussed the detection of inactive periods to obtain energy savings with slight impact on network performance. Nedevschi et al. [2008] then presented two network power management schemes: adapting the rate based on offered workload during packet processing and sleeping during idle times.

3.2.5. Traffic Engineering and Routing in Wired Networks. Based on energy saving strategies proposed for single devices, network-wide energy conservation can be achieved by exploring energy-efficient traffic engineering and routing methods. Zhang et al. [2010] proposed an intradomain traffic engineering mechanism, GreenTE, which is able to guarantee given performance requirements while a maximum number of links are put into sleep mode. Vasic and Kostic [2010] argued that "a complete network energy saving solution requires a network-wide approach that works in conjunction with local measures" such as sleeping and rate adaptation. They then presented Energy-Aware Traffic engineering (EATe), achieving the same traffic rates while reducing network energy consumption by spreading load among multiple paths. Then, in Vasic et al. [2011], they propose a REsPoNse framework, in which a few energy-critical paths are identified and utilized, and traffic is shaped to enable the network to enter a low-power state. Compared with old methods, this framework can overcome the optimality-scalability tradeoff problem. Recently, Cianfrani et al. [2012] proposed power-aware

OSPF routing protocols that aim at providing routing services with the minimum number of links by modifying Dijkstra's algorithm and sharing the shortest path trees of underutilized routers. However, quality of service is recognized as a tradeoff.

3.3. Challenges and Research Directions

For wireless networks, the need for increased capacity is leading to a trend toward heterogeneous network architectures in which macrocells provide area coverage, but most of the capacity is provided by small cells. In addition to providing high capacity, such architectures have a high potential for reducing the energy consumption of these networks. Enabling efficient heterogeneous networks is an active area of research for both academia and industry. One important challenge is the cost-effective deployment of small cells. To achieve this, recent research has focused on providing wireless backhaul and energy harvesting, since backhaul and power are two significant cost factors. Improved idle mode control is another important area and is essential to enable energy efficiency with large numbers of small cells. In addition, better scaling of power with load and improving power amplifier efficiency is particularly relevant for macrocells and picocells. When moving toward smaller cells, processing becomes the limiting factor for energy consumption. Therefore, low-complexity detection and coding algorithms and low-power processors are also an important area for research to enable further energy reductions.

For wired networks, it is believed that a complete energy-saving solution requires both local and network-wide optimization strategies that work in conjunction with each other. From the local perspective, it is fundamental to design efficient network equipment with multiple energy-saving modes (e.g., sleeping and rate adaptation). Although this area of research has been largely explored, designing and producing network devices that can quickly adjust their modes and responses to dynamic network conditions is still challenging. A comprehensive design of energy-aware network devices should take into account not only the energy efficiency issue, but also the effects on perceived Quality of Service (QoS) and resilience. This needs further research efforts. From the global network perspective, most of the work is concentrated on proposing energy-efficient routing protocols. However, how to incorporate these protocols into real networks is still an open problem. Among the many issues, the scalability problem appears to be the most important one. Because real networks are usually of large scale, their designed protocol must be distributed and able to scale out easily. At the same time, it is still an open question how to trade off between energy saving and QoS, ensuring network stability while achieving energy conservation. Both remain important research topics that need further attention.

4. SERVERS

The Server domain includes computing and storage servers [Warkozek et al. 2012], as well as other components such as processors, memory, cabinets, and the like (but excluding communication equipment, which is part of the Network domain). It also considers aspects such as component layout within a rack and component architecture. As the second domain of IT equipment within a data center, its consumption contributes a large portion to the total consumption of a data center. A single rack of servers can consume more than 20kW [Fanara 2007], which is equal to the average power of 35 households in Austria during one year [Bittermann and Gollner 2011]. Considering that, in the United States alone, the total number of installed servers tripled from 2000 to 2010 (as estimated by the EPA [Fanara 2007]) to more than 15.8 million, improving server energy efficiency represents a top-priority tasks in the IT industry.

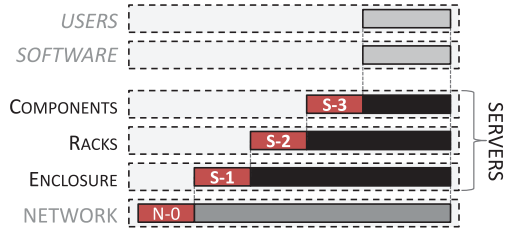

Fig. 6. Losses and wastes of server domain.

4.1. Context

In a perfect data center, the Server domain, along with the Network domain, would consist only of hardware equipment that consumes energy. Therefore, an obvious goal of every data center owner is to reduce the consumption of all supporting hardware equipment because it represents an energy loss. However, energy loss and waste do not stop there since servers can also contribute to energy waste due to poor server equipment usage policy, as well as energy loss due to a poor energy supply and internal subsystems. As shown in Figure 6, systems of the Server domain include the server enclosure, such as server cabinets. Server racks represent another system, and components within a rack, such as CPU, memory, hard-disk, and the like, are the third system.

—**Enclosure (S-1):** Enclosures may differ depending on the type of cooling applied to a data center. The most common air-based cooling, based on Computer Room Air Conditioners (CRACs), requires enclosures to have air inlets and outlets on opposite sides. The second type of cooling is indirect liquid cooling. Chilled water is delivered to the enclosure where it is used to absorb heat from the air that is used to cool servers. The enclosure can contain a closed loop of air or implement rear-door (or side-door) cooling, in which the cooled air is pushed back into the server room. Finally, direct liquid cooling solutions have been recently gaining interest [Haywood et al. 2012]. This type of cooling is particularly efficient for powerful and heavily loaded servers, as in High Performance Computing (HPC) applications; however, it may be also useful for cloud infrastructures. In enclosures with direct liquid cooling, warm water is used to cool server components directly, most commonly through the use of cold plates [Coolit 2013] or microchannels [IBM 2013]. Recently, other approaches based on the immersion of the whole server in a dielectric fluid have emerged (e.g., Iceotope system [Icetope 2013]). Liquid cooling approaches provide significant energy savings (up to around 40% compared to air-based cooling), but have an impact on hardware cost, complexity, and compatibility with other equipment.

—**Racks (S-2):** The idle power consumption of a server can be more than 50% of its peak power consumption [Takouna et al. 2011]. Moreover, “most servers consume between 70 and 85 percent of full operational power” [Emerson 2009], which certainly does not represent a proportional increase of energy consumption with respect to system output. Consequently, “a facility operating at just 20 percent capacity may consume 80 percent of the energy as the same facility operating at 100 percent capacity” [Emerson 2009]. Additionally, this includes a huge energy waste by running servers idle without any useful output or with low utilization in the 10–50% utilization range, which is usually the case in typical data centers [Hoelzle and Barroso 2013]. Finally, racks containing components that are not used at all (e.g., graphics cards) also contribute to energy loss. Another source of energy loss is fans, which have typical efficiency of around 60% (i.e., around 40% of power is lost due to heat dissipation). Additionally, if fan speed is not well adjusted to server load and temperature, a significant amount of energy is wasted.

—**Components (S-3):** The energy efficiency of server components drastically affects the overall efficiency of a server. Most focus on components that take a bigger slice of the total energy consumption, such as the CPU, which can consume more than a third of total server energy consumption [Fan et al. 2007]. The typical Thermal Design Power (TDP) of today's processors can fall in the range of 80W–103W or 91 W on average [Emerson 2009]. However, this power is not proportional to its output. As a rule of thumb, CPU power increases by approximately k^2 when CPU frequency increases by k [Mudge and Holzle 2010]. Practical experiment results are given in Takouna et al. [2011], where a VM utilized 100% of a single physical core while consuming 26W. On the other hand, when a VM with the same performance ran on two physical cores, each being 50% utilized, it consumed only 17W. However, servers with large number of slower CPU cores can lead to lower utilization (i.e., a bin-packing problem in which smaller bins cause a bigger bin-packing problem [Mudge and Holzle 2010]).

In addition to underutilized CPU cores that affect dynamic power consumption, caches can also be poorly used or underutilized, which adds to the static power consumption of a processor. Since cache takes more than 40% of the processor die area [Apparao et al. 2008], it can significantly increase static power consumption, which in modern processors accounts for 20–40% of total power consumption [Kaxiras and Martonosi 2008]. Memory also creates energy overheads since it is built to provide high performance to meet increasing CPU demands and thus has grown in density, functionality, and scalability [Tolentino et al. 2009]. This has resulted in neglecting the energy efficiency of the memory subsystem. Finally, the disk system has proved to be another power drain that can generate an energy cost of as much as 25% annually while also occupying up to 75% of the floor space in a data center [Wang et al. 2008].

To mitigate all this energy loss and waste, a number of actions can be performed. Following our approach from Section 2, these actions include:

—**L1.** Reducing the heat load of server components such as the CPU fulfils this goal. This can be achieved by using more energy-efficient components and their architectures; for example, using slower, so-called wimpy CPU cores that are more power efficient [Mudge and Holzle 2010], as in the FAWN project [Andersen et al. 2009] where they utilize wimpy cores to build an energy-efficient key-value storage system. Another recognized approach is limiting input energy to a specific component (S-1) or an entire rack (S-2), also referred to as power capping [Bhattacharya et al. 2012]. Similarly, in the case of the memory subsystem, performance can be adjusted (i.e., throughput is used to mitigate high temperatures) and thus avoid energy loss through heat [Lin et al. 2007]. Energy loss can also be reduced by using compact server configurations that exclude components that are not used (e.g., Google uses such an approach in building its data centers).
—**L2.** Following goal L1, goal L2 provides additional energy savings by reducing energy consumed by supporting systems, such as cooling and power supplies inside the server enclosure (S-3) and the servers themselves (S-2). For example, Google places backup batteries next to racks, thereby avoiding large UPS units that require their own cooling systems [Wired 2013]. With this approach, goal L1 is also achieved since large UPS units leak electricity due to their low efficiency [Greenberg et al. 2006]. In addition to cooling and power supply systems, during idle run, subsystems such as cache can be turned off on most modern processors that employ more sophisticated hardware [Dharwar et al. 2012].
—**W1.** Using components that can automatically scale their power consumption based on current load would move toward achieving this goal; for example, using dynamic voltage and frequency scaling (DVFS)-capable CPUs that provide different P-states

(power modes while being utilized), as well as sleep C-states (power modes while idle). Dharwar et al. [2012] provide an overview of these techniques along with power capping. The same applies for other components, such as memory and storage disks, which can be put into a low power state while idle. However, this is beneficial only when there are frequent idle periods; otherwise, this can create even bigger power consumption overheads due to spin-up in the case of storage disks [Wang et al. 2008]. Using DRPM [Gurumurthi et al. 2003] with dynamic spin speed may represent a more flexible solution for gradually scaling the performance of a disk. Other energy-saving techniques include MAID [Colarelli and Grunwald 2002], BUD [Ruan et al. 2009a], EERAID [Li and Wang 2004], and PDC [Pinheiro and Bianchini 2004].

Choosing the right processor architecture can also contribute to more efficient energy usage. Due to the nature of applications running in a cloud (e.g., web search, video hosting, and MapReduce), emphasis is placed on parallelism and thus on multicore processors instead of high-speed single-core processors [Mudge and Holzle 2010]. However, using single-threaded operations still beats multithreaded operations on slower CPUs due to the higher software development and optimization costs [Mudge and Holzle 2010]. Therefore, optimization should also be done in the Appliance domain to develop middleware for transparent workload parallelization.

—**W2.** As shown by Tavarageri and Sadayappan [2013], bigger cache size does not necessarily mean a lower miss rate. Therefore, choosing the right size cache can decrease energy waste and achieve this goal. Additionally, using cache subsystems for storage disks to reduce reads and writes from/to the disk and increase its idle time can also contribute to energy savings [Wang et al. 2008]. Such onboard controller caches can already be found on modern hardware.

4.2. State of the Art

Server enclosures such as cabinets are important for optimal cooling and power supply systems, where later are out of scope of this survey. Although there is some research work in this field, most innovations come from the industry and production environments as best practices. Some important literature includes the book by Hoelzle and Barroso [2013] covering Google's practices inside data centers, whereas Greenberg et al. [2006] provide best practices learned from benchmarking 22 data centers.

4.2.1. Server Cooling. Choosing an optimal enclosure design affects the efficiency of the power supply and cooling systems, and, via these systems, the server racks as well. As shown in Snyder et al. [2006], localized cooling, specifically Embedded Thermoelectric Cooling (eTEC), can reduce the temperature of localized hot spots generated by modern processors and therefore reduce power consumption. Park and Yang [2013] compare eTEC with a vapor compression refrigeration system for cooling microprocessors. They show how eTEC can achieve from 3% up to 10% in power savings and vapor up to 25%. The general conclusion is that localized cooling of the right component can produce some worthwhile improvements. Additionally, Haywood et al. [2012] suggest using the heat generated by CPUs to drive a cooling process, specifically, a single-effect lithium bromide (Li-Br) refrigeration system. Ayoub et al. [2012] provide an overview of thermal management solutions for memory subsystems. They also present JETC, a management system for server memory and CPUs with a combined energy-efficient thermal and cooling solution. By applying such an approach, they consider dependencies between CPU and memory, as well as their shared cooling subsystem and achieve a 50.7% average energy reduction.

4.2.2. Processor Architecture and Design. Combining different types of server components has proved promising when it comes to applying energy saving schemes. As part of the EuroCloud project, Zer et al. [2010] propose a new architecture for low-power servers

based on ARM processor technology. Within the CoolEmAll project [Berge et al. 2012], the prototype of the RECS system is developed and evaluated. The system, developed by the Christmann company, may include up to 18 heterogeneous computing nodes or even 72 nodes based on ARM CPUs within a single rack unit. This high density of nodes combined with fine-grained monitoring and control allows for reduced space, resources, and power consumption. Generally, solutions based on a high number of densely packed low-power processors, so-called microservers, are one of the trends visible recently on the market. In addition to physical prototype development, CoolEmAll also proposes blueprints that define efficient hardware for data centers. Furthermore, the project also defined a specification of the so-called Data Center Efficiency Building Blocks [Vor dem Berge et al. 2014] to be used to model and simulate energy efficiency of data center components, including servers, racks, and enclosures. Dreslinski et al. [2009] propose a cluster architecture with multicore processors. The idea is to use the same processors for single- and multi-threaded operations, where a processor with four cores can be reconfigured to run only one overclocked core using power from those three that are turned off. Finally, Mudge and Holzle [2010] give an overview of challenges for choosing and building energy-efficient processors for cloud infrastructures.

4.2.3. DVFS and Alternatives. One of the most notable techniques for reducing energy consumption in a CPU is reducing its power input due to its disproportional energy consumption; this is referred to as DVFS. A large body of research work is currently trying to utilize DVFS in order to reduce energy consumption, which includes algorithms such as those presented in Anghel et al. [2011] and Cioara et al. [2011a], or combining the DVFS with other components, such as memory and disk Chetsa et al. [2012]. Other works that utilize DVFS include Chen et al. [2012a] and Kim et al. [2012], whereas Kahng et al. [2013] propose some improvements over DVFS itself. Going even deeper into power limitation, Megalingam et al. [2009] propose a novel clocking scheme on a pipelined RISC CPU that is able to reduce power consumption by 50%. Unlike DVFS approaches or ones using clock gating, vCAP [Hankendi et al. 2013] uses co-scheduling for resource allocation in order to maximize the performance under power and performance constraints. It identifies nonscalable VMs in terms of performance and consolidates them. This is an example of how the CMS domain (i.e., VM management) in combination with the Server domain can provide more benefits for energy- saving techniques.

4.2.4. Cache Management. Turning off parts of a cache to reduce static power consumption (also known as leakage power) is proposed by Powell et al. [2001] and Kim et al. [2013]. Instead of simply turning off parts of the cache that are not used, Tavarageri and Sadayappan [2013] propose a compile-time analysis to determine useful cache size for a given system configuration. Additionally, in order to avoid memory write-backs to those cache parts that are being turned off, de Langen and Juurlink [2009] propose a cache organization called the clean/dirty cache (CD cache) that combines the properties of write-back and write-through. A smart cache is presented in Sundararajan et al. [2011] that allows reconfiguration of both size and associativity, thus dynamically changing the hierarchy as a program runs. In addition to improving management over existing cache subsystems, using novel cache architectures and technologies can cut energy loss at the start. Dreslinski et al. [2008] suggest using near-threshold cache architectures to reduce energy consumption. Additionally, they combine this with traditional cache methods to maintain performance.

4.2.5. Storage Systems. In addition to optimizing the cache subsystem itself, the cache can be used to achieve energy efficiency goals for other subsystems such as storage disks. Chen et al. [2012b] exploit the catching scheme to improve the energy efficiency

of RAID disk systems. Along with energy efficiency, Felter et al. [2011] also consider disk reliability. And, in addition to disk reliability, lifetime and performance of a cache memory that is implemented using SSD is considered in Lee and Koh [2009]. Instead of using aggressive prefetching, Ge et al. [2011] present DiscPOP, a power-aware buffer management method that populates cache by exploiting the relationship between I/O access and application pattern behavior, which includes information from the CMS and Appliance domain. Another example of smart prefetching is presented in Chen and Zhang [2008], where the authors extend data disk idle mode by populating cache memory with bursty pattern disk access. A similar approach is researched in Ruan et al. [2009b], where the authors suggest redirecting I/O requests to disk buffers instead of to data disks. Using a prefetching scheme also applies in the disk buffers approach shown in Manzanares et al. [2008]. A further step in using buffers is suggested by Nijim et al. [2009] by combining a buffer disk approach with a cache approach. They use a flash memory cache on top of disk buffers to store most popular data, thus providing fast access to this data without affecting the disks. Wang et al. [2008] also combine memory-level (cache) and disk-level (RAID) redundancy in order to save energy. These papers, along with Bostoen et al. [2013] and Zhou and Mandagere [2012], provide a good overview of relevant work done in the field of storage disk energy efficiency. Moreover, SSD disks and their utilization in energy-efficient storage systems are discussed in Scarfo [2013], whereas HDD technology is discussed in Shiroishi et al. [2009].

4.3. Challenges and Research Directions

Utilizing low power modes for server components has proved beneficial only for long idle modes, which are not that common in a production environment [Hoelzle and Barroso 2013]. Although servers do not show high utilization rates [Hoelzle and Barroso 2013], they still require promptness due to elasticity requirements, and they are usually performing some light tasks. Therefore, the goal is to achieve self-scalability of server components, on both the hardware and software levels. This includes an energy consumption increase/decrease proportional to provided performance. This can be achieved by utilizing techniques such as DVFS, which has become a common feature of modern processors. Another goal is to proportionally scale available resources with power consumption (i.e., consolidating underutilized components and achieving zero power consumption for idle ones). This can also be achieved by using low-power components when demand is low, in combination with traditional components for high-performance requirements.

5. CLOUD MANAGEMENT SYSTEM (CMS)

Managing and monitoring a cloud infrastructure with regard to energy efficiency and consumption has been identified as the main concern within a data center facility according to Emerson [2010]. Thus, the CMS plays an important role in trying to improve efficiency, increase utilization, and thus lower energy loss/waste within a data center.

5.1. Context

The CMS domain includes the scheduler, monitoring system, virtualization technology, and all other software components responsible for managing physical and virtual machines within a cloud (e.g., OpenStack [OpenStack 2012] and Xen hypervisor [Citrix 2012]). A scheduler's main function is to deploy resources for fulfilling customer requests. Its supporting task is providing a monitoring system that gives additional information about allocated and available resources, such as utilization, QoS, and the like. Additionally, virtualization technology is used for better resource management and on-demand deployment and offers high scalability for cloud infrastructures.

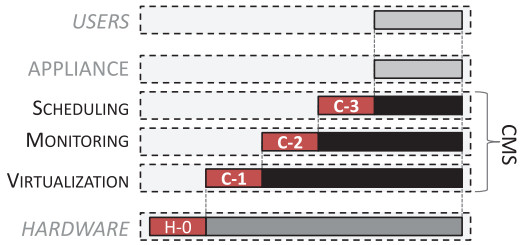

Fig. 7. Losses and wastes of CMS domain.

Based on this, the energy efficiency of the CMS can be examined through its component systems and includes both energy loss and waste, as shown in Figure 7.

—**Virtualization (C-1):** Virtualization technology provides an additional infrastructure layer on top of which multiple VMs can be deployed [Uhlig et al. 2005]. Although virtualization technology can improve resource utilization [Mastelic and Brandic 2013], it also consumes resources and thus creates an energy consumption overhead, mostly through a hypervisor [Jin et al. 2012]. As reported in Jin et al. [2012], a hypervisor based on full virtualization (i.e., KVM [RedHat 2012]) creates much higher overhead (11.6%) than one based on paravirtualization (0.47%; e.g., Xen [Citrix 2012]) and as opposed to using physical machines. Additionally, overly large VM images are sources of additional losses (e.g., too large memory allocation and storage size).

—**Monitoring system (C-2):** The monitoring system provides information used for managing an infrastructure and providing QoS [Emeakaroha et al. 2012]. However, gathering monitoring metrics consumes resources (e.g., monitoring agents and probes) and thus creates an energy consumption that is considered a loss according to the model represented in Section 2. This can be due to cumbersome monitoring systems whose monitoring agents are heavyweight processes that consume lots of memory and CPU power even when idle. Aceto et al. [2013] give an overview of commercial and open-source monitoring tools, as well as monitoring systems in general. Database storage for metric values is another example of memory cluttered with a huge amount of unused data.

—**Scheduler (C-3):** Managing cloud resources should not be overdone; for example, re-scheduling VMs every couple of minutes might give optimal deployment at the moment, but the rescheduling itself would probably consume more energy than it saves. Furthermore, migrations can lead to performance overhead as well as energy overhead. Although a performance loss can be avoided by using live migrations of VMs [Liu et al. 2009], the resulting energy overhead is often overlooked. When migrating a VM from one node to another, both nodes must be powered on until the migration is complete [Petrucci et al. 2010]. This includes both time and energy overheads for the migration, which is only rarely considered in the literature in the context of job placement [Hermenier et al. 2009].

In addition to these issues, H-0 in Figure 7 represents energy delivered to hardware equipment that was not fully utilized by the CMS domain (e.g., idle machines). Although H-0 can be directly related to the Hardware domain, it can also be minimized from the CMS perspective, for example, by consolidating underutilized machines and turning off idle ones [Feller et al. 2012].

To reduce these wastes and loses, a number of actions can be taken, according to the goals defined in Section 2:

—**L1**. Goal L1 can be achieved during the development phase of the CMS by implementing functions that can directly control hardware equipment because the CMS has "knowledge" of which resources are required and which are not (e.g., shutting down idle machines [Borgetto et al. 2012b]). The CMS can go beyond controlling only servers to expand its control to the network system or even to the cooling and power supply systems [Lago et al. 2011]. In this way, energy delivered to hardware equipment that is not utilized by the CMS (H-0) could be significantly reduced.
—**L2**. To meet goal L2, the CMS should use lightweight supporting subsystems, such as monitoring (C-2) and virtualization (C-1) technologies, and avoid cumbersome systems that provide large numbers of functionalities that are not utilized by the cloud manager. This includes lightweight monitoring systems [Ma et al. 2012] and the selection of appropriate virtualization technology, namely, full-virtualization vs. para-virtualization or even microkernel architectures [Armand and Gien 2009].
—**W1**. Running the CMS supporting systems idle still consumes resources and therefore wastes energy (C-1 and C-2). For this reason, CMS subsystems should be implemented in a modular fashion, in which modules are loaded only when they are actually required (e.g., the monitoring agent that loads plugins for initialized metrics and removes them once they are no longer required [Mastelic et al. 2012]). This also includes minimizing resource consumption while running in idle mode (e.g., using lightweight hypervisors).
—**W2**. CMS system energy waste (C-3) can be avoided by optimizing the scheduler and measuring not only its results, but also its tradeoffs for achieving those results (e.g., how much resources a single scheduling action takes and how many actions are taken). This includes optimization of the scheduling algorithm and technology used for its implementation.

5.2. State of the Art

Several research papers focus on different levels of potential actions at the CMS level to mitigate energy savings. We can distinguish four levels of actions. First, a VM can be reconfigured to change its resource requirements; in this way, the stress on the system is lower and energy consumption reduced. Furthermore, the physical machines themselves can be adjusted to their actual load so as to reduce their power consumption. Second, the placement of VMs can be optimized such that the most efficient physical machines are used. Third, VMs can be moved between physical machines, consolidating the load on fewer hosts and powering off unused machines. Finally, the scheduling of VMs over time can be adapted to reduce resource consumption at any given time. All these actions can be combined, and the use of several levers in the same framework is described in the following literature. Also, it must be taken into account the potential degradation of QoS induced. The approaches differ in the kind of constraints put on the QoS. PhD dissertations [Borgetto 2013; Feller 2012; Treutner 2012] or surveys [Beloglazov et al. 2011] are the primary sources of literature reviews.

5.2.1. VM Reconfiguration. Considering the first possibility, VM reconfiguration and hardware adjustment, Zhang et al. [2005] propose VMs that self-adapt their resource allocation to their demands. Similarly, Borgetto et al. [2012b] propose VM reconfiguration in which the middleware adapts the VM resources' demands to their needs. The authors propose proactive VM reconfiguration models, taking also into account the time needed to change the state of the physical machines (power on and off). Cardosa et al. [2009] explore the problem by handling several parameters of CMS for resource-sharing VMs, including minimum, maximum, and proportion of CPU being allocated. Kim et al. [2011] use DVFS-enabled infrastructure to adjust the hardware demands to actual real-time services needs. On the side of the hypervisors, Nathuji and Schwan

[2007] present VirtualPower, an extension that associates VMs with software CPU power state, as compared to the hypervisor conventional power states of a CPU. This allows hardware and software to be coordinated to use the best power mode and to use DVFS also in virtualized modes. Stoess et al. [2007] developed a low-level, fine-grained energy account system for hypervisors to allow power capping for guests. Additionally, working on the infrastructure itself in coordination with the VM management is also investigated.

5.2.2. VM Placement. On the VM placement side, Beloglazov and Buyya [2010] presented an architecture for mapping VMs to servers, applying an adapted version of the Best Fit Decreasing heuristic, a family of heuristics designed originally for bin-packing. Solutions given by heuristics can be far from optimal, especially in the presence of heterogeneity. And, a solution using the minimum number of servers is not necessarily the solution requiring less energy. Sorting criteria are also required for the servers to decide which bins are to be filled first: A VM is mapped to the server that shows the least increase in energy consumption. Similarly, Borgetto et al. [2012a] proposed several means to sort the servers and the VM for the mapping phase, using several sorts of best-fit and first-fit algorithms together with an ad-hoc algorithm derived from vector packing. In Barbagallo et al. [2010], the authors use bio-inspired heuristics to find the most energy-efficient hosts, whereas Mazzucco et al. [2010] propose to maximize revenues in a cloud by turning on and off physical machines.

Interestingly, a number of works not dedicated to CMS can easily be adapted. Jobs are handled in a cluster infrastructure, but seeing these as VMs does not change the approach. For instance, Kamitsos et al. [2010] utilize a Markov decision process in order to find an optimal policy for powering nodes on and off, which makes it possible to find an optimal tradeoff between performance and energy. In Petrucci et al. [2010], the problem of job placement is described as a linear program. They solve it periodically using a control loop. They focus on a heterogeneous cluster enabling DVFS, and they propose a set of constraints for energy reduction while allowing task migration. Similarly Borgetto et al. [2012a] use linear program modeling, taking into account some SLA for jobs, and they propose vector packing heuristics to solve it. In Hoyer et al. [2010], statistical allocation planning is proposed through two methods. The first approach allocates pessimistically to each job the maximum resource ratio it might need, developing an allocation directed by vector packing. The optimistic second approach overbooks each node while still guaranteeing to each job a certain performance threshold with dynamic monitoring of VM instances.

5.2.3. VM Migration and Consolidation. The third possibility is investigating VM (live) migration combined with physical machine consolidation. Liu et al. [2011a] have studied live migration of VMs in order to model the performance and energy use of migration. They show that migration is an I/O intensive application and that it consumes energy on both ends. The architectural framework proposed in Banerjee et al. [2010] for green clouds also achieves VM reconfiguration, allocation, and reallocation. The authors use a CPU power model to monitor the energy consumption of the cloud. The algorithm they propose to dynamically consolidate VMs significantly reduces the global power consumption of their infrastructure. Zhao and Huang [2009] have implemented a distributed load balancing algorithm using live migration for Eucalyptus [Nurmi et al. 2009], an open-source cloud computing platform offering Infrastructure as a Service (IaaS). They do not consider the memory capacity of the servers at all.

In OpenNebula, Choi et al. [2008] propose a machine learning framework that learns from experience when and where to migrate a VM in case of overload. In this approach, all possible migrations must be evaluated, leading to scalability problems for big infrastructures. Designed in the course of the GAMES project, the Green Cloud Scheduler

is integrated with OpenNebula. It proactively detects overprovisioned computing resources and identifies the most appropriate adaptation decisions to dynamically adjust them to the incoming workload. It generates adaptation action plans consisting of consolidation actions and hibernating or waking up servers and also uses a learning phase [Cioara et al. 2011b]. Berral et al. [2010] make dynamic resource allocation decisions using machine learning. They favor the allocation of new jobs to already powered nodes, using migration if necessary.

Entropy [Hermenier et al. 2009] uses constraint programming for the dynamic consolidation of resources in homogeneous clusters. It uses migration and accounts for migration overhead. In the context of the Fit4Green project, Quan et al. [2011] propose a framework for VM placement over a federation of data centers built using constraint programming and an Entropy system. Kumar et al. [2009] have developed and evaluated vManage, which places workloads under consideration of power, thermal, and performance aspects using stabilized first-fit and best-fit heuristics. pMapper [Verma et al. 2008] and Snooze [Feller et al. 2010] are other examples for cluster infrastructures. Snooze is based on a hierarchical agent structure that manages the placement and migration of VMs under the control of a centralized decision point. Snooze is extensible and can easily integrate different algorithms.

5.2.4. VM Scheduling. Finally, smart VM scheduling is also a source of energy savings. Burge et al. [2007] handle the request scheduling in a heterogeneous data center scenario. They focus on the decision of where and when to deploy a customer's job, and when deployed, a job can't move. They employ economic models considering the varying patience of customers, job length, consumed energy, job revenue, cancellation costs, and more. Their conclusion is that even using very simple heuristics (e.g., shutting down a server that has been idle for the past few minutes) can save a significant amount of energy. Steinder et al. [2007] have investigated a similar scenario. Beloglazov et al. [2012] propose energy-efficiency management of clouds through architectural guidelines, as well as QoS-aware scheduling algorithms and resource allocation policies. They perform simulations on their CloudSim toolkit. The scheduling of applications is also investigated in Berral et al. [2010] and Polverini et al. [2014].

5.3. Challenges and Research Directions

The main challenges in CMS and energy efficiency are the following: First, it is necessary to account for the precise energy consumption of each VM. In today's CMS, this is reduced to a simple calculation based on the number of hosted VMs on one host. Because each application will require different resources (some may use more CPU, memory, disk, or network resources), the share for each VM must be mathematically and precisely modeled.

Second, the interdependencies between possible leverages in the CMS must be further investigated. For example, mixing an initial allocation of VMs to physical hosts with post-adjustment of these hosts using DVFS can be counterproductive and suboptimal. Indeed, in that case, it can happen that the final setting is actually consuming more energy.

6. APPLIANCE

The Appliance subdomain represents a part of the Software domain, which performs actual useful work for cloud users. In a perfect cloud infrastructure, only Appliances would be consuming resources and thus energy. From a provider's perspective, appliance efficiency is only considered for SaaS and PaaS applications because an appliance is then under control of the provider and thus part of the cloud computing infrastructure. On the other hand, for lower level services (e.g., IaaS), an appliance is deployed

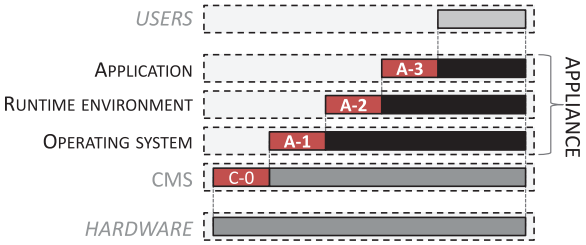

Fig. 8. Losses and wastes of appliance domain.

by a user, thus the user is responsible for its efficiency. This scenario falls under the User domain perspective. To date, software designers usually look at the quantity and proportionality of performance given the resource utilization. Now, to ensure energy-efficiency, software designers also need to consider the quantity and proportionality of resource utilization given the performance.

6.1. Context

The Appliance has a relatively smaller impact on the overall energy consumption than some other elements of the cloud infrastructure such as servers. On the other hand, appliances are responsible for the useful work, which is ultimately delivered to users. Hence, to adequately assess and manage the energy efficiency of the cloud, appliances must be taken into consideration. Three subsystems can be recognized in the appliance: an application that is used by the end user and that performs a main task of the appliance, a runtime environment required for running the application, and an operating system, which serves as a bridge between the physical or virtual machine and the software running on top of it.

The energy efficiency of appliances affects both energy loss and waste according to the model presented in Section 2; these are shown in Figure 8.

—**Application (A-1):** The application is the core part of the software appliance. There are different types of applications used in clouds. One of the most common types is a portal delivering Web content to end users. Other typical applications include databases and Web services. Some of these applications may even include large distributed computations, graphical rendering, simulations, and complex workflows hidden behind cloud Web interfaces. A well-known example of more advanced processing is MapReduce [Dean and Ghemawat 2008]. An application provides the core functionality to a user. Nevertheless, even on this level, losses and wastes of energy may take place. First, energy is usually consumed by additional components of the application. These modules are integral parts of the application, but they are also responsible for its nonfunctional aspects, such as security, reliability, control, logging, and the like. When the appliance is not strongly utilized by end users, these modules can be a source of energy consumption unrelated to any useful work. Energy is also consumed by supporting subsystems that are responsible for the maximization of appliance utilization and for dynamic adaptation of the appliance to load (according to goal W2). These subsystems are needed to optimize appliance efficiency (e.g., by stopping some services in the case of low load), but energy used by them is wasted because it is not used to deliver the core functionality of the appliance. Finally, part of the energy consumed by the application is not fully utilized by users. Energy can be consumed by the application running threads, processes or services, and allocated data structures in memory and on hard disks without producing any output.

For example, if a lower number of Web servers is sufficient for end users, then any additional servers waste energy.

—**Runtime environment (A-2):** Applications usually need a runtime environment to be executed from Java VMs or software-interpreting script languages such as Python, through Web servers such as Apache or Tomcat, and on to more complex systems. From the perspective of the appliance main task, energy consumed by the runtime environment is a loss that should be minimized (goal L2). Loss can be also caused by runtime environment overheads (e.g., programming languages or lack of optimization for given application type and hardware).

—**Operating system (A-3):** Both application and runtime environments must be executed on top of an operating system. The operating system can be an off-the-shelf system or a specific distribution tailored to appliance needs. Again, energy consumed purely by the operating system is a loss from the perspective of the appliance's main task. Especially heavy operating systems whose majority of functionality is not used by the appliance results in significant overheads (e.g., related to OS services, maintenance tasks, etc.).

In addition to these listed issues from the Appliance perspective, energy spent for running the CMS (C-0) is considered as entirely lost because it performs supporting tasks rather than the main task of the appliance.

A number of actions can be taken to reduce these energy losses and wastes according to the goals defined in Section 2. These goals with regards to appliances are as follow:

—**L1**. Proper implementation of cloud appliances can help to reduce energy losses. This can be done during the development phase by optimizing the implementation, as well as by using lightweight programming languages and only required libraries (A-2). The first step to achieve this goal is the use of a fine-grained monitoring and estimation of power usage in order to identify processes responsible for high energy consumption.

—**L2**. Although the Appliance subdomain represents IT software equipment, it can still have supporting systems that cause energy losses by creating a resource consumption overhead (e.g., a small application running on a heavy operating system) while using only a small percentage of its functions (A-3). Therefore, goal L2 includes reducing energy losses by proper implementation of applications and selection of an appropriate underlying technology. This should also include the use of reduced and customized operating systems and runtime environments. Similarly, as in the case of L1, precise information about the parts of a software responsible for high energy consumption must be identified to apply appropriate optimization.

—**W1**. Optimization of the appliance can reduce energy consumption by decreasing its resource usage or increasing its performance. Such an approach targets a goal (W1) of trying to reduce resource consumption while performing the same task. Applied techniques can focus on smart appliance management by switching off or reducing specific functionality in case of low or no load (A-1). To achieve low wastes, decisions should take into account available hardware so that the number of threads/processes or internal load balancing are optimized with energy efficiency in mind. Additionally, to meet goal W1, appliances need to be highly scalable in order to fully utilize available resources. For example, they should provide performance proportional to the consumed energy by scaling to a high number of cores, big cache sizes, high CPU frequency, and more. Otherwise, use of these resources should be treated as energy waste because they cause higher power usage without a proportional increase of performance. In the worst case, many cores and a large amount of memory can be allocated to an appliance that produces very little useful work.

—**W2**. Minimizing the unnecessary use of the appliance depends on the way users access it. Any smart functions applied must avoid breaking SLAs set with users. However, even with these constraints, a number of actions can be taken to reduce useless energy consumption. These techniques can include serving requests in batches, reducing the numbers of backups and checkpoints, limiting the number of service instances or threads, and adjusting the frequency of monitoring, polling, caching, and indexing. Overheads can be also related to the relevant functionality of appliances (e.g., security and resilience). Hence, applying most of these techniques requires finding a tradeoff between energy efficiency and other key aspect of appliances, such as performance, resilience, and security.

6.2. State of the Art

The energy efficiency of cloud appliances depends on a number of aspects including appliance development, compilation, deployment, and runtime phases. In addition, it is related to interaction with other elements of the cloud infrastructure, especially hardware, virtualization technology, and CMSs.

6.2.1. Design and Development. Agrawal and Sabharwal [2012] cover many issues related to the energy efficiency of software products. They provide recommendations and techniques for developing energy-efficient software, especially concentrating on reducing power usage by idle appliances. The authors also show that limiting wake-up events and changing timer activities leads to significant energy savings.

Some key principles to produce power-efficient software are proposed in Saxe [2010]. First, the amount of resources consumed should directly correspond to the amount of useful work done by the software appliance. In particular, if the appliance's useful work is lower, then the system should run in a lower state and the power usage should be decreased to the extent related to the useful work reduction. This corresponds to achieving the W1 goal defined in this article. Second, the software should minimize power usage in an idle state by reducing the amount of unnecessary computation (e.g., using a push instead of a pull mechanism), which enables it to remain dormant until action is actually required. This corresponds to achieving the L2 goal defined in this article. Third, if possible, software requests to access additional resources should be done infrequently and in batches, thus decreasing the number of unnecessary wake-ups. Additionally, as indicated in Smith and Sommerville [2010], attention should be paid to details such as avoiding memory leaks or freeing unallocated memory. Otherwise, "these problems will cause increased interference from the host operating system, resulting in additional energy consumption" [Smith and Sommerville 2010].

6.2.2. Compilers. Fakhar et al. [2012] propose a green compiler that applies a number of techniques to make code more energy efficient. These techniques are split into strategies for compilers and software development. They include cache skipping, use of register operands, instruction clustering and reordering, loop optimization, and more. They address the problem of overheads related to the use of energy-efficiency optimizations in the compiler, which corresponds to the W2 goal. Other research work on compilers that take energy efficiency into account are the encc [Falk and Lokuciejewski 2010] or Coffee [Raghavan et al. 2008] compilers; however, these do not focus on software development for clouds.

6.2.3. Application Monitoring. To improve the energy efficiency of appliances, their power usage must be monitored. Identifying the consumption of particular applications is a nontrivial problem, and attempts have been made to do this. For example, PowerTOP is a utility created by Intel that monitors a system and reports to the user which processes are responsible for wakeups that prevent a CPU from entering a sleep state. Other

tools that could be used to estimate application power usage are Joulemeter [Kansal et al. 2010] and ectop, developed within the scope of the CoolEmAll project [Berge et al. 2012]. There are also approaches to estimate power usage of servers based on specific characteristics of executed applications, such as those presented in Witkowski et al. [2013]. These solutions additionally allow users to identify which combination of application classes and hardware configurations is the most efficient. They focus more on HPC applications, but this is consistent with a current hot topic: HPC in the cloud and moving scientific applications to the cloud. Additionally, a similar methodology could be applied to cloud applications. Some attempts to do this were featured in projects such as Hemera [2013] and Magellan [2013].

Beik [2012] proposes an energy-aware software layer for more efficient energy usage. It collects micro- and macrometrics in order to efficiently use and deploy shared services in a shared cloud infrastructure.

6.2.4. Application Platforms. Studies have shown that a common situation in today's software is that a substantial amount of power is being consumed while system utilization is low. For example,a typical blade server can consume 50% of its peak power at only 10% of its utilization. Examples of overheads related to system monitoring are presented in Smith and Sommerville [2010] who indicate that an event-based architecture in which nodes are only contacted when they are needed to do some work would be more efficient in terms of power consumption but may suffer from poor performance or inaccurate information reporting. Engineers must examine tradeoffs of this type, and, if possible, implementations should be modified to suit system requirements.

Energy consumption overheads are related most often to monitoring, virtualization (addressed in the previous section), and operating systems, which are often responsible for significant power usage compared to the appliance itself. Therefore, substantial effort was invested into research on distributed cloud operating systems [Pianese et al. 2010; Smets-Solanes et al. 2011]. Nevertheless, their overhead and energy-efficiency characteristics should be studied in more detail.

6.3. Challenges and Research Directions

The main challenges related to the energy efficiency of cloud appliances include an appropriate appliance development process, minimizing appliance overheads, optimal selection of hardware and its configuration for given appliances, and proportional use of energy with regard to the useful work done.

Generally, it is important to enable software optimization with respect to energy consumption. Currently, software engineers usually optimize codes to achieve high performance, so guidelines for energy-efficiency optimization would be very valuable. Development of energy-efficient appliances requires the use of green compilers that optimize code for an energy-efficient mode. In addition to automated compiler optimizations, energy-efficient design patterns should be defined for developers. These could include single processor programs as well as distributed computing patterns (e.g., Map Reduce). Energy-efficiency goals are partially consistent with high-performance goals because scalability and short execution times often lead to minimized use of energy. However, sometimes performance and energy-efficiency goals are contradictory, and then the use of appropriate patterns and compiler options can be needed.

Another challenge, related to the goal W2 as well as to L1 and L2, is to reduce appliance overheads not related to the useful work. To this end, more work is needed on minimizing the overhead of appliance supporting components, OSs, libraries, and virtualization. The latter might include dynamically adjusting the size of VMs or providing sandboxes for applications instead of VMs and entire operating systems.

Proper assignment of appliances and hardware resources requires further investigation and the detailed classification of applications. Based on this, an optimal allocation of hardware to application classes should be studied. For instance, appliances suitable for microservers should be identified and ported.

Finally, running cloud appliances in an energy-efficient way requires communication between appliances and other domains, especially the CMS, to make optimal decisions. In particular, common decisions must be made based on both CMS and appliance monitoring, taking into account processing progress, appliance load, performance, state, data size, and more. These decisions may include migration of appliances, adjusting appliance size (e.g., VM size), defining the mode to be set, and the like. To this end, metrics that define appliance productivity and energy efficiency must be defined and measured.

7. INTERACTIONS BETWEEN CLOUD COMPUTING INFRASTRUCTURE DOMAINS

Cloud computing infrastructure represents a tightly coupled system composed of domains described in previous sections. Although, each domain can be analyzed separately, each domain influences another. Therefore, the entire cloud computing infrastructure, from the data center building to the smallest component, such as a CPU, has to be analyzed as a whole as well. In this section, we provide an overview of interactions between different infrastructure domains.

Appliance. The appliance is the smallest unit of manageable elements in cloud computing and represents the software that a user ultimately interacts with. For this reason, the greatest energy savings require studies of relations between appliances and all other domains, in particular the CMS and servers. Optimization of appliances to specific types of hardware may bring significant energy savings. For example, General-Purpose GPUs (GPGPU) are very energy-efficient provided that the application (or its parts) is implemented to make the most of GPGPU advantages. Similarly, some appliances can be run on microservers equipped with processors, such as ARMs, but not all can be easily ported without significant performance penalties. Even for a given hardware type, its power state may affect specific appliances in different ways. For example, depending on appliance characteristics, changes of CPU frequency and voltage will cause different performance and power usage values for CPU-bound and data-intensive applications.

CMS. The CMS, being at the center of application placement and scheduling management, must take these facts into consideration since its influence on other domains can be large. For instance, the local temperature and thus the behavior of fans and cooling infrastructure can be managed in thermal-aware solutions [Fu et al. 2010; Borgetto 2013]. However, most CMSs do not encompass this aspect in their solution, missing an important point. Finally, the way a system is implemented (e.g., scalability, components), how it interacts with underlying layers (e.g., hardware components, communication libraries, etc.), and how it is designed (e.g., architecture, supporting modules) affects the overall energy efficiency of the infrastructure. Losses and wastes are caused by both inefficiency of underlying layers and by their interactions with certain systems.

Servers. To support smart scheduling, hardware matching, and optimal decision making, the CMS needs detailed information about appliances and underlying hardware. This information includes progress, performance, state, and data size, as well as hardware metrics. In the context of energy efficiency, the most notable metric is the power consumption of a server. It is usually acquired with a power metering device, such as PowerMon [Bedard et al. 2010], or those integrated in the Power Distribution Unit

(PDU) or UPS unit. More detailed measurements can be performed for each component of a server, such as measuring instant current values of CPU power consumption with a circuit, as proposed in Borovyi et al. [2009]. Modeling VM power consumption is a next step in obtaining more detailed monitoring data [Mobius et al. 2013]. Furthermore, power consumption reductions can also be studied at a global scale via resource allocation. Le et al. [2009] propose cost reduction in a geographically distributed system. Their objective is to handle efficiently the variability between the energy costs of data centers and their architectural differences. They also use the time zone where these are located, as well as their proximity to green power sources. A similar approach is followed by Garg et al. [2009].

Network. Compared to power consumption for computing and cooling in a data center, the power consumption for network transport is still relatively small. As a result, this enables the location of computation at data centers where, for example, energy from renewable resources is available or where less energy for cooling is required due to a cool climate. This flexibility of computational placement enabled by efficient high- bandwidth network connections can result in a significant reduction of energy consumption for computation. However, when placing computation far from the end user, this also results in an increased latency limited by the speed of light in the optical fiber. Additionally, when using migration, the impact on the network cannot be completely ignored. Indeed, even if several researchers suggest that the impact of traffic can be ignored in terms of power consumption (i.e., the switches and routers consume roughly the same amount of energy no matter the bytes transferred [Hlavacs et al. 2009]), it cannot be so when considering that network components can be switched off or bandwidth adapted as in Adaptive Link Rate (ALR) for saving energy when not being used. Using models for power consumption during migrations [Liu et al. 2011b] can add to overall power consumption awareness when using such optimizations.

Cooling and power supply. The domains of cooling, power supply, and data center building are only briefly covered in this article, but surveys by Shuja et al. [2012], Beloglazov et al. [2011], and Jing et al. [2013] cover these from the energy efficiency perspective. Hoelzle and Barroso [2013] and Zomaya and Lee [2012] cover data center building, cooling, and power supply as related to energy efficiency, as well as cost. A comprehensive description of energy-efficient thermal management methods for data centers can be also found in Joshi and Kumar [2012].

Metrics. Finally, the overall energy efficiency of a data center can be measured using the Power Usage Effectiveness (PUE), which represents the ratio between total energy consumption of the facility and the ICT equipment. Details of PUE levels and measurement specification were defined in Avelar et al. [2012]. However, in this article, we focus on ICT equipment optimization, and PUE is not sufficient to represent such a level of detail. A metric such as Data Center infrastructure Effectiveness (DCiE) [Belady 2008] shows the inverse of PUE and thus inherits the same shortages. For example, PUE Scalability measures power proportionality—how the used power scales with load [Avelar et al. 2012]. Additionally, metrics focused on IT energy efficiency have been proposed. Examples of such metrics include TUE and ITUE, introduced in Patterson et al. [2013], which express total energy delivered into a data center divided by energy consumed by computational components and total energy delivered into ICT equipment divided by energy consumed by computational components, respectively. Other metrics include Carbon Usage Effectiveness (CUE), Water Usage Effectiveness (WUE), and Energy Reuse Effectiveness (ERE) and are covered in surveys [Kulseitova and Fong 2013; Cavdar and Alagoz 2012].

8. CONCLUSION

In this article, we analyzed the energy efficiency of a data center's ICT equipment, including the hardware and software that drives the cloud computing. First, we described our approach, which can be applied to an arbitrary system composed of smaller components/subsystems. Second, we introduced a breakdown of the cloud computing infrastructure by including all hardware and software equipment located in a data center. Third, we used a systematic approach to analyze energy efficiency of the ICT equipment and the software running on top of it by reviewing the available literature. Thus, we provided a holistic and uniform overview of data center ICT equipment with regards to energy efficiency.

Our analysis showed that many standard energy efficiency techniques do not work for cloud computing environments out of the box; rather, they have to be at least adapted or even designed from the scratch. This is due to the stratification of the cloud computing infrastructure, which comprises systems and components from different research areas, such as power supply, cooling, computing, and more. Optimizing these systems separately does improve the energy efficiency of the entire system; however, applying shared energy-efficiency techniques to multiple systems or their components can significantly improve energy efficiency if the techniques are aware of their interactions and dependencies.

REFERENCES

Dennis Abts, Michael R. Marty, Philip M. Wells, Peter Klausler, and Hong Liu. 2010. Energy proportional datacenter networks. *SIGARCH Computer Architecture News* 38, 3 (June 2010), 338–347.

Giuseppe Aceto, Alessio Botta, Walter de Donato, and Antonio Pescapè. 2013. Cloud monitoring: A survey. *Computer Networks* 0 (2013). DOI:http://dx.doi.org/10.1016/j.comnet.2013.04.001

Advisory Group. 2008. *ICT for Energy Efficiency*. Technical Report. DG-Information Society and Media, European Commission. Retrieved from http://ec.europa.eu/information_society/activities/sustainable_growth/docs/consultations/advisory_group_reports/ad-hoc_advisory_group_report.pdf.

A. Agrawal and M. Sabharwal. 2012. Enabling green IT through energy aware software. *IT Professional* PP, 99 (2012), 1. DOI:http://dx.doi.org/10.1109/MITP.2012.104

Mohammad Al-Fares, Alexander Loukissas, and Amin Vahdat. 2008. A scalable, commodity data center network architecture. *SIGCOMM Computer Communication Review* 38, 4 (August 2008), 63–74.

David G. Andersen, Jason Franklin, Michael Kaminsky, Amar Phanishayee, Lawrence Tan, and Vijay Vasudevan. 2009. FAWN: A fast array of wimpy nodes. In *Proceedings of the ACM SIGOPS 22Nd Symposium on Operating Systems Principles (SOSP'09)*. ACM, New York, NY, 1–14. DOI:http://dx.doi.org/10.1145/1629575.1629577

I. Anghel, T. Cioara, I. Salomie, G. Copil, D. Moldovan, and C. Pop. 2011. Dynamic frequency scaling algorithms for improving the CPU's energy efficiency. In *Proceedings of the 2011 International Conference on Intelligent Computer Communication and Processing (ICCP'11)*. 485–491. DOI:http://dx.doi.org/10.1109/ICCP.2011.6047920

P. Apparao, R. Iyer, and D. Newell. 2008. Implications of cache asymmetry on server consolidation performance. In *Proceedings of the IEEE International Symposium on Workload Characterization (IISWC'08)*. 24–32. DOI:http://dx.doi.org/10.1109/IISWC.2008.4636088

F. Armand and M. Gien. 2009. A practical look at micro-kernels and virtual machine monitors. In *Proceedings of the 6th IEEE Consumer Communications and Networking Conference (CCNC'09). IEEE*. 1–7. DOI:http://dx.doi.org/10.1109/CCNC.2009.4784874

ASHRAE. 2012. *Data Center Networking Equipment Issues and Best Practices*. Technical Report. ASHRAE Technical Committee (TC) 9.9.

Victor Avelar, Dan Azevedo, and Alan French. 2012. PUE (TM): A Comprehensive Examination of the Metric. Retrieved from http://www.thegreengrid.org/.

R. Ayoub, R. Nath, and T. Rosing. 2012. JETC: Joint energy thermal and cooling management for memory and CPU subsystems in servers. In *Proceedings of the 2012 IEEE 18th International Symposium on High Performance Computer Architecture (HPCA'12)*. 1–12. DOI:http://dx.doi.org/10.1109/HPCA.2012.6169035

Ayan Banerjee, Tridib Mukherjee, Georgios Varsamopoulos, and Sandeep K. S. Gupta. 2010. Cooling-aware and thermal-aware workload placement for green HPC data centers. In *Proceedings of the International Green Computing Conference*. IEEE, 245–256.

Donato Barbagallo, Elisabetta Di Nitto, Daniel J. Dubois, and Raffaela Mirandola. 2010. A bio-inspired algorithm for energy optimization in a self-organizing data center. In *Proceedings of the 1st International Conference on Self-organizing Architectures (SOAR'09)*. Springer-Verlag, Berlin, 127–151.

M. F. Bari, R. Boutaba, R. Esteves, L. Z. Granville, M. Podlesny, M. G. Rabbani, Qi Zhang, and M. F. Zhani. 2013. Data center network virtualization: A survey. *IEEE Communications Surveys Tutorials* 15, 2 (Second 2013), 909–928.

D. Bedard, Min Yeol Lim, R. Fowler, and A. Porterfield. 2010. PowerMon: Fine-grained and integrated power monitoring for commodity computer systems. In *Proceedings of the IEEE SoutheastCon 2010 (SoutheastCon'10)*. 479–484. DOI:http://dx.doi.org/10.1109/SECON.2010.5453824

R. Beik. 2012. Green cloud computing: An energy-aware layer in software architecture. In *Proceedings of the 2012 Spring Congress on Engineering and Technology (S-CET'12)*. 1–4. DOI:http://dx.doi.org/10.1109/SCET.2012.6341950

Christian Belady. 2008. Green Grid Data Center Power Efficiency Metrics: PUE and DCiE. Retrieved from http://www.thegreengrid.org/.

Christian Belady. 2011. *Projecting Annual New Datacenter Construction Market Size*. Technical Report. Microsoft Corp., US.

Anton Beloglazov, Jemal Abawajy, and Rajkumar Buyya. 2012. Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future Generation Computer Systems* 28, 5 (May 2012), 755–768. DOI:http://dx.doi.org/10.1016/j.future.2011.04.017

Anton Beloglazov and Rajkumar Buyya. 2010. Energy efficient allocation of virtual machines in cloud data centers. In *Proceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGRID'10)*. IEEE Computer Society, Washington, DC, 577–578. DOI:http://dx.doi.org/10.1109/CCGRID.2010.45

Anton Beloglazov, Rajkumar Buyya, Young Choon Lee, and Albert Y. Zomaya. 2011. A taxonomy and survey of energy-efficient data centers and cloud computing systems. *Advances in Computers* 82 (2011), 47–111.

Micha Berge, Georges Costa, Andreas Kopecki, Ariel Oleksiak, Jean-Marc Pierson, Tomasz Piontek, Eugen Volk, and Stefan Wesner. 2012. Modeling and simulation of data center energy-efficiency in CoolEmAll. In *Energy Efficient Data Centers*, Jyrki Huusko, Hermann Meer, Sonja Klingert, and Andrey Somov (Eds.). Lecture Notes in Computer Science, Vol. 7396. Springer, Berlin, 25–36. DOI:http://dx.doi.org/10.1007/978-3-642-33645-4_3

Josep Ll. Berral, Íñigo Goiri, Ramón Nou, Ferran Julià, Jordi Guitart, Ricard Gavaldà, and Jordi Torres. 2010. Towards energy-aware scheduling in data centers using machine learning. In *Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking (e-Energy'10)*. ACM, New York, NY, 215–224. DOI:http://dx.doi.org/10.1145/1791314.1791349

A. A. Bhattacharya, D. Culler, A. Kansal, S. Govindan, and S. Sankar. 2012. The need for speed and stability in data center power capping. In *Proceedings of the 2012 International Green Computing Conference (IGCC'12)*. 1–10. DOI:http://dx.doi.org/10.1109/IGCC.2012.6322253

Aruna Prem Bianzino, Claude Chaudet, Dario Rossi, and Jean-Louis Rougier. 2012. A survey of green networking research. *IEEE Communications Surveys and Tutorials* 14, 1 (2012), 3–20.

Wolfgang Bittermann and Manfred Gollner. 2011. *Modelling of Power Consumption in Private Households in Austria According to Type of Usage*. Technical Report. Statistics Austria, Directorate Spatial Statistics, Energy, Vienna, Austria.

Raffaele Bolla, Roberto Bruschi, Franco Davoli, and Flavio Cucchietti. 2011. Energy efficiency in the future internet: A survey of existing approaches and trends in energy-aware fixed network infrastructures. *IEEE Communications Surveys and Tutorials* 13, 2 (2011), 223–244.

Damien Borgetto. 2013. *Allocation et Rallocation de Services pour les conomies d'nergie dans les clusters et les cloues*. Ph.D. Dissertation. University Paul Sabatier, Toulouse 3.

Damien Borgetto, Henri Casanova, Georges Da Costa, and Jean-Marc Pierson. 2012a. Energy-aware service allocation. *Future Generation Computer Systems* 28, 5 (May 2012), 769–779. DOI:http://dx.doi.org/10.1016/j.future.2011.04.018

Damien Borgetto, Michael Maurer, Georges Da-Costa, Jean-Marc Pierson, and Ivona Brandic. 2012c. Energy-efficient and SLA-aware management of IaaS clouds. In *Proceedings of the 3rd International Conference on Future Energy Systems: Where Energy, Computing and Communication Meet (e-Energy'12)*. ACM, New York, NY, Article 25, 10 pages. DOI:http://dx.doi.org/10.1145/2208828.2208853

A. Borovyi, V. Kochan, Z. Dombrovskyy, V. Turchenko, and A. Sachenko. 2009. Device for measuring instant current values of CPU's energy consumption. In *Proceedings of the IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'09)*. 126–130. DOI:http://dx.doi.org/10.1109/IDAACS.2009.5343010

Tom Bostoen, Sape Mullender, and Yolande Berbers. 2013. Power-reduction techniques for data-center storage systems. *ACM Computer Survey* 45, 3, Article 33 (July 2013), 38 pages. DOI:http://dx.doi.org/10.1145/2480741.2480750

Jennifer Burge, Parthasarathy Ranganathan, and Janet L. Wiener. 2007. Cost-aware scheduling for heterogeneous enterprise machines (CASH'EM). In *CLUSTER*. IEEE, 481–487.

Michael Cardosa, Madhukar R. Korupolu, and Aameek Singh. 2009. Shares and utilities based power consolidation in virtualized server environments. In *Proceedings of the 11th IFIP/IEEE International Conference on Symposium on Integrated Network Management (IM'09)*. IEEE Press, Piscataway, NJ, 327–334.

D. Cavdar and F. Alagoz. 2012. A survey of research on greening data centers. In *Proceedings of the 2012 IEEE Global Communications Conference (GLOBECOM'12)*. 3237–3242.

Feng Chen and Xiaodong Zhang. 2008. Caching for bursts (C-Burst): Let hard disks sleep well and work energetically. In *Proceedings of the 2008 ACM/IEEE International Symposium on Low Power Electronics and Design (ISLPED'08)*. 141–146. DOI:http://dx.doi.org/10.1145/1393921.1393961

Longxiang Chen, Panruo Wu, Zizhong Chen, Rong Ge, and Ziliang Zong. 2012a. Energy efficient parallel matrix-matrix multiplication for DVFS-enabled clusters. In *Proceedings of the 2012 41st International Conference on Parallel Processing Workshops (ICPPW'12)*. 239–245. DOI:http://dx.doi.org/10.1109/ICPPW.2012.36

Tseng-Yi Chen, Tsung-Tai Yeh, Hsin-Wen Wei, Yu-Xun Fang, Wei-Kuan Shih, and Tsan-Sheng Hsu. 2012b. CacheRAID: An efficient adaptive write cache policy to conserve RAID disk array energy. In *Proceedings of the 2012 IEEE 5th International Conference on Utility and Cloud Computing (UCC'12)*. 117–124. DOI:http://dx.doi.org/10.1109/UCC.2012.26

G. L. T. Chetsa, L. Lefevre, J. Pierson, P. Stolf, and G. Da Costa. 2012. Beyond CPU frequency scaling for a fine-grained energy control of HPC systems. In *Proceedings of the 2012 IEEE 24th International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD'12)*. 132–138. DOI:http://dx.doi.org/10.1109/SBAC-PAD.2012.32

Hyung Won Choi, Hukeun Kwak, Andrew Sohn, and Kyusik Chung. 2008. Autonomous learning for efficient resource utilization of dynamic VM migration. In *Proceedings of the 22nd Annual International Conference on Supercomputing (ICS'08)*. ACM, New York, NY, 185–194. DOI:http://dx.doi.org/10.1145/1375527.1375556

Antonio Cianfrani, Vincenzo Eramo, Marco Listanti, Marco Polverini, and Athanasios V. Vasilakos. 2012. An OSPF-integrated routing strategy for QoS-aware energy saving in IP backbone networks. *IEEE Transactions on Network and Service Management* 9, 3 (2012), 254–267.

T. Cioara, I. Anghel, I. Salomie, G. Copil, D. Moldovan, and M. Grindean. 2011a. Time series based dynamic frequency scaling solution for optimizing the CPU energy consumption. In *Proceedings of the 2011 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP'11)*. 477–483. DOI:http://dx.doi.org/10.1109/ICCP.2011.6047919

Tudor Cioara, Ionut Anghel, Ioan Salomie, Georgiana Copil, Daniel Moldovan, and Alexander Kipp. 2011b. Energy aware dynamic resource consolidation algorithm for virtualized service centers based on reinforcement learning. In *ISPDC*. IEEE, 163–169.

Citrix. 2012. Xen - Hypervisor. Retrieved from http://xen.org/.

Holger Claussen, Imran Ashraf, and Lester Tse Wee Ho. 2010. Dynamic idle mode procedures for femtocells. *Bell Labs Technical Journal* 15, 2 (Aug. 2010), 95–116.

Holger Claussen, Hamid Reza Karimi, and Bernard Mulgrew. 2005. Low complexity detection of high-order modulations in multiple antenna systems. *IEE Proceedings Communications* 152, 6 (December 2005), 789–796.

Holger Claussen, Lester Tse Wee Ho, and Florian Pivit. 2009. Leveraging advances in mobile broadband technology to improve environmental sustainability. *Telecommunications Journal of Australia* 59, 1 (Feb. 2009), 4. 1–4.18.

Dennis Colarelli and Dirk Grunwald. 2002. Massive arrays of idle disks for storage archives. In *Proceedings of the 2002 ACM/IEEE Conference on Supercomputing (Supercomputing'02)*. IEEE Computer Society Press, Los Alamitos, CA, 1–11.

Coolit. 2013. Coolit Systems. Retrieved from http://www.coolitsystems.com/.

P. de Langen and B. Juurlink. 2009. Limiting the number of dirty cache lines. In *Proceedings of the Design, Automation Test in Europe Conference Exhibition (DATE'09)*. 670–675. DOI:http://dx.doi.org/10.1109/DATE.2009.5090750

Jeffrey Dean and Sanjay Ghemawat. 2008. MapReduce: Simplified data processing on large clusters. *Communications of the ACM* 51, 1 (Jan. 2008), 107–113. DOI:http://dx.doi.org/10.1145/1327452.1327492

D. Dharwar, S. S. Bhat, V. Srinivasan, D. Sarma, and P. K. Banerjee. 2012. Approaches towards energy-efficiency in the cloud for emerging markets. In *Proceedings of the 2012 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM'12)*. 1–6. DOI:http://dx.doi.org/10.1109/CCEM.2012.6354599

R. G. Dreslinski, G. K. Chen, T. Mudge, D. Blaauw, D Sylvester, and K. Flautner. 2008. Reconfigurable energy efficient near threshold cache architectures. In *Proceedings of the 2008 41st IEEE/ACM International Symposium on Microarchitecture (MICRO-41)*. 459–470. DOI:http://dx.doi.org/10.1109/MICRO.2008.4771813

Ronald G. Dreslinski, David Fick, David Blaauw, Dennis Sylvester, and Trevor Mudge. 2009. Reconfigurable multicore server processors for low power operation. In *Proceedings of the 9th International Workshop on Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS'09)*. Springer-Verlag, Berlin, 247–254. DOI:http://dx.doi.org/10.1007/978-3-642-03138-0_27

EARTH. 2011. Homepage. Retrieved from https://www.ict-earth.eu/.

ECONET. 2013. Homepage. Retrieved from http://www.econet-project.eu/.

U.S. EIA. 2013. *International Energy Outlook 2013 with Projections to 2040*. Technical Report. U.S. Energy Information Administration, Office of Energy Analysis, U.S. Department of Energy. Retrieved from http://www.eia.gov/forecasts/ieo/pdf/0484(2013).pdf.

Vincent C. Emeakaroha, Marco A. S. Netto, Rodrigo N. Calheiros, Ivona Brandic, Rajkumar Buyya, and César A. F. De Rose. 2012. Towards autonomic detection of SLA violations in cloud infrastructures. *Future Generation Computer Systems* 28, 7 (2012), 1017–1029. DOI:http://dx.doi.org/10.1016/j.future.2011.08.018

Emerson. 2008. *Data Center Users Group Special Report: Inside the Data Center 2008 and Beyond*. Technical Report. Data Center Users' Group, Emerson Electric Co., Emerson Network Power, Ohio.

Emerson. 2009. *Energy Logic: Reducing Data Center Energy Consumption by Creating Savings That Cascade across Systems*. Technical Report. Emerson Network Power, Emerson Electric Co., Emerson Network Power, Ohio.

Emerson. 2010. *Data Center Users Group Special Report: Growing Concerns over Data Center Infrastructure Monitoring and Management*. Technical Report. Data Center Users' Group, Emerson Electric Co., Emerson Network Power, Ohio.

Enerdata. 2011. Average Electricity Consumption per Electrified Household. Retrieved from http://www.wec-indicators.enerdata.eu/household-electricity-use.html.

Enerdata. 2014. Global Energy Statistical Yearbook 2014. Retrieved from http://yearbook.enerdata.net/.

EU. 2008. *Addressing the Challenge of Energy Efficiency through Information and Communication Technologies*. Technical Report. European Commission, European Communities.

EU. 2010. *A Digital Agenda for Europe*. Technical Report. European Commission, European Communities. http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=com:2010:0245:fin:e n:pdf.

EU. 2011. *Energy Efficiency Plan 2011*. Technical Report. European Commission, European Communities.

Faiza Fakhar, Barkha Javed, Raihan ur Rasool, Owais Malik, and Khurram Zulfiqar. 2012. Software level green computing for large scale systems. *Journal of Cloud Computing* 1, 1 (2012), 1–17. DOI:http://dx.doi.org/10.1186/2192-113X-1-4

Heiko Falk and Paul Lokuciejewski. 2010. A compiler framework for the reduction of worst-case execution times. *Real-Time Systems* 46, 2 (2010), 251–300. DOI:http://dx.doi.org/10.1007/s11241-010-9101-x

Xiaobo Fan, Wolf-Dietrich Weber, and Luiz Andre Barroso. 2007. Power provisioning for a warehouse-sized computer. *SIGARCH Computer Architecture News* 35, 2 (June 2007), 13–23. DOI:http://dx.doi.org/10.1145/1273440.1250665

Andrew Fanara. 2007. *Report to Congress on Server and Data Center Energy Efficiency*. Technical Report. U.S. Environmental Protection Agency, ENERGY STAR Program.

L. M. Feeney and M. Nilsson. 2001. Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'01)*. Vol. 3. IEEE, 1548–1557.

Eugen Feller. 2012. *Autonomic an Energy-Efficient Management of Large-Scale Virtualized Data Centers*. Ph.D. Dissertation. University Rennes 1.

E. Feller, C. Morin, and A. Esnault. 2012. A case for fully decentralized dynamic VM consolidation in clouds. In *Proceedings of the 2012 IEEE 4th International Conference onCloud Computing Technology and Science (CloudCom'12)*. 26–33. DOI:http://dx.doi.org/10.1109/CloudCom.2012.6427585

Eugen Feller, Louis Rilling, Christine Morin, Renaud Lottiaux, and Daniel Leprince. 2010. Snooze: A scalable, fault-tolerant and distributed consolidation manager for large-scale clusters. In *Proceedings of the 2010 IEEE/ACM International Conference on Green Computing and Communications & International Conference on Cyber, Physical and Social Computing (GREENCOM-CPSCOM'10)*. IEEE Computer Society, Washington, DC, 125–132. DOI:http://dx.doi.org/10.1109/GreenCom-CPSCom.2010.62

W. Felter, A. Hylick, and J. Carter. 2011. Reliability-aware energy management for hybrid storage systems. In *Proceedings of the 2011 IEEE 27th Symposium on Mass Storage Systems and Technologies (MSST'11)*. 1–13. DOI: http://dx.doi.org/10.1109/MSST.2011.5937221

I. Foster, Yong Zhao, I. Raicu, and Shiyong Lu. 2008. Cloud computing and grid computing 360-degree compared. In *Proceedings of the 2008 Grid Computing Environments Workshop (GCE'08)*. 1–10.

Yong Fu, Chenyang Lu, and Hongan Wang. 2010. Robust control-theoretic thermal balancing for server clusters. In *IPDPS*. IEEE, 1–11.

Saurabh Kumar Garg, Chee Shin Yeo, Arun Anandasivam, and Rajkumar Buyya. 2009. Energy-efficient scheduling of HPC applications in cloud computing environments. *CoRR* abs/0909.1146 (2009).

Chang Ge, Zhili Sun, and Ning Wang. 2013. A survey of power-saving techniques on data centers and content delivery networks. *IEEE Communications Surveys Tutorials* 15, 3 (Third 2013), 1334–1354.

Xiongzi Ge, Dan Feng, and D. H. C. Du. 2011. DiscPOP: Power-aware buffer management for disk accesses. In *Proceedings of the 2011 International Green Computing Conference and Workshops (IGCC'11)*. 1–6. DOI: http://dx.doi.org/10.1109/IGCC.2011.6008590

A. Grebennikov and S. Bulja. 2012. High-efficiency Doherty power amplifiers: Historical aspect and modern trends. *Proceedings of the IEEE* 100, 12 (2012), 3190–3219.

Steve Greenberg, Evan Mills, Bill Tschudi, Peter Rumsey, and Bruce Myatt. 2006. Best practices for data centers: Lessons learned from benchmarking 22 data centers. In *ACEE Summer Study on Energy Efficiency in Buildings*.

GreenTouch. 2012. Homepage. Retrieved from http://www.greentouch.org/.

C. Gunaratne, K. Christensen, B. Nordman, and S. Suen. 2008. Reducing the energy consumption of ethernet with adaptive link rate (ALR). *IEEE Transactions on Computers* 57, 4 (2008), 448–461.

Chuanxiong Guo, Guohan Lu, Dan Li, Haitao Wu, Xuan Zhang, Yunfeng Shi, Chen Tian, Yongguang Zhang, and Songwu Lu. 2009. BCube: A high performance, server-centric network architecture for modular data centers. *SIGCOMM Computer Communication Review* 39, 4 (Aug. 2009), 63–74.

Chuanxiong Guo, Haitao Wu, Kun Tan, Lei Shi, Yongguang Zhang, and Songwu Lu. 2008. Dcell: A scalable and fault-tolerant network structure for data centers. *SIGCOMM Computer Communication Review* 38, 4 (Aug. 2008), 75–86.

Maruti Gupta and Suresh Singh. 2003. Greening of the internet. In *SIGCOMM*. 19–26.

M. Gupta and S. Singh. 2007a. Using low-power modes for energy conservation in ethernet LANs. In *Proceedings of the 26th IEEE International Conference on Computer Communications (INFOCOM'07). IEEE,* 2451–2455.

Maruti Gupta and Suresh Singh. 2007b. Using low-power modes for energy conservation in ethernet LANs. In *INFOCOM*. 2451–2455.

Sudhanva Gurumurthi, Anand Sivasubramaniam, Mahmut Kandemir, and Hubertus Franke. 2003. DRPM: Dynamic speed control for power management in server class disks. *SIGARCH Computer Architecture News* 31, 2 (May 2003), 169–181. DOI: http://dx.doi.org/10.1145/871656.859638

C. Hankendi, S. Reda, and A. K. Coskun. 2013. vCap: Adaptive power capping for virtualized servers. In *Proceedings of the 2013 IEEE International Symposium on Low Power Electronics and Design (ISLPED'13)*.415–420. DOI: http://dx.doi.org/10.1109/ISLPED.2013.6629334

A. Haywood, J. Sherbeck, P. Phelan, G. Varsamopoulos, and S. K. S. Gupta. 2012. Investigating a relationship among CPU and system temperatures, thermal power, and CPU tasking levels. In *Proceedings of the 2012 13th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm'12)*.821–827. DOI: http://dx.doi.org/10.1109/ITHERM.2012.6231511

Brandon Heller, Srini Seetharaman, Priya Mahadevan, Yiannis Yiakoumis, Puneet Sharma, Sujata Banerjee, and Nick McKeown. 2010. ElasticTree: Saving energy in data center networks. In *Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation (NSDI'10)*. USENIX Association, Berkeley, CA, 17.

Hemera. 2013. Hemera. Retrieved http://www.grid5000.fr/mediawiki/index.php/Hemera.

Fabien Hermenier, Xavier Lorca, Jean-Marc Menaud, Gilles Muller, and Julia Lawall. 2009. Entropy: A consolidation manager for clusters. In *Proceedings of the 2009 ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments (VEE'09)*. ACM, New York, NY, 41–50. DOI: http://dx.doi.org/10.1145/1508293.1508300

Helmut Hlavacs, Georges Da Costa, and Jean-Marc Pierson. 2009. Energy consumption of residential and professional switches (regular paper). In *International Conference on Computational Science and Engineering (CSE), Vancouver, Canada, 29/08/2009-31/08/2009*. IEEE Computer Society, 240–246.

B. M. Hochwald and S. Ten Brink. 2003. Achieving near-capacity on a multiple-antenna channel. *Communications, IEEE Transactions on* 51, 3 (2003), 389–399.

Urs Hoelzle and Luiz Andre Barroso. 2013. *The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines* (2nd ed.). Morgan and Claypool.

Marko Hoyer, Kiril Schrder, and Wolfgang Nebel. 2010. Statistical static capacity management in virtualized data centers supporting fine grained QoS specification. In *e-Energy*, Herman de Meer, Suresh Singh, and Torsten Braun (Eds.). ACM, 51–60.

Lei Huang, Qin Jia, Xin Wang, Shuang Yang, and Baochun Li. 2011. PCube: Improving power efficiency in data center networks. In *Proceedings of the 2011 IEEE International Conference on Cloud Computing (CLOUD'11)*. 65–72.

IBM. 2013. Implementing an IBM System x iDataPlex Solution. Retrieved from http://www.redbooks.ibm.com/redbooks/pdfs/sg247629.pdf.

Icetope. 2013. Iceotope. http://www.iceotope.com/.

Yichao Jin, Yonggang Wen, and Qinghua Chen. 2012. Energy efficiency and server virtualization in data centers: An empirical investigation. In *Proceedings of the 2012 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. 133–138. DOI:http://dx.doi.org/10.1109/INFCOMW.2012.6193474

Si-Yuan Jing, Shahzad Ali, Kun She, and Yi Zhong. 2013. State-of-the-art research study for green cloud computing. *The Journal of Supercomputing* 65, 1 (2013), 445–468. DOI:http://dx.doi.org/10.1007/s11227-011-0722-1

Yogendra Joshi and Pramod Kumar. 2012. *Energy Efficient Thermal Management of Data Centers* (14th ed.). Springer.

A. B. Kahng, Seokhyeong Kang, R. Kumar, and J. Sartori. 2013. Enhancing the efficiency of energy-constrained DVFS designs. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 21, 10 (2013), 1769–1782. DOI:http://dx.doi.org/10.1109/TVLSI.2012.2219084

Ioannis Kamitsos, Lachlan L. H. Andrew, Hongseok Kim, and Mung Chiang. 2010. Optimal sleep patterns for serving delay-tolerant jobs. In *e-Energy*, Herman de Meer, Suresh Singh, and Torsten Braun (Eds.). ACM, 31–40.

Aman Kansal, Feng Zhao, Jie Liu, Nupur Kothari, and Arka A. Bhattacharya. 2010. Virtual machine power metering and provisioning. In *Proceedings of the 1st ACM Symposium on Cloud Computing (SoCC'10)*. ACM, New York, NY, 39–50. DOI:http://dx.doi.org/10.1145/1807128.1807136

Stefanos Kaxiras and Margaret Martonosi. 2008. *Computer Architecture Techniques for Power-Efficiency* (1st ed.). Morgan and Claypool.

D. C. Kilper, G. Atkinson, S. K. Korotky, S. Goyal, P. Vetter, D. Suvakovic, and O. Blume. 2011. Power trends in communication networks. *IEEE Journal on Selected Topics in Quantum Electron.* 17, 2 (March-April 2011), 275–284.

Hyunhee Kim, Jung Ho Ahn, and Jihong Kim. 2013. Exploiting replicated cache blocks to reduce L2 cache leakage in CMPs. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 21, 10 (2013), 1863–1877. DOI:http://dx.doi.org/10.1109/TVLSI.2012.2220791

Kyong Hoon Kim, Anton Beloglazov, and Rajkumar Buyya. 2011. Power-aware provisioning of virtual machines for real-time cloud services. *Concurrent Computing: Practical Experiments* 23, 13 (Sept. 2011), 1491–1505. DOI:http://dx.doi.org/10.1002/cpe.1712

Shin Gyu Kim, Chanho Choi, Hyeonsang Eom, H. Y. Yeom, and Huichung Byun. 2012. Energy-centric DVFS controling method for multi-core platforms. In *High Performance Computing, Networking, Storage and Analysis (SCC), 2012 SC Companion*. 685–690. DOI:http://dx.doi.org/10.1109/SC.Companion.2012.94

Jonathan G. Koomey. 2008. Worldwide electricity used in data centers. *Environmental Research Letters* 3, 3 (2008), 034008 (8pp).

Jonathan G. Koomey. 2011. Growth in data center electricity use 2005 to 2010. *Environmental Research Letters* 3, 3 (2011), 034008.

A. Kulseitova and Ang Tan Fong. 2013. A survey of energy-efficient techniques in cloud data centers. In *Proceedings of the 2013 International conference on ICT for Smart Society (ICISS'13)*. 1–5.

Sanjay Kumar, Vanish Talwar, Vibhore Kumar, Parthasarathy Ranganathan, and Karsten Schwan. 2009. vManage: Loosely coupled platform and virtualization management in data centers. In *Proceedings of the 6th International Conference on Autonomic Computing (ICAC'09)*. ACM, New York, NY, 127–136. DOI:http://dx.doi.org/10.1145/1555228.1555262

Daniel Guimaraes do Lago, Edmundo R. M. Madeira, and Luiz Fernando Bittencourt. 2011. Power-aware virtual machine scheduling on clouds using active cooling control and DVFS. In *Proceedings of the 9th International Workshop on Middleware for Grids, Clouds and e-Science (MGC'11)*. ACM, New York, NY, , Article 2, 6 pages. DOI:http://dx.doi.org/10.1145/2089002.2089004

Nikolaos Laoutaris, Michael Sirivianos, Xiaoyuan Yang, and Pablo Rodriguez. 2011. Inter-datacenter bulk transfers with netstitcher. *SIGCOMM Computer Communication Review* 41, 4 (Aug. 2011), 74–85.

K. Le, R. Bianchini, M. Martonosi, and T. Nguyen. 2009. Cost-and energy-aware load distribution across data centers. *Proceedings of HotPower*.

DongKyu Lee and Kern Koh. 2009. Popularity based cache management scheme for RAID which uses an SSD as a cache. In *Proceedings of the 4th International Conference on Computer Sciences and Convergence Information Technology (ICCIT'09)*. 913–915. DOI:http://dx.doi.org/10.1109/ICCIT.2009.310

Dong Li and Jun Wang. 2004. EERAID: Energy efficient redundant and inexpensive disk array. In *Proceedings of the 11th Workshop on ACM SIGOPS European Workshop (EW 11)*. ACM, New York, NY, Article 29. DOI:http://dx.doi.org/10.1145/1133572.1133577

Jiang Lin, Hongzhong Zheng, Zhichun Zhu, Howard David, and Zhao Zhang. 2007. Thermal modeling and management of DRAM memory systems. *SIGARCH Computer Architecture News* 35, 2 (June 2007), 312–322. DOI:http://dx.doi.org/10.1145/1273440.1250701

Haikun Liu, Cheng-Zhong Xu, Hai Jin, Jiayu Gong, and Xiaofei Liao. 2011a. Performance and energy modeling for live migration of virtual machines. In *Proceedings of the 20th International Symposium on High Performance Distributed Computing (HPDC'11)*. ACM, New York, NY, 171–182. DOI:http://dx.doi.org/10.1145/1996130.1996154

Haikun Liu, Cheng-Zhong Xu, Hai Jin, Jiayu Gong, and Xiaofei Liao. 2011b. Performance and energy modeling for live migration of virtual machines. In *Proceedings of the 20th International Symposium on High Performance Distributed Computing (HPDC'11)*. ACM, New York, NY, 171–182. DOI:http://dx.doi.org/10.1145/1996130.1996154

Liang Liu, Hao Wang, Xue Liu, Xing Jin, Wen Bo He, Qing Bo Wang, and Ying Chen. 2009. GreenCloud: A new architecture for green data center. In *Proceedings of the 6th International Conference Industry Session on Autonomic Computing and Communications Industry Session (ICAC-INDST'09)*. ACM, New York, NY, 29–38. DOI:http://dx.doi.org/10.1145/1555312.1555319

D. Lopez-Perez, Xiaoli Chu, A. V. Vasilakos, and H. Claussen. 2014. Power minimization based resource allocation for interference mitigation in OFDMA femtocell networks. *IEEE Journal on Selected Areas in Communications* 32, 2 (Feb. 2014), 333–344.

Kun Ma, Runyuan Sun, and A. Abraham. 2012. Toward a lightweight framework for monitoring public clouds. In *Proceedings of the 2012 4th International Conference on Computational Aspects of Social Networks (CASoN'12)*. .361–365. DOI:http://dx.doi.org/10.1109/CASoN.2012.6412429

Magellan. 2013. Magellan. Retrieved from http://www.cloudbook.net/directories/research-clouds/research-project.php?i d=100047.

Priya Mahadevan, Puneet Sharma, Sujata Banerjee, and Parthasarathy Ranganathan. 2009. A power benchmarking framework for network devices. In *Proceedings of the 8th International IFIP-TC 6 Networking Conference (NETWORKING'09)*. Springer-Verlag, Berlin, 795–808.

Ajay Mahimkar, Angela Chiu, Robert Doverspike, Mark D. Feuer, Peter Magill, Emmanuil Mavrogiorgis, Jorge Pastor, Sheryl L. Woodward, and Jennifer Yates. 2011. Bandwidth on demand for inter-data center communication. In *Proceedings of the 10th ACM Workshop on Hot Topics in Networks (HotNets-X'11)*. ACM, New York, NY, Article 24, 6 pages.

A. Manzanares, K. Bellam, and Xiao Qin. 2008. A prefetching scheme for energy conservation in parallel disk systems. In *Proceedings of the IEEE International Symposium on Parallel and Distributed Processing (IPDPS'08)*. 1–5. DOI:http://dx.doi.org/10.1109/IPDPS.2008.4536421

T. Mastelic and I. Brandic. 2013. TimeCap: Methodology for comparing IT infrastructures based on time and capacity metrics. In *Proceedings of the 2013 IEEE 6th International Conference on Cloud Computing (CLOUD'13)*. 131–138. DOI:http://dx.doi.org/10.1109/CLOUD.2013.130

Toni Mastelic, Vincent Emeakaroha, Michael Maurer, and Ivona Brandic. 2012. M4Cloud - generic application level monitoring for resource-shared cloud environments. In *Proceedings of the 2nd International Conference on Cloud Computing and Services Science (CLOSER'12)*.

Michele Mazzucco, Dmytro Dyachuk, and Ralph Deters. 2010. Maximizing cloud providers' revenues via energy aware allocation policies. In *IEEE CLOUD*. IEEE, 131–138.

R. K. Megalingam, T. S. Hassan, P. Vivek, A. Mohan, and M. T. Rao. 2009. Power consumption reduction in CPU datapath using a novel clocking scheme. In *Proceedings of the 2nd International Conference on Computer Science and Information Technology (ICCSIT'09)*. 529–533. DOI:http://dx.doi.org/10.1109/ICCSIT.2009.5234658

Peter Mell and Tim Grance. 2009. *The NIST Definition of Cloud Computing* (26th ed.). Technical Report. National Institute of Standards and Technology, Information Technology Laboratory. Retrieved from http://www.csrc.nist.gov/groups/SNS/cloud-computing/.

R. Y. Mesleh, H. Haas, S. Sinanovic, Chang Wook Ahn, and Sangboh Yun. 2008. Spatial modulation. *IEEE Transactions on Vehicular Technology* 57, 4 (2008), 2228–2241.

C. Mobius, W. Dargie, and A. Schill. 2013. Power consumption estimation models for processors, virtual machines, and servers. *IEEE Transactions on Parallel and Distributed Systems* PP, 99 (2013), 1–1. DOI:http://dx.doi.org/10.1109/TPDS.2013.183

Francois Moisan and Didier Bosseboeuf. 2010. *Energy Efficiency: A Recipe for Success*. Technical Report. World Energy Council, London, UK.

T. Mudge and U. Holzle. 2010. Challenges and opportunities for extremely energy-efficient processors. *IEEE Micro* 30, 4 (July-Aug. 2010), 20–24. DOI:http://dx.doi.org/10.1109/MM.2010.61

Ripal Nathuji and Karsten Schwan. 2007. VirtualPower: Coordinated power management in virtualized enterprise systems. In *Proceedings of 21st ACM SIGOPS Symposium on Operating Systems Principles (SOSP'07)*. ACM, New York, NY, 265–278. DOI:http://dx.doi.org/10.1145/1294261.1294287

Sergiu Nedevschi, Lucian Popa, Gianluca Iannaccone, Sylvia Ratnasamy, and David Wetherall. 2008. Reducing network energy consumption via sleeping and rate-adaptation. In *NSDI*. 323–336.

M. Nijim, A. Manzanares, Xiaojun Ruan, and Xiao Qin. 2009. HYBUD: An energy-efficient architecture for hybrid parallel disk systems. In *Proceedings of 18th International Conference on Computer Communications and Networks (ICCCN'09)*. 1–6. DOI:http://dx.doi.org/10.1109/ICCCN.2009.5235264

Daniel Nurmi, Rich Wolski, Chris Grzegorczyk, Graziano Obertelli, Sunil Soman, Lamia Youseff, and Dmitrii Zagorodnov. 2009. The eucalyptus open-source cloud-computing system. In *Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID'09)*. IEEE Computer Society, Washington, DC, 124–131. DOI:http://dx.doi.org/10.1109/CCGRID.2009.93

OpenStack. 2012. OpenStack - Open Source Software for Building Private and Public Clouds. Retrieved from http://www.openstack.org/.

Won Ho Park and C.-H. K. Yang. 2013. Effects of using advanced cooling systems on the overall power consumption of processors. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 21, 9 (2013), 1644–1654. DOI:http://dx.doi.org/10.1109/TVLSI.2012.2217386

Michael K. Patterson, Stephen W. Poole, Chung-Hsing Hsu, Don Maxwell, William Tschudi, Henry Coles, David J. Martinez, and Natalie Bates. 2013. TUE, a new energy-efficiency metric applied at ornls jaguar. In *Supercomputing*, JulianMartin Kunkel, Thomas Ludwig, and HansWerner Meuer (Eds.). Lecture Notes in Computer Science, Vol. 7905. Springer, Berlin, 372–382. DOI:http://dx.doi.org/10.1007/978-3-642-38750-0_28

Vinicius Petrucci, Orlando Loques, and Daniel Mossé. 2010. A dynamic optimization model for power and performance management of virtualized clusters. In *Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking (e-Energy'10)*. ACM, New York, NY, 225–233. DOI:http://dx.doi.org/10.1145/1791314.1791350

F. Pianese, P. Bosch, A. Duminuco, N. Janssens, T. Stathopoulos, and M. Steiner. 2010. Toward a cloud operating system. In *Proceedings of the IEEE/IFIP Network Operations and Management Symposium Workshops (NOMS Wksps)*. 335–342. DOI:http://dx.doi.org/10.1109/NOMSW.2010.5486552

Eduardo Pinheiro and Ricardo Bianchini. 2004. Energy conservation techniques for disk array-based servers. In *Proceedings of the 18th Annual International Conference on Supercomputing (ICS'04)*. ACM, New York, NY, 68–78. DOI:http://dx.doi.org/10.1145/1006209.1006220

M. Polverini, A. Cianfrani, Shaolei Ren, and A. V. Vasilakos. 2014. Thermal-aware scheduling of batch jobs in geographically distributed data centers. *IEEE Transactions on Cloud Computing* 2, 1 (Jan. 2014), 71–84. DOI:http://dx.doi.org/10.1109/TCC.2013.2295823

M. Powell, S.-H. Yang, B. Falsafi, K. Roy, and T. N. Vijaykumar. 2001. Reducing leakage in a high-performance deep-submicron instruction cache. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 9, 1 (2001), 77–89. DOI:http://dx.doi.org/10.1109/92.920821

Dang Minh Quan, Robert Basmadjian, Hermann de Meer, Ricardo Lent, Toktam Mahmoodi, Domenico Sannelli, Federico Mezza, Luigi Telesca, and Corenten Dupont. 2011. Energy efficient resource allocation strategy for cloud data centres. In *ISCIS*, Erol Gelenbe, Ricardo Lent, and Georgia Sakellari (Eds.). Springer, 133–141.

Praveen Raghavan, Andy Lambrechts, Javed Absar, Murali Jayapala, Francky Catthoor, and Diederik Verkest. 2008. Coffee: COmpiler framework for energy-aware exploration. In *High Performance Embedded Architectures and Compilers*, Per Stenstr, Michel Dubois, Manolis Katevenis, Rajiv Gupta, and Theo Ungerer (Eds.). Lecture Notes in Computer Science, Vol. 4917. Springer, Berlin, 193–208. DOI:http://dx.doi.org/10.1007/978-3-540-77560-7_14

Rouzbeh Razavi and Holger Claussen. 2012. Urban small cell deployments: Impact on the network energy consumption. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC'12)*.

RedHat. 2012. KVM - Kernel Based Virtual Machine. Retrieved from http://www.linux-kvm.org.

Xiaojun Ruan, A. Manzanares, Shu Yin, Ziliang Zong, and Xiao Qin. 2009a. Performance evaluation of energy-efficient parallel I/O systems with write buffer disks. In *Proceedings of the International Conference on Parallel Processing (ICPP'09)*. 164–171. DOI:http://dx.doi.org/10.1109/ICPP.2009.10

Xiaojun Ruan, Shu Yin, A. Manzanares, Jiong Xie, Zhiyang Ding, J. Majors, and Xiao Qin. 2009b. ECOS: An energy-efficient cluster storage system. In *Proceedings o the 2009 IEEE 28th International Performance Computing and Communications Conference (IPCCC'09)*. 79–86. DOI:http://dx.doi.org/10.1109/PCCC.2009.5403814

Eric Saxe. 2010. Power-efficient software. *Queue* 8, 1, Article 10 (Jan. 2010), 8 pages. DOI:http://dx.doi.org/10.1145/1698223.1698225

A. Scarfo. 2013. All silicon data center, the energy perspectives. In *Proceedings of the 2013 27th International Conference on Advanced Information Networking and Applications Workshops (WAINA'13)*. 1617–1622. DOI:http://dx.doi.org/10.1109/WAINA.2013.186

Y. Shiroishi, K. Fukuda, I. Tagawa, H. Iwasaki, S. Takenoiri, H. Tanaka, H. Mutoh, and N. Yoshikawa. 2009. Future options for HDD storage. *IEEE Transactions on Magnetics* 45, 10 (2009), 3816–3822. DOI:http://dx.doi.org/10.1109/TMAG.2009.2024879

Junaid Shuja, Sajjad A. Madani, Kashif Bilal, Khizar Hayat, Samee U. Khan, and Shahzad Sarwar. 2012. Energy-efficient data centers. *Computing* 94, 12 (2012), 973–994. DOI:http://dx.doi.org/10.1007/s00607-012-0211-2

J.-P. Smets-Solanes, C. Cerin, and R. Courteaud. 2011. SlapOS: A multi-purpose distributed cloud operating system based on an ERP billing model. In *Proceedings of the 2011 IEEE International Conference on Services Computing (SCC'11)*. 765–766. DOI:http://dx.doi.org/10.1109/SCC.2011.97

James W. Smith and Ian Sommerville. 2010. Green cloud: A literature review of energy-aware computing and cloud computing. Retrieved from http://blogs.cs.st-andrews.ac.uk/jws7/files/2010/10/LSCITS-Submission-Abstract-formatted.pdf.

G. Jeffrey Snyder, M. Soto, R. Alley, D. Koester, and B. Conner. 2006. Hot spot cooling using embedded thermoelectric coolers. In *Proceedings of the 2006 IEEE 22nd Annual IEEE Semiconductor Thermal Measurement and Management Symposium*. 135–143. DOI:http://dx.doi.org/10.1109/STHERM.2006.1625219

M. Steinder, I. Whalley, D. Carrera, I. Gaweda, and D. Chess. 2007. Server virtualization in autonomic management of heterogeneous workloads. In *Proceeding of the 10th IFIP/IEEE International Symposium on Integrated Network Management (IM'07)*. pp. 139–148. DOI:10.1109/INM.2007.374778

Jan Stoess, Christian Lang, and Frank Bellosa. 2007. Energy management for hypervisor-based virtual machines. In *Proceedings o the 2007 USENIX Annual Technical Conference on Proceedings of the USENIX Annual Technical Conference (ATC'07)*. USENIX Association, Berkeley, CA, Article 1, 14 pages.

K. T. Sundararajan, T. M. Jones, and N. Topham. 2011. Smart cache: A self adaptive cache architecture for energy efficiency. In *Proceedings of the 2011 International Conference on Embedded Computer Systems (SAMOS'11)*. 41–50. DOI:http://dx.doi.org/10.1109/SAMOS.2011.6045443

I. Takouna, W. Dawoud, and C. Meinel. 2011. Accurate mutlicore processor power models for power-aware resource management. In *Proceedings of the 2011 IEEE 9th International Conference on Dependable, Autonomic and Secure Computing (DASC'11)*. 419–426. DOI:http://dx.doi.org/10.1109/DASC.2011.85

S. Tavarageri and P. Sadayappan. 2013. A compiler analysis to determine useful cache size for energy efficiency. In *Proceedings of the 2013 IEEE 27th International Parallel and Distributed Processing Symposium Workshops PhD Forum (IPDPSW'13)*. 923–930. DOI:http://dx.doi.org/10.1109/IPDPSW.2013.268

M. E. Tolentino, J. Turner, and K. W. Cameron. 2009. Memory MISER: Improving main memory energy efficiency in servers. *IEEE Transactions on Computers* 58, 3 (2009), 336–350. DOI:http://dx.doi.org/10.1109/TC.2008.177

Thomas Treutner. 2012. *Energy Efficient Heterogeneous Virtualized Data Centers*. Ph.D. Dissertation. University of Vienna.

R. Uhlig, G. Neiger, D. Rodgers, A. L. Santoni, F. C. M. Martins, A. V. Anderson, S. M. Bennett, A. Kagi, F. H. Leung, and L. Smith. 2005. Intel virtualization technology. *Computer* 38, 5 (2005), 48–56. DOI:http://dx.doi.org/10.1109/MC.2005.163

Nedeljko Vasic, Prateek Bhurat, Dejan M. Novakovic, Marco Canini, Satyam Shekhar, and Dejan Kostic. 2011. Identifying and using energy-critical paths. In *CoNEXT*. 18.

Nedeljko Vasic and Dejan Kostic. 2010. Energy-aware traffic engineering. In *e-Energy*. 169–178.

Akshat Verma, Puneet Ahuja, and Anindya Neogi. 2008. pMapper: Power and migration cost aware application placement in virtualized systems. In *Proceedings of the 9th ACM/IFIP/USENIX International Conference on Middleware (Middleware'08)*. Springer-Verlag, New York, NY, 243–264.

Micha Vor dem Berge, Georges Da Costa, Mateusz Jarus, Ariel Oleksiak, Wojtek Pitek, and Eugen Volk. 2014. Modeling data center building blocks for energy-efficiency and thermal simulations. In *Proceedings of 2nd International Workshop on Energy-Efficient Data Centers, e-Energy 2013 Conference,*Berkeley, US, May 2013. 335–342.

Jun Wang, Xiaoyu Yao, and Huijun Zhu. 2008. Exploiting in-memory and on-disk redundancy to conserve energy in storage systems. *Computers, IEEE Transactions on* 57, 6 (2008), 733–747. http://dx.doi.org/10.1109/TC.2008.43

Lin Wang, Fa Zhang, J. Arjona Aroca, A. V. Vasilakos, Kai Zheng, Chenying Hou, Dan Li, and Zhiyong Liu. 2014a. GreenDCN: A general framework for achieving energy efficiency in data center networks. *Selected Areas in Communications, IEEE Journal on* 32, 1 (January 2014), 4–15.

Lin Wang, Fa Zhang, Athanasios V. Vasilakos, Chenying Hou, and Zhiyong Liu. 2014b. Joint virtual machine assignment and traffic engineering for green data center networks. *SIGMETRICS Performance Evaluation Review* 41, 3 (Jan. 2014), 107–112.

G. Warkozek, E. Drayer, V. Debusschere, and S. Bacha. 2012. A new approach to model energy consumption of servers in data centers. In *Industrial Technology (ICIT), 2012 IEEE International Conference on*. 211–216. http://dx.doi.org/10.1109/ICIT.2012.6209940

Wired. 2013. Google Throws Open Doors to Its Top-Secret Data Center. Retrieved from http://www.wired.com/wiredenterprise/2012/10/ff-inside-google-data-center/a ll/.

Michal Witkowski, Ariel Oleksiak, Tomasz Piontek, and Jan Weglarz. 2013. Practical power consumption estimation for real life HPC applications. *Future Generation Computer Systems* 29, 1 (Jan. 2013), 208–217. http://dx.doi.org/10.1016/j.future.2012.06.003

Di Xie, Ning Ding, Y. Charlie Hu, and Ramana Kompella. 2012. The only constant is change: Incorporating time-varying network reservations in data centers. In *Proceedings of the ACM SIGCOMM 2012 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM'12)*. ACM, New York, NY, 199–210. http://dx.doi.org/10.1145/2342356.2342397

Hong Yang and T. L. Marzetta. 2013. Performance of conjugate and zero-forcing beamforming in large-scale antenna systems. *IEEE Journal on Selected Areas in Communications* 31, 2 (2013), 172–179.

Emre Zer, Krisztin Flautner, Sachin Idgunji, Ali Saidi, Yiannakis Sazeides, Bushra Ahsan, Nikolas Ladas, Chrysostomos Nicopoulos, Isidoros Sideris, Babak Falsafi, Almutaz Adileh, Michael Ferdman, Pejman Lotfi-Kamran, Mika Kuulusa, Pol Marchal, and Nikolas Minas. 2010. EuroCloud: Energy-conscious 3D server-on-chip for green Cloud Services. In *Proceedings of the Workshop on Architectural Concerns in Large Datacenters in conjunction with ISCA-2010*.

Mingui Zhang, Cheng Yi, Bin Liu, and Beichuan Zhang. 2010. GreenTE: Power-aware traffic engineering. In *ICNP*. 21–30.

Yuting Zhang, Azer Bestavros, Mina Guirguis, Ibrahim Matta, and Richard West. 2005. Friendly virtual machines: Leveraging a feedback-control model for application adaptation. In *VEE*, Michael Hind and Jan Vitek (Eds.). ACM, 2–12.

Yi Zhao and Wenlong Huang. 2009. Adaptive distributed load balancing algorithm based on live migration of virtual machines in cloud. In *Proceedings of the 2009 5th International Joint Conference on INC, IMS and IDC (NCM'09)*. IEEE Computer Society, Washington, DC, 170–175.

Pin Zhou and Nagapramod Mandagere. 2012. *Harnessing Green IT: Principles and Practices*. Wiley Online Library, IEEE Computer Society, Chapter Green Data Storage.

A. Y. Zomaya and Y. C. Lee. 2012. *Energy Efficient Distributed Computing Systems*. Wiley.