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 a great significance comes with a 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 the infrastructure, an analysis of the current status is required.

In this paper we perform a comprehensive analysis of an infrastructure supporting the Cloud Computing paradigm with regards to the energy efficiency. First, we define a systematic approach for analyzing 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. Secondly, we utilize this approach for analyzing available scientific and industrial literature on state of the art practices in data centers and its equipment. Finally, we extract existing challenges and highlight future research directions.

Categories and Subject Descriptors: C.0 [**General**]: Systems architectures; K.6.2 [**Installation Management**]: Computer selection; Computing equipment management; Performance and usage measurement; K.6.3 [**Software Management**]: Software selection; Software maintenance; K.7.2 [**Organizations**]; C.2.5 [**Local and Wide-Area Networks**]: Internet; H.3.5 [**Online Information Services**]: Web-based services; Commercial services; C.5.5 [**Servers**]; H.3.4 [**Systems and Software**]: Distributed systems; Performance evaluation; C.1.0 [**General**]; C.2.1 [**Network Architecture and Design**]: Network communications; Network topology; Wireless communication; C.2.3 [**Network Operations**]: Network management; Network monitoring; D.4.1 [**Process Management**]: Scheduling; D.3.4 [**Processors**]: Compilers; K.1 [**The Computer Industry**]: Suppliers

General Terms: Management, Performance

Additional Key Words and Phrases: Cloud Computing, Energy Efficiency, Data Center, Network, Servers, Cloud Management System, Appliances

ACM Reference Format:
Toni Mastelic, Ariel Oleksiak, Holger Claussen, Ivona Brandic, Jean-Marc Pierson, and Athanasios V. Vasilakos, 2014. Cloud Computing: Survey on Energy Efficiency. *ACM Comput. Surv.* 0, 0, Article 0 (2014), 35 pages.
`DOI`:http://dx.doi.org/10.1145/0000000.0000000

Author's addresses: T. Mastelic and Ivona Brandic, Distributed Systems Group, Faculty of Informatics, Vienna University of Technology; A. Oleksiak, Applications Department, Poznan Supercomputing and Networking Center, and Institute of Computing Science, Poznan University of Technology; H. Claussen, Small Cells Research, Alcatel-Lucent, Bell Labs; J. Pierson, Computer Science, University of Toulouse; A. V. Vasilakos, Department of Computer and Telecommunications Engineering, University of Western Macedonia.

`DOI`:http://dx.doi.org/10.1145/0000000.0000000

1. INTRODUCTION

1.1. Motivation

New technological breakthroughs and massive production provide cheaper and easy-to-use products that are more accessible to a common person, which leads to a worldwide usage of emerging technologies. One of the main enablers of technological progress, and more generally our civilization today, is the energy that drives this machinery. However, due to its global usage the technological machinery creates an ever rising demand for more energy. Figure 1 shows electrical power consumption on a world scale. Looking only from 1990 until today power consumption doubled from 10k TWh up to 20k TWh worldwide [Enerdata 2012]. Future projections estimate almost 40k TWh until 2040, which makes it a 2.2 percent increase per year [EIA 2013].

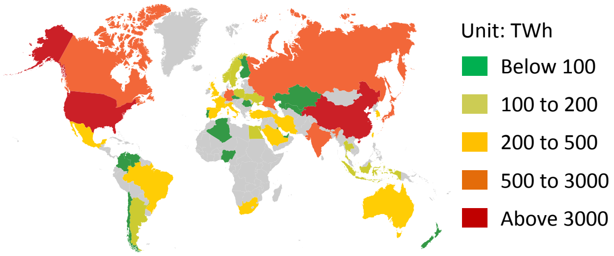

Fig. 1: World power consumption [Enerdata 2012].

In order to enhance sustainability of the energy supply and to reduce emissions of greenhouse gasses 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 resorts, 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 are spread over 300 up to 4500 square meters [Emerson 2008] hosting up to several thousands of server units. A typical 500 square meter data center can consume 27048 killowatt-hours (kWh) per day [Emerson 2009], which is more than the consumption of over 2500 households in 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 server installed base due to the 2008 financial crisis and the use of virtualization instead of hardware efficiency improvements [Koomey 2011]. However, even at the break of the crisis in 2008, 63% of data center managers claimed their actions to operate and expand will not be affected by an economic situation, and more than 80% of them had plans to renovate/expand existing (47%) or build a new data center (38%) [Emerson 2008]. Moreover, only 13% of them anticipated that their capacity will be sufficient beyond 2014 [Emerson 2008].

Perhaps the current situation is due to a still higher focus on high availability than on energy efficiency [Emerson 2008]. However, in their survey in 2010 Data Center Users' Group [Emerson 2010] identified heat density (cooling), energy efficiency and power density amongst five major data center concerns. Taking in account that a global annual data center construction size for 2020 is projected to $78 billion, which is almost twice than it was in 2010 [Belady 2011], stresses the importance of dealing with the energy efficiency and the environmental impact of the ICT.

1.2. The focus of the survey

In this survey we investigate energy efficiency of an infrastructure that powers the ICT machinery. As a representative of ICT technologies we take Cloud Computing, the leading and most promising ICT approach, which makes a large portion of the total ICT energy consumption for 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 as ICT equipment. In this survey we focus on energy efficiency of the ICT equipment separated into two domains, namely Server and Network domain. We also cover software solutions running on top of the ICT equipment, which include the Cloud Management System (CMS) domain for managing a Cloud infrastructure and Appliance domain that represents a software for servicing users.

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

1.3. The goal of the paper

The main goals of this survey are as follows:

- 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 the ICT equipment supporting the Cloud Computing concept.
- Discover and present correlations between different ICT domains with regard to the energy efficiency.
- Highlight existing research areas and future challenges.

Further in this document we describe our approach in Section 2, including the goals for improving the energy efficiency. Domains and their systems are described and analyzed in Sections 3, 4, 5 and 6 by providing a context to the energy efficiency goals, covering the state of the art and highlighting research 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, namely ”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 and community, as well as hybrid between the listed ones. Regardless to its deployment or a service model, Cloud Computing services are powered by large data centers comprised of numerous virtualized server instances, high-bandwidth networks, as well as supporting systems such as cooling and power supply. The listed equipment can be classified into two types as shown in Figure 2, namely hardware and software equipment [Hoelzle and Barroso 2013].

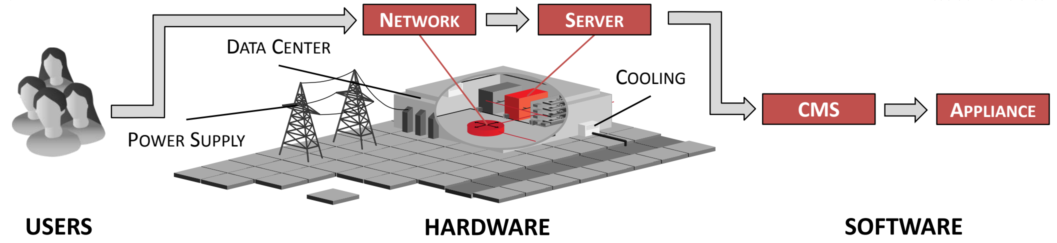

Fig. 2: Cloud Computing data center domains.

Hardware includes both ICT equipment and supporting equipment within a data center, as defined in [Avelar et al. 2012]. ICT equipment includes (1) *Network* and (2) *Server* domains as 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 as supporting equipment, and are covered only briefly in this survey. (1) Network and (2) 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, namely: (3) *Cloud Management System (CMS)* that is used for managing the entire data center, and (4) *Appliances*, which include software used by a user. (3) CMS and (4) Appliances are described and analyzed in Sections 5 and 6, respectively.

In this survey, 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 the analysis, we first define the terminology used in the context of energy efficiency. Furthermore, as 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 order to keep the structure of the paper.

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 for a data center equipment is extremely difficult [Fanara 2007], as it represents a complex system with large number of components from various research areas such as computing, networking, management, etc. The provided service of such a system is too diverse to be covered with all the details.

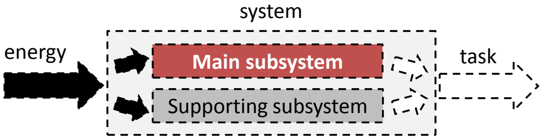

Fig. 3: A system and (sub)systems.

On the one hand, surveys such as [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 difference between energy used by the ICT equipment 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 the energy efficiency from a more general perspective.

Figure 3 shows an arbitrary system as a set of interconnected components, where 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 certain task, where a task is an abstract assignment that the system has to perform in order to fulfill its purpose. In order to improve the energy efficiency of a system, first it is necessary to identify problems degrading the efficiency.

Therefore, we identify two critical points where energy is not used in an efficient way, but instead it is *lost* or *wasted*. Both terms define inefficient use of energy from an agnostic point of view, where *energy loss* refers to an energy brought to the system, but not consumed for its main task, e.g., energy lost due to a transport and conversion. This also includes energy used by supporting subsystems, such as cooling or lighting within a data center, where the main task is the provision of Cloud services. *Energy waste* refers to an energy used by the system's main task, but without an useful output, e.g., energy used while running in an idle mode. Additionally, useless work by the system is also considered as energy waste, e.g., for a cooling subsystem, this would mean keeping the cooling at maximum during the night when the temperatures are lower. Both critical points are shown on Figure 4.

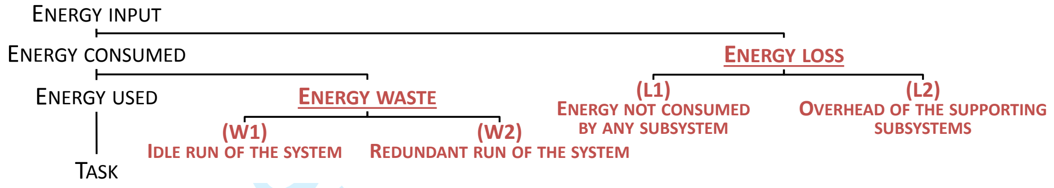

Fig. 4: Critical points within a system where energy is *lost* or *wasted*.

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, e.g., implementing a single cooling unit for the entire cabinet, instead of cooling each rack server separately.
- (W1) The third goal is to reduce an **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 the system output, e.g., for providing as twice as much bandwidth, a network router requires twice the amount of energy or less.
- (W2) The fourth goal is to minimize the energy consumption where the **system performs redundant operations**. This can be done by implementing smart functions and subsystems, e.g., implementing optimized algorithm, which does not require redundant steps in order to perform the same task.

The listed goals are taken as a basis for the literature review in order to find current as well as future research directions, which focus on improving energy efficiency of Cloud Computing infrastructure. Figure 5 shows data center domains and their energy cascades as they are covered in this paper, starting from the Network and Server domains to the CMS and Appliance.

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

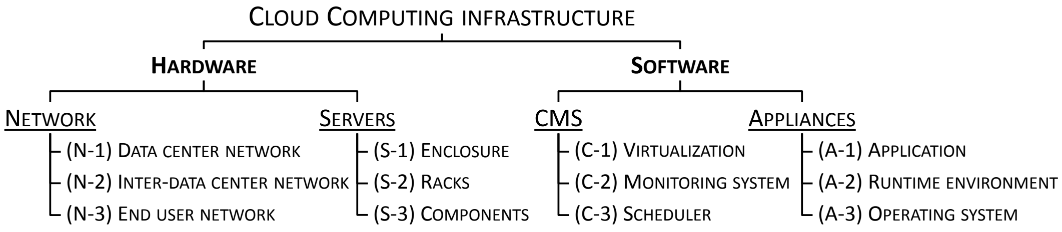

Fig. 5: Cloud computing infrastructure domains and related systems.

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 the network traffic within this period [Kilper et al. 2011].

3.1. Context

The energy consumption of the Network domain consists of three main systems. First, the connections inside of a data center. Second, the fixed network between data centers. Finally, the end user network that increasingly provides the wireless last hop to end-users, which access the services via smart phones, tablets, and laptops. Based on this breakdown, each system brings its own energy wastes and losses as shown in Figure 6.

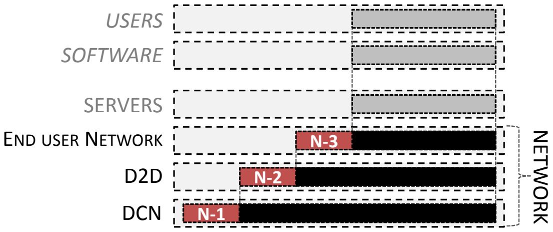

Fig. 6: Losses and wastes of network domain

- **DCN - Data center network (N-1):** Within a data center, the energy consumption of the network currently accounts for up to 5 percent of the total energy consumption of the data center [Fanara 2007]. As shown by [Abts et al. 2010] the network power accounts for approximately 20% of the total power when the servers are utilized at 100%. However, it goes up to 50% when the utilization of servers decreases to 15%. This is due 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. Moreover, poor network architectures not suited for Cloud applications can increase energy waste by unnecessary re-routing of the traffic or keeping some parts of 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 they still need to communicate between each other. Another application of these networks is when applications or data are migrated between data centers dependent on the time of day, to minimize delay, energy consumption, or costs. As observed by [Wang et al. 2014b] this communication includes background, non-interactive, 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 kind of network traffic, the 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 for detection and error correction [Feeney and Nilsson 2001], which all represent energy consumption overhead created by supporting tasks rather then the main task, i.e., data delivery.

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

- **L1.** Reducing a heat load of a network equipment inside of a data center (N-1) would reduce its energy consumption and a consumption of its cooling subsystem as well. This can be achieved by adapting a design of a network equipment as suggested by [ASHRAE 2012], by implementing front to rear air flow. This would also increase its reliability by a factor of 2.
- **L2.** Goal (L1) also brings benefits to the goal (L2), i.e., by reducing heat load a smaller cooling subsystem can be installed, which consumes less energy. Although it comprises a basic network equipment, failure handling supported by redundant equipment can also be considered as a subsystem since it does not perform the main task of the system. Therefore, moving away from traditional 2N tree topology towards 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, where 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 adaption [Lopez-Perez et al. 2014] [Gunaratne et al. 2008] or simply turning off unused ports, links and switches inside of a data center (N-1) [Heller et al. 2010] would reduce idle energy consumption and therefore achieve this goal. Except tweaking only 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 idle time of the network by using unutilized network bandwidth for bulk transfers between data centers, or exploiting benefits of different data rates as proposed in [Mahimkar et al. 2011].
- **W2.** Achieving this goal depends mostly on how a network is used by the servers, as well as Software and finally the User domain. However, some optimization can still be done by observing communication patterns and reducing unnecessary traffic. Such 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 of the prominent ones are the Bell Labs led GreenTouch$^{\text{TM}}$consortium, and the EU funded projects EARTH and ECONET.

GreenTouch$^{\text{TM}}$[2012] is a consortium of over 50 leading Information and Communications Technology (ICT) organizations, both industry and academic, dedicated to reducing carbon footprint of the communication networks. The goal of GreenTouch is to identify key components that can increase network energy efficiency by a factor of 1000 by year 2015 compared to 2010 levels. This goal will be achieved by delivering the architecture, specifications and technologies.

EARTH [2011] is a European Union funded IP research project in FP7 focused on mobile communication systems and their energy efficiency. The target of the project is a reduction of the energy consumption by at least 50% focussing 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 European Union 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 focussed on the network architectures, scaling of energy consumption with load, and low complexity processing.

Wireless network architectures. From a wireless architecture perspective, moving from traditional macrocellular networks to a HetNet architecture is one of the most impactful changes with high potential for reducing the energy consumption [Claussen et al. 2009]. In macrocellular networks, due to the fact that the energy is transmitted in a relatively un-focussed 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 where the transmit power is not the limiting factor any more. In [Razavi and Claussen 2012], the authors have shown that for an urban area, the 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 the cable losses and improve efficiency in the order of 3 dB. Furthermore. increasing the number of antennas at the base station to “large scale antenna systems” in combination with beamforming, which focusses the energy to the user and also reduces interference can significantly improve the energy efficiency [Yang and Marzetta 2013].

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 the 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 are their power amplifiers, which are currently relatively inefficient and even consume a large amount of power when the cell is only lightly loaded. One approach for addressing this problem is presented in [Grebennikov and Bulja 2012] using multi-stage Doherty power amplifiers. A further important area are idle modes, that allow network equipment to be switched off while not required and 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 where 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.

Low complexity processing in wireless networks. Finally, processing for wireless communications becomes more complex to maximize the capacity within the 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 have shown that receiver complexity can be reduced significantly with the concept of spatial modulation. Examples of earlier work have focused on low complexity detection algorithms [Hochwald and Ten Brink 2003], and 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.

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 I: Research areas in wireless networks and relevant literature.

The topic of reducing energy consumption in fixed access network has been well-studied. 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 in [Bolla et al. 2011] and [Bianzino et al. 2012]. Additional surveys include [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.

Scaling of energy consumption with load in wired networks. The first attention has been focused on designing energy-aware network devices in which the power consumption is manageable according to the traffic load. Among them, sleeping and rate adaptation are two representative approaches. In [Gupta and Singh 2007b] the authors proposed to take 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.

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 intra-domain traffic engineering mechanism”, GreenTE, which is able to guarantee given performance requirements while a maximum number of links is 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 adaption. They then presented Energy-Aware Traffic engineering (EATe), achieving the same traffic rates while reducing the energy consumption of the network by spreading load among multiple paths. Then, in [Vasic et al. 2011] they propose a REsPoNse framework, where 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 trade-off problem. Recently, [Cianfrani et al. 2012] proposed power-aware OSPF routing protocols, which aims at providing routing services with the minimum number of links by modifying Dijkstra’s algorithm and sharing the shortest path trees of under-utilized routers. However, quality of service is recognized as a trade-off.

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]

Table II: Research areas in wired networks and relevant literature.

3.3. Challenges and Research directions

For wireless networks, the need for increased capacity is leading to a trend towards heterogeneous network architectures where macrocells provide area coverage, but most of the capacity is provided by small cells. In addition to providing high capacity, such architecture have a high potential of 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 focussed 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 towards smaller cells, the 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 between each other. From the local perspective, it is fundamental to design efficient network equipments with multiple energy saving modes, called 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 response 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 the perceived 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 in real networks is still an open problem. Among the many issues, the scalability problem appears as the most important one. As real networks are usually of large scale, it is required that the designed protocol should be distributed and can scale out easily. At the same time, it is still open how to trade off between energy saving and QoS, ensuring network stability while achieving energy conservation. Both remain as important research topics that need further attention.

4. SERVERS

The Server domain includes computing and storage servers [Warkozek et al. 2012], as well as its components such as processor, memory, cabinets, etc., except communication equipment that are part of the Network domain. It also considers aspects such as component layout within a rack and component architecture. As the second domain that belongs to an 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 20 kW [Fanara 2007], which is equal to the average power of 35 households in Austria during one year [Bittermann and Gollner 2011]. Considering that only in the U.S. a total number of installed servers tripled from 2000 to 2010 as estimated by EPA [Fanara 2007], and thus reaching a number of over 15.8 million, improving energy efficiency of the servers represents a top priority tasks in the IT industry.

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 consumption of all supporting hardware equipment as they all represent an energy loss. However, energy loss and waste do not stop there, since servers can also contribute to energy waste due to a poor usage policy of the server equipment, as well as energy loss due to a poor energy supply and internal subsystems. Systems of the Server domain include server enclosure, such as server cabinets. Server racks represent another system, and finally components within a rack, such as CPU, memory, hard-disk, etc., are the third system.

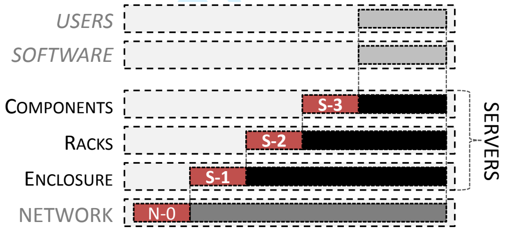

Fig. 7: Losses and wastes of server domain

- **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 CRAC units (computer room air conditioners), requires from 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 heavy loaded servers as for 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. The most common way to do this is the use of cold plates [Coolit 2013] or microchannels [IBM 2013]. Recently, other approaches based on 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), however, have an impact on the 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 peek 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 the 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 card, are contributing to an energy loss. Another source of energy loss are fans which have typical efficiency around 60%, i.e. around 40% of power is lost due to heat dissipation. Additionally, if the fan speed is not well adjusted to the server load and temperature a significant part of energy is wasted.
- **Components (S-3):** Energy efficiency of server components drastically affects the overall efficiency of a server. Focus is given on components that take a bigger slice of the total energy consumption such as the CPU, which can consume more than a third of the total server energy consumption [Fan et al. 2007]. A typical TDP (thermal design power) of today’s processors can fall in the range from 80 to 103 Watts, or 91 W in 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 26 watts. On the other hand, when the VM with the same performance ran on 2 physical cores, each being 50% utilized, it consumed only 17 watts. However, servers with large number of slower CPU cores can lead to lower utilization, i.e., a bin-packing problem where smaller bins cause a bigger bin-packing problem [Mudge and Holzle 2010].

  Except underutilized CPU cores that effect 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 that in modern processors accounts for 20-40% of the total power consumption [Kaxiras and Martonosi 2008]. Memory also creates energy overheads since it is built to provide high performance in order to fulfill ever growing CPU demands, thus growing in density, functionality and scalability [Tolentino et al. 2009]. This resulted in neglecting the energy efficiency of the memory subsystem. Finally, the disk system has proven to be another power drainer 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].

In order to mitigate the previously described 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 fulfills this goal. This can be achieved by using more energy efficient components and their architectures, e.g., using slower, so called wimpy CPU cores that are more power efficient [Mudge and Holzle 2010] such as in FAWN project [Andersen et al. 2009] where they utilize wimpy cores for building 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 case of the memory subsystem, adjusting its performance, i.e., throughput is used for mitigating high temperatures, and thus avoiding energy loss through heat [Lin et al. 2007]. Energy loss can also be reduced by using compact server configurations which excludes components that are not used, e.g., Google uses such an approach for building its data centers.
- **L2.** Following the goal L1, the goal L2 provides additional energy savings by reducing energy consumed by supporting systems such as cooling and power supply inside server enclosure (S-3) and servers themselves (S-2). For example, Google places backup batteries next to racks, therefore avoiding large UPS units that require its own cooling system [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 a current load would move towards achieving this goal, e.g., using DVFS capable CPUs that provide different P-states (power modes while being utilized), as well as sleep C-states (power modes while being idle). The authors in [Dharwar et al. 2012] provide an overview of these techniques along side with power capping. The same applies for other components such as memory and storage disks, which can be put to a low power state while being idle. However, this is beneficial only when there are frequent idle periods. In the contrary this can create even bigger power consumption overheads due to a spin-up in case of a storage disk [Wang et al. 2008]. However, using DRPM [Gurumurthi et al. 2003] with dynamic spin speed can perhaps represent 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 a right processor architecture can also contribute to a more efficient energy usage. Due to the nature of applications running in a Cloud (e.g., web search, video hosting and MapReduce), emphasis is given 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 multi-threaded 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, for developing middlewares for a transparent workload parallelization.
- **W2.** As shown by [Tavarageri and Sadayappan 2013] bigger cache size does not necessarily mean lower miss-rate. Therefore, choosing a right size cache can decrease energy waste and achieve this goal. Additionally, using cache subsystem for storage

disks in order to reduce reads and writes from/to the disk and increase its idle time, can also contribute in energy savings [Wang et al. 2008]. Such onboard controller cache 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 that are out of scope of this survey. Although there is some research work in this field, most of the innovations come from the industry and production environments as best practices. Some of the important literature includes the book by [Hoelzle and Barroso 2013] covering Google's practices inside data centers, while [Greenberg et al. 2006] provides best practices learned from benchmarking 22 data centers.

Server cooling. While 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, in specific Embedded thermoelectric cooling (eTEC), can reduce the temperature of localized hot spots generated by modern processors and therefore reduce its power consumption. The authors in [Park and Yang 2013] compare eTEC with vapor compression refrigeration system for cooling microprocessors. They show how eTEC can achieve 3% up to 10% of power savings, and with vapor 25%. The general conclusion is that by using localized cooling of the right component can give some worthwhile improvements. Additionally, the authors in [Haywood et al. 2012] suggest using heat generated by the 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 subsystem. They also present JETC, a management system for server memory and CPUs with combined energy thermal and cooling solution. By applying such an approach they consider dependencies between CPU and memory, as well as their shared cooling subsystem, and finally achieve 50.7% average energy reduction.

Processor architecture and design. Combining different types of server components has proven to be promising when it comes to applying energy saving schemes. As part of the EuroCloud project the authors in [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 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 to reduce space, resources and power. Generally, solutions based on a high number of densely packed low power processors, so-called micro-servers, are one of trends visible recently on the market. In addition to the physical prototype development CoolEmAll also proposes blueprints defining 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. In [Dreslinski et al. 2009] the authors propose a cluster architecture with multicore processors. The idea is to use the same processors for single-threaded and multi-threaded operations, where a processor with four cores can be reconfigured to run only one overclocked core using a power from those three that are turned off. Finally, the article [Mudge and Holzle 2010] gives an overview of challenges for choosing and building energy efficiency processors for Cloud infrastructures.

DVFS and alternatives. One of the most notable techniques for reducing energy consumption in a CPU is basically reducing its power input due to its disproportional energy consumption, 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 the ones 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], while the authors in [Kahng et al. 2013] propose some improvements over the DVFS itself. Going even deeper into power limitation, authors in [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] use co-scheduling for resource allocation in order to maximize the performance under power and performance constraints. It identifies non scalable VMs in terms of performance and consolidates them together. This is an example of how CMS domain, i.e., VM management, in combination with the Server domain can provide more benefits for energy saving techniques.

Cache management. Turning off parts of cache in order to reduce static power consumption (also known as leakage power) is proposed in [Powell et al. 2001], and [Kim et al. 2013]. Instead of simply turning off parts of cache that are not used, the authors in [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 of the cache parts that are being turned off, the authors in [de Langen and Juurlink 2009] propose 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], which allows reconfiguration of both size and associativity, i.e., dynamically changing hierarchy as a program runs. Except improving management over existing cache subsystems, using novel cache architectures and technologies can cut energy loss at the start. The authors in [Dreslinski et al. 2008] suggest using near threshold cache architectures in order to reduce energy consumption. Additionally, they combine it with traditional cache in order to maintain performance.

Storage systems. Except optimizing cache subsystem itself, the cache can be used for achieving energy efficiency goals for other subsystem such as storage disks. The authors in [Chen et al. 2012b] exploit the catching scheme to improve energy efficiency of RAID disk systems. Along with the energy efficiency, the authors in [Felter et al. 2011] also consider disk reliability. 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, the authors in [Ge et al. 2011] present DiscPOP, a power-aware buffer management which 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 for disk buffers approach as shown in [Manzanares et al. 2008]. A step further in using buffers is suggested by authors in [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 for storing most popular data, thus providing fast access to this data without affecting disks. The authors in [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 a field of storage disk energy efficiency. Moreover, SSD disk and their utilization for

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 III: Research areas in server domain and relevant literature.

energy efficient storage systems is discussed in [Scarfo 2013], while HDD technology is discussed in [Shiroishi et al. 2009].

4.3. Challenges and Research directions

Utilizing low power modes for server components has proven to be 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 are usually performing some light tasks. Therefore, the goal is to achieve self-scalability of server components, both on hardware and software level. This includes energy consumption increase/decrease proportional to the provided performance. This can be achieved by utilizing techniques such as DVFS, which has become a common feature of a modern processor. Another goal is to proportionably scale available resources with a power consumption, i.e., consolidating underutilized components and achieving a zero power consumption of the 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 is identified as the main concern within a data center facility according to [Emerson 2010]. Thus, the Cloud Management System (CMS) plays an important role when trying to improve the efficiency, increase utilization, and thus lower the energy loss/waste within a data center.

5.1. Context

The CMS domain includes 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 is recognized as its main purpose, as its main function is to deploy resources for fulfilling customer requests. Its supporting task is a monitoring system that provides additional information about the allocated and available resources such as utilization, QoS, etc. Additionally, virtualization technology is used for better resource

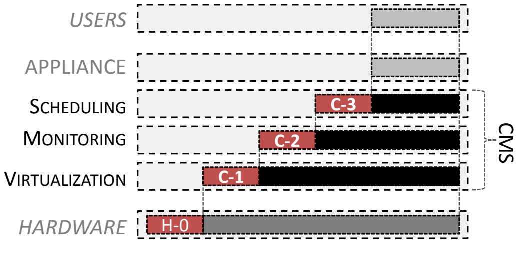

Fig. 8: Losses and wastes of CMS domain

management and on-demand deployment offering a high scalability of a Cloud infrastructure.

Based on this, the energy efficiency of the CMS can be examined through its above mentioned systems, which include both energy loss and waste as shown in Figure 8.

- **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%), such as Xen [Citrix 2012], as opposed to using physical machines. Additionally, too big 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 overhead, which is considered as 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 just for running 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 where memory is cluttered with a huge amount of data that is not being used.
- **Scheduler (C-3):** Managing Cloud resources should not be overdone, e.g., rescheduling VMs every couple of minutes would perhaps give optimal deployment at the moment, however the re-scheduling itself would probably consume more energy than it saves. Furthermore, using migrations can lead to a performance overhead, as well as the energy overhead. While 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 the above listed issues, H-0 in Figure 8 represents energy delivered to a hardware equipment that was not fully utilized by a 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, e.g., by consolidating underutilized machines and turning of the idle ones [Feller et al. 2012].

In order to reduce above listed 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, since the CMS has a "knowledge" of which resources are required and what not, e.g., shutting down idle machines [Borgetto et al. 2012b]. The CMS can go beyond controlling only the servers, but expand its control to the Network system, or even Cooling and Power Supply systems [Lago et al. 2011]. This way, energy delivered to a 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 number 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. paravirtualization, or even micro-kernel 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, the CMS subsystems should be implemented in a modular fashion, where 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 an idle mode, e.g., using lightweight hypervisors.
- **W2**. Energy waste of the CMS system (C-3) can be avoided by optimizing the scheduler and measuring not only its results, but also its trade-offs 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 a technology used for its implementation.

5.2. State of the art

Several research papers focus on different levels of potential actions at the Cloud Management System level to mitigate energy savings. We can distinguish 4 levels of actions. First a VM can be reconfigured in order to change its resource requirements. This way, the stress on the system is lower and the energy consumed reduced. Furthermore, the physical machines themselves can be adjusted to their actual load so as to reduce their power consumption. Second, the placement of VM 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 so as to reduce the resource consumption at any given period of time. All these actions can be combined and often several levers are used in the same framework in the following literature. Also they must take into account the potential degradation of quality of service induced. The approaches differ in the kind of constraints they put on the QoS. The PhD dissertations [Borgetto 2013], [Feller 2012] and [Treutner 2012] or surveys like [Beloglazov et al. 2011] are primary sources of literature reviews, among others.

VM reconfiguration. Considering the first possibility, VM reconfiguration and hardware adjustment, [Zhang et al. 2005] propose virtual machines that self-adapt their resource allocation to their demands. Similarly, [Borgetto et al. 2012c] propose VM reconfiguration, where the middleware adapts the VM resources' demands to their needs. The authors propose pro-active VM reconfiguration models taking also into account the time needed to change the state of the physical machines (power on to off, and vice-versa). [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 associate VMs with software CPU power state, as compared to the hypervisors conventional power states of a CPU. This allows hardware and software to be coordinated to use the best power mode, using 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.

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. 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 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, several sort 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 most energy-efficient hosts, while [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 Cloud Management Systems can easily be adapted. Jobs are handled in a cluster infrastructure, but seeing these as VM 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 propose a set of constraints for energy reduction while allowing task migration. Similarly [Borgetto et al. 2012a] use a linear program modeling taking into account some SLA for jobs and 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 the maximum resource ratio it might need to each job, 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.

VM migration and consolidation. The third possibility is investigating VM (live) migration combined with physical machines consolidation. [Liu et al. 2011a] have studied live migration of virtual machines in order to model the performance and energy of the 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 virtual machine 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 the over provisioned 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 using also 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

Research area	**References**
Energy efficiency in general	[Borgetto 2013], Feller [Feller 2012] [Treutner 2012] [Beloglazov et al. 2011]
VM Reconfiguration and hardware management	[Zhang et al. 2005] [Borgetto et al. 2012c] [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. 2008] [Beloglazov et al. 2012] [Berral et al. 2010] [Polverini et al. 2014]

Table IV: Research areas in CMS domain and relevant literature.

migration overhead. In the context of the Fit4Green project, the authors in [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 for instance 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.

VM scheduling. Finally, smart VM scheduling is also a source of energy savings. [Burge et al. 2007] handle the request scheduling in heterogeneous data center scenario. They focus on the decision 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, cancelation costs, etc. Their conclusion is that even using very simple heuristics, e.g., shutting down a server that has been idle for the last minutes, can save a significant amount of energy. [Steinder et al. 2008] have investigated 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 be able to account for each virtual machine a precise energy consumption. In today's CMS, this is reduced to a simple calculation based on the number of hosted virtual machines on one host. Since each application will require different resources (some may use more CPU, memory, disk or network resources), the share for each virtual machine must be mathematically and precisely modeled.

Second, the interdependencies between possible leverages in the CMS must be further investigated: Mixing for instance an initial allocation of virtual machines to physical hosts with post-adjustment of these hosts using DVFS can be counter-productive and sub-optimal. 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. It means that on a perfect Cloud infrastructure only Appliances would be consuming resources and thus energy. From a provider's perspective, efficiency of appliances is only considered for Software as a Service and a Platform as a Service, since 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., Infrastructure as a Service), an appliance is deployed by a user, thus the user is responsible for its efficiency. This scenario falls under the User domain perspective. To date, software designers were usually looking 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

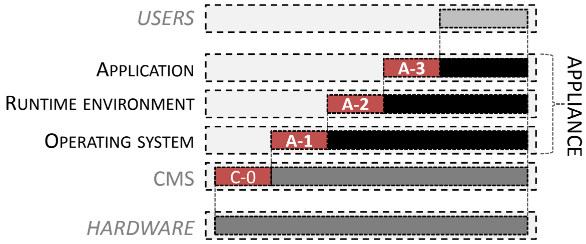

Fig. 9: Losses and wastes of appliance domain

The Appliance has a relatively smaller impact on the overall energy consumption than some other elements of 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 energy efficiency of the Cloud, appliances must be taken into consideration. Three subsystems can be recognized for the appliance. They include an application that is used by the end user and which performs a main task of the appliance, a runtime environment required for running the application, and finally an operating system, which serves as a bridge between physical or virtual machine and the software running on top of it.

Energy-efficiency of appliances affects both energy loss and waste according to the model presented in Section 2, and are shown in Figure 9.

— **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 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 of all, energy is usually consumed by additional components of the application. These modules are integral parts of the application but they are responsible for its non-functional aspects such as security, reliability, control, logging, etc. When the appliance is not strongly utilized by end users these modules can be source of energy consumption which is not related to any useful work. Energy is also consumed by supporting subsystems which are responsible for the maximization of the appliance utilization and for dynamic adaptation of the appliance to load (according to goal W2). These subsystems are needed to optimize the appliance efficiency (e.g. by stopping some services in the case of low load) but energy used by them is a waste as 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, allocated data structures in memory and on hard disc without producing any output. For example, if a lower number of Web servers would be sufficient for end users so the additional servers waste energy.

- **Runtime environment (A-2):** Applications usually need a runtime environment to be executed, from java virtual machines or software interpreting script languages such as Python, through web servers such as Apache or Tomcat, through more complex systems. From the perspective of the appliance main task, energy consumed by the runtime environment is a loss, which should be minimized (goal L2). Loss can be also caused by overheads of runtime environments, e.g. programming languages or lack of optimization for given application type and hardware.
- **Operating system (A-3):** Both Application and runtime environment 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 the 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 the above listed issues from the Appliance perspective, energy spent for running the CMS (C-0) is entirely considered as lost since it performs supporting tasks, rather than the main task of the appliance.

A number of actions can be taken to reduce the energy losses and wastes presented above, 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 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 losses of energy 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) trying to reduce resource consumption while performing the same task. Applied techniques can focus on smart management of an appliance 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, etc. Otherwise, use of these resources should be treated as energy waste as they cause higher power usage without 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 Service Level Agreements 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 he number of service instances or threads, adjusting frequencies of monitoring polling, caching and indexing. Overheads can be also related to relevant functionality of appliances, e.g. security and resilience. Hence, applying most of these techniques requires finding a trade-off between energy-efficiency and other key aspect of appliances such as performance, resilience and security.

6.2. State of the art

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 Cloud infrastructure, especially hardware, virtualization technology, and Cloud Management Systems.

Design and development. [Agrawal and Sabharwal 2012] cover many issues related to 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 have also shown 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 the system should run in a lower state and the 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 paper. 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 paper. Third, if possible, software requests to access additional resources should be done infrequently and in batches, decreasing number of unnecessary wake-ups. Additionally, as indicated in [Smith and Sommerville 2010], attention should be paid to some 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].

Compilers. In [Fakhar et al. 2012], the authors 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 re-ordering, loop optimization, etc. 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 into account energy-efficiency are the encc [Falk and Lokuciejewski 2010] or Coffee [Raghavan et al. 2008] compilers, however they are not focusing on software development for Clouds.

Application monitoring. To improve energy-efficiency of appliances, their power usage must be monitored. Identifying consumption of particular applications is a non-trivial problem. However, there have been attempts to do this. For examples, PowerTOP is a utility created by Intel that monitors a system and reports, which processes are responsible for wakeups that prevent a CPU from entering a sleep state to the user. 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 presented in [Witkowski et al. 2013]. These solutions additionally allow to identify which combination of application classes and hardware configurations are the most efficient. They focus more on High Performance Computing (HPC) applications, however, this is consistent with one of the current hot topics which is HPC in the Cloud and moving scientific applications to the Cloud. Additionally, a similar methodology could be applied to Cloud applications. Some attempts to this were done in projects such as Hemera [2013] and Magellan [2013].

In [Beik 2012] the author proposes an energy-aware software layer for more efficient energy usage. It collects micro and macro metrics in order to efficiently use and deploy shared services in a shared Cloud infrastructure.

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]. In this paper, the authors indicate that an event based architecture where 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 the 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.

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]

Table V: Research areas in appliance domain and relevant literature.

6.3. Challenges and Research directions

The main challenges related to energy-efficiency of Cloud appliances include an appropriate appliance development process, minimizing overheads of appliances, 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 the 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 could optimize code for an energy-efficient mode. In addition to automated compiler optimizations energy-efficient design patterns should be defined for developers. They 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 as 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, which are 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 virtual machines or providing sand boxes for applications instead of virtual machines with the whole operating systems.

Proper assignment of appliances and hardware resources requires further investigation and 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, e.g. taking into account processing progress, appliance load, performance, state, data size, etc. These decisions may include migration of appliances, adjusting appliance size (e.g. VM size), defining the mode to be set, etc. To this end metrics that define the appliances productivity and energy efficiency must be defined and measured which is an additional challenge.

7. INTERACTIONS BETWEEN CLOUD COMPUTING INFRASTRUCTURE DOMAINS

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

Appliance. Starting with the Appliance, it is a 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 basically all other domains, in particular, 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 the GPGPU advantages. Similarly, some of the appliances can be run on microservers equipped with processors such as ARMs, but not all of them 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 the management of the application placement and scheduling, must take these facts into consideration since its influence on other domains can be large. For instance, the local temperature, hence 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 now 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 the underlying layers itself and their interaction with a certain system.

Servers. In order to support smart scheduling and hardware matching, and finally making an optimal decision, the CMS needs detailed information about appliances and underlying hardware. This information includes progress, performance, state, 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 the ones 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 the CPU power consumption with a circuit proposed in [Borovyi et al. 2009]. Modeling a VM power consumption is a step further in order to obtain 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 the power consumption for computing and cooling in a data center, the power consumption for the network transport is still relatively small. As a result this enables placement of computation at data centers where for example energy from renewable resources is available or less energy for cooling is required due to a cool climate. This flexibility of placement of computation, enabled by efficient high bandwidth network connections can result in a significant reduction of the energy consumption for computation. However, when placing computation far away from the end user, this also results in an increased latency limited by the speed of light in the optical fibre. Additionally, when using migration, the impact on the network can not be completely ignored. Indeed, even if several researchers suggest that the impact of the traffic can be ignored in terms of power consumption (i.e. the switches and routers consumes roughly the same amount of energy, whatever the bytes transferred [Hlavacs et al. 2009]), it can not be so when considering also that network components can be switched off or bandwidth adapted like in ALR (Adaptive Link Rate) for saving energy when not being used. Using models for power consumption during migrations such as [Liu et al. 2011b] can add to overall power consumption awareness for using such optimizations.

Cooling and power supply. Concerning the domains such as cooling, power supply and data center building that are only briefly covered in this paper, surveys such as [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 related to the 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 was defined in [Avelar et al. 2012]. However, in this paper we focus on the ICT equipment optimization, where PUE is not sufficient to represent such level of details being on the level of a data center. A metric such as Data Center infrastructure Effectiveness (DCiE) [Belady 2008] shows only the inverse of the PUE, hence inheriting the same shortages. For example, PUE Scalability measures the power proportionality - how the used power scales with the 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) among others, and are covered in surveys [Kulseitova and Fong 2013] and [Cavdar and Alagoz 2012].

8. CONCLUSION

In this paper we analyzed the energy efficiency of a data center ICT equipment, including hardware and software that drives the Cloud Computing. First, we described our approach that can be applied to an arbitrary system composed of smaller components/subsystems. Second, we introduced a breakdown of the Cloud Computing infrastructure by including hardware and software equipment located in a data center. Third, we used a systematic approach for analyzing energy efficiency of the ICT equipment and the software running on top of it, by going through available literature. This way, we provided a holistic and uniform overview of the data center ICT equipment with regards to the energy efficiency.

Our analysis showed that many of the standard energy efficiency techniques do not work for Cloud Computing environments simply out of the box, rather they have to be at least adapted, or even designed from the scratch. This is due to a stratification of the Cloud Computing infrastructure, which comprises systems and components from different research areas, such as power supply, cooling, computing, etc. Optimizing these systems separately does improve the energy efficiency of the entire system. However, applying shared energy efficiency techniques on multiple systems or their components can significantly improve the 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 Comput. Archit. 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. 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 Comput. Commun. Rev.* 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, USA, 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 *Intelligent Computer Communication and Processing (ICCP), 2011 IEEE International Conference on*. 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 *Workload Characterization, 2008. IISWC 2008. IEEE International Symposium on*. 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 *Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th 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. http://www.thegreengrid.org/. (2012).

R. Ayoub, R. Nath, and T. Rosing. 2012. JETC: Joint energy thermal and cooling management for memory and CPU subsystems in servers. In *High Performance Computer Architecture (HPCA), 2012 IEEE 18th International Symposium on*. 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 *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 First International Conference on Self-organizing Architectures (SOAR'09)*. Springer-Verlag, Berlin, Heidelberg, 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. *Communications Surveys Tutorials, IEEE* 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 *IEEE SoutheastCon 2010 (SoutheastCon), Proceedings of the*. 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 *Engineering and Technology (S-CET), 2012 Spring Congress on*. 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. http://www.thegreengrid.org/. (2008).

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 Gener. Comput. Syst.* 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, USA, 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 Heidelberg, 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, USA, 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 *Green Computing Conference (IGCC), 2012 International*. 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 Gener. Comput. Syst.* 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. 2012b. 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, USA, Article 25, 10 pages. DOI: http://dx.doi.org/10.1145/2208828.2208853

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, USA, 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 *Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 2009. IDAACS 2009. IEEE International Workshop on*. 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 Comput. Surv.* 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, USA, 327–334.

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

Feng Chen and Xiaodong Zhang. 2008. Caching for bursts (C-Burst): let hard disks sleep well and work energetically. In *Low Power Electronics and Design (ISLPED), 2008 ACM/IEEE International Symposium on*. 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 *Parallel Processing Workshops (ICPPW), 2012 41st International Conference on*. 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 *Utility and Cloud Computing (UCC), 2012 IEEE Fifth International Conference on*. 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 *Computer Architecture and High Performance Computing (SBAC-PAD), 2012 IEEE 24th International Symposium on*. 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, USA, 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 *Intelligent Computer Communication and Processing (ICCP), 2011 IEEE International Conference on*. 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. http://xen.org/. (2012).

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 (February 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, USA, 1–11.

Coolit. 2013. Coolit systems. http://www.coolitsystems.com/. (2013).

P. de Langen and B. Juurlink. 2009. Limiting the number of dirty cache lines. In *Design, Automation Test in Europe Conference Exhibition, 2009. 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. *Commun. 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 *Cloud Computing in Emerging Markets (CCEM), 2012 IEEE International Conference on*. 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 *Microarchitecture, 2008. MICRO-41. 2008 41st IEEE/ACM International Symposium on*. 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, Heidelberg, 247–254. DOI:http://dx.doi.org/10.1007/978-3-642-03138-0_27

EARTH. 2011. https://www.ict-earth.eu/. (2011).

ECONET. 2013. http://www.econet-project.eu/. (2013).

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. 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, USA.

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, USA.

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, USA.

Enerdata. 2011. http://www.wec-indicators.enerdata.eu/household-electricity-use.html. (2011).

Enerdata. 2012. http://yearbook.enerdata.net/. (2012).

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:en: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 Comput. Archit. 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 *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, Vol. 3. 1548–1557.

Eugen Feller. 2012. *Autonomic an Energy-Efficient Management of Large-Scale virtualized Data Centers, PhD. Dissertation*. 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 *Cloud Computing Technology and Science (CloudCom), 2012 IEEE 4th International Conference on*. 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 Int'L Conference on Green Computing and Communications & Int'L Conference on Cyber, Physical and Social Computing (GREENCOM-CPSCOM '10)*. IEEE Computer Society, Washington, DC, USA, 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 *Mass Storage Systems and Technologies (MSST), 2011 IEEE 27th Symposium on*. 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 *Grid Computing Environments Workshop, 2008. 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. *Communications Surveys Tutorials, IEEE* 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 *Green Computing Conference and Workshops (IGCC), 2011 International*. 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. *Proc. 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. (2012). 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). *Computers, IEEE Transactions on* 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 Comput. Commun. Rev.* 39, 4 (August 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 Comput. Commun. Rev.* 38, 4 (August 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 *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. 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 Comput. Archit. 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 *Low Power Electronics and Design (ISLPED), 2013 IEEE International Symposium on*. 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 *Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm), 2012 13th IEEE Intersociety Conference on*. 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, USA, 17–17.

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

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, USA, 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 Publishers.

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 *Cloud Computing (CLOUD), 2011 IEEE International Conference on*. 65–72.

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

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

Yichao Jin, Yonggang Wen, and Qinghua Chen. 2012. Energy efficiency and server virtualization in data centers: An empirical investigation. In *Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on*. 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* (xiv ed.). Springer. 626 pages.

A.B. Kahng, Seokhyeong Kang, R. Kumar, and J. Sartori. 2013. Enhancing the Efficiency of Energy-Constrained DVFS Designs. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on* 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, USA, 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 Publishers.

D.C. Kilper, G. Atkinson, S.K. Korotky, S. Goyal, P. Vetter, D. Suvakovic, and O. Blume. 2011. Power Trends in Communication Networks. *IEEE J. Sel. Topics 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. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on* 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. *Concurr. Comput. : Pract. Exper.* 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 *ICT for Smart Society (ICISS), 2013 International Conference on*. 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, USA, 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, USA, 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 Comput. Commun. Rev.* 41, 4 (August 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* (2009).

DongKyu Lee and Kern Koh. 2009. Popularity Based Cache Management Scheme for RAID Which Uses an SSD as a Cache. In *Computer Sciences and Convergence Information Technology, 2009. ICCIT '09. Fourth International Conference on*. 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, USA, 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 Comput. Archit. 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, USA, 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, USA, 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, USA, 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. *Selected Areas in Communications, IEEE Journal on* 32, 2 (February 2014), 333–344.

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

Magellan. 2013. Magellan. http://www.cloudbook.net/directories/research-clouds/research-project.php?id=100047. (2013).

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, Heidelberg, 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)*. ACM, New York, NY, USA, Article 24, 6 pages.

A. Manzanares, K. Bellam, and Xiao Qin. 2008. A prefetching scheme for energy conservation in parallel disk systems. In *Parallel and Distributed Processing, 2008. IPDPS 2008. IEEE International Symposium on*. 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 *Cloud Computing (CLOUD), 2013 IEEE Sixth International Conference on*. 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 *CLOSER 2012, 2st International Conference on Cloud Computing and Services Science*.

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 *Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on*. 529–533. DOI:http://dx.doi.org/10.1109/ICCSIT.2009.5234658

Peter Mell and Tim Grance. 2009. *The NIST Definition of Cloud Computing* (26 ed.). Technical Report. National Institute of Standards and Technology, Information Technology Laboratory. 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. *Parallel and Distributed Systems, IEEE Transactions on* 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. *Micro, IEEE* 30, 4 (July-August 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 Twenty-first ACM SIGOPS Symposium on Operating Systems Principles (SOSP '07)*. ACM, New York, NY, USA, 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 *Computer Communications and Networks, 2009. ICCCN 2009. Proceedings of 18th Internatonal Conference on*. 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, USA, 124–131. DOI:http://dx.doi.org/10.1109/CCGRID.2009.93

OpenStack. 2012. OpenStack - Open source software for building private and public clouds. http://www.openstack.org/. (2012).

Won Ho Park and C.-H.K. Yang. 2013. Effects of Using Advanced Cooling Systems on the Overall Power Consumption of Processors. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on* 21, 9 (2013), 1644–1654. DOI:http://dx.doi.org/10.1109/TVLSI.2012.2217386

MichaelK. Patterson, StephenW. Poole, Chung-Hsing Hsu, Don Maxwell, William Tschudi, Henry Coles, DavidJ. 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 Heidelberg, 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, USA, 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 *Network Operations and Management Symposium Workshops (NOMS Wksps), 2010 IEEE/IFIP*. 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, USA, 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. *Cloud Computing, IEEE Transactions on* 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. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on* 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 Stenstrm, Michel Dubois, Manolis Katevenis, Rajiv Gupta, and Theo Ungerer (Eds.). Lecture Notes in Computer Science, Vol. 4917. Springer Berlin Heidelberg, 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 *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*. Paris.

RedHat. 2012. KVM - Kernel based virtual machine. http://www.linux-kvm.org. (2012).

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 *Parallel Processing, 2009. ICPP '09. International Conference on*. 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 *Performance Computing and Communications Conference (IPCCC), 2009 IEEE 28th International*. 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 *Advanced Information Networking and Applications Workshops (WAINA), 2013 27th International Conference on*. 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. *Magnetics, IEEE Transactions on* 45, 10 (2009), 3816–3822. DOI:http://dx.doi.org/10.1109/TMAG.2009.2024879

Junaid Shuja, SajjadA. Madani, Kashif Bilal, Khizar Hayat, SameeU. 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 *Services Computing (SCC), 2011 IEEE International Conference on*. 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. (2010).

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

Malgorzata Steinder, Ian Whalley, David Carrera, Ilona Gaweda, and David Chess. 2008. Server virtualization in autonomic management of heterogeneous workloads. (2008), 94–95.

Jan Stoess, Christian Lang, and Frank Bellosa. 2007. Energy Management for Hypervisor-based Virtual Machines. In *2007 USENIX Annual Technical Conference on Proceedings of the USENIX Annual Technical Conference (ATC'07)*. USENIX Association, Berkeley, CA, USA, 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 *Embedded Computer Systems (SAMOS), 2011 International Conference on*. 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 *Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference on*. 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 *Parallel and Distributed Processing Symposium Workshops PhD Forum (IPDPSW), 2013 IEEE 27th International*. 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. *Computers, IEEE Transactions on* 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, Inc., New York, NY, USA, 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 *Procedings 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. `DOI:`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 Perform. Eval. Rev.* 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. `DOI:`http://dx.doi.org/10.1109/ICIT.2012.6209940

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

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. `DOI:`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, USA, 199–210. `DOI:`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 *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 Fifth International Joint Conference on INC, IMS and IDC (NCM '09)*. IEEE Computer Society, Washington, DC, USA, 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.