

# On the Feasibility of Collaborative Green Data Center Ecosystems

Anna Agustí-Torra<sup>a</sup>, Frederic Raspall<sup>a</sup>, David Remondo<sup>a</sup>, David Rincón<sup>a,1</sup>, Giovanni Giuliani<sup>b</sup>

<sup>a</sup>Dept. of Telematics Engineering, Universitat Politècnica de Catalunya (UPC), Barcelona, Spain

<sup>b</sup>HP Italy Innovation Center, Milan, Italy

---

## Abstract

The increasing awareness of the impact of the IT sector on the environment, together with economic factors, have fueled many research efforts to reduce the energy expenditure of data centers. Recent work proposes to achieve additional energy savings by exploiting, in concert with customers, services' workloads and to reduce data centers' carbon footprints by adopting demand-response mechanisms between data centers and their energy providers. In this paper, we debate about the incentives that customers and data centers should have to adopt such measures and propose a new service type and pricing scheme that is economically attractive and technically realizable. Simulation results based on real measurements confirm that our scheme can achieve additional energy savings while preserving service performance.

**Keywords:** energy-efficient computing and networking, energy-aware data centers

---

## 1. Introduction

The advances in virtualization technology of the last years have shaped the evolution of the IT sector towards a model where organizations no longer sustain their IT infrastructure but rent it from third parties. This "cloud" paradigm is attractive for companies and providers as it allows for a quicker deployment of new services, relieves companies from their maintenance, and can significantly cut down companies' and providers' costs. As a result, we are witnessing a rapid deployment of new data centers (DCs) of increasing size, complexity and heterogeneity in service offerings. In parallel, several studies have estimated that the IT sector is responsible for about 1 – 2% of the greenhouse gas (GHG) emissions worldwide [1]; DCs accounting for 1.3% of the overall energy consumed [2]. The impact of CO<sub>2</sub> emissions on climate change and the rapid concentration of IT services in DCs have raised the concern about their energy sustainability. Moreover, energy consumption accounts for a significant share of DCs' operational costs. Thus, rather than a concern, energy saving has become a necessity for DCs' economic viability. Many regulations are also being established worldwide to limit corporate emissions, increasing the pressure to cap carbon footprint, to promote the procurement of power with source mixes with a larger portion of renewable energy [3].

Within a DC, energy consumption reduction is tackled at various levels. At the *component* level, for example, by employing more efficient power supplies, multiple spin rate drives, and memories or CPUs with several energy states or clock gating. At the *system* level, by employing Dynamic Frequency and Voltage Scaling (DFVS) techniques to adapt CPU parameters according to load, adjusting performance not to exceed

some power limit (*capping*), introducing power *profiles*, or supporting variable-speed fans. At an *architectural* level, by carefully locating racks, server clusters or network interconnects to optimize the effectiveness of cooling and ventilation. Last, at the *operations* level, by trying to adapt IT component utilization to workloads. These techniques seek to exploit components' power attributes by tuning them according to utilization (or even shutting down components). Examples include consolidation of workloads and virtual servers into fewer, more energy-efficient physical servers (so that some servers and network elements can be switched off), or migrating/re-locating these to other DCs with more advantageous conditions due to the weather or the cost of energy [4].

While DC-holistic approaches can significantly cut down power consumption, further savings and lower environmental impact may be achieved by widening the scope of the solution to span the entire energy consumption chain. In this direction, [5] proposes a concerted strategy toward both energy and CO<sub>2</sub> emission reduction where the three parties in the DC *ecosystem* interact; namely, the DC, its customers, and the DC's energy provider (EP). The approach relies on two ideas: first, a higher *flexibility* in the service agreements between the parties; second, their dynamic *collaboration*. *Flexibility* is supported by the inclusion of energy-related (*green*) clauses in the Service Level Agreements (SLAs), established between the DC and its customers, and the power Supply-Demand Agreements (SDAs), established between DCs and EPs. *Collaboration* allows a party to request the other to adjust its behaviour so that some energy/power consumption objective is met, and is implemented by the exchange of messages between a DC and its EP or the DC and its customers. Thus, for the EP to avoid resorting to fossil energy sources (e.g. diesel engines) due to a peak in the total energy demanded, the EP can instead request one or more of its customer DCs to temporarily reduce their power consump-

---

\*Corresponding author. Tel.: +34 934137056. Fax: +34 934137007.

Email address: drincon@entel.upc.edu (David Rincón)

tion; if a DC accepts the request, it will adjust the performance of the hosted IT applications to reduce its energy consumption (e.g. by consolidating virtual machines (VM) in fewer physical servers, delaying the execution of tasks, etc.). In case of energy surplus, the performance of the IT services provided by the DC can be upgraded (by increasing the physical resources allocated to them, advancing the execution of scheduled jobs, or accepting workloads from other DCs).

The idea of reducing power usage in a demand-response (DR) fashion when supply is scarce is not new. [6] describes a use case where a company reduces its power demand by automatically adjusting lighting, thermal settings or rack power distribution units (PDUs) upon an EP's request. While this form of collaboration or those in [5] and [7] can effectively lead to an environment friendly DC ecosystem, all the different parties must have an incentive to adopt them –aside from environmental stewardship–, as this may critically impact their businesses.

[5], [6] and [7] suggest the incentives to be financial, based on *rewards* or *discounts*. EPs will make discounts to DCs that diminish their power demands. As this may require changing the operating conditions of applications, DCs' *customers should agree in advance* to reductions in performance and be compensated with lower tariffs *accordingly*. Otherwise, DCs would have little interest in collaborating with EPs: the economic loss that resulted from compensating customers for SLA violations could outweigh EP's rewards and cause customers to choose other providers. Still, even if DC customers adhered to flexibilize the committed performance levels, finding a suitable reward/pricing scheme is hard given the multilaterality of the DC ecosystem: customers' discounts might not pay off the performance degradation experienced; on the other hand, while a DC would see its energy expenditures reduced, it could still see its profit diminished (due to lower revenues resulting from the use of lower tariffs during energy saving episodes). We conclude that, for any energy saving collaborative framework as those described to succeed, *it must permit reaching some equilibrium point where customers are satisfied with the service received (relative to its price), and where DCs and EPs do not see their profits depressed*. A second condition is that the technical requirements to implement it must be simple enough for DCs to adopt it. In this regard, we note that, while [5] establishes a framework and methods to deploy such a paradigm in a technically realizable manner, these do not warrant the above requirement by themselves. This paper seeks to contribute in this direction by proposing a service model and pricing scheme that is attractive for users, profitable for DCs and EPs and technically feasible, while promoting energy savings in a collaborative manner. We discuss benefits that its adoption would bring to all the parties and assess its advantages and disadvantages with results obtained with a simulator (contrasted with measurements from real DCs), fed with real and synthetic service demands and load patterns, real server parameters and typical energy costs.

This paper is organized as follows. Section 2 briefly describes the components of the DC ecosystem. Section 3 summarizes related work, including ideas in [5] that contextualize and motivate this work. In Section 4 we discuss potential barriers

that may preclude the adoption of collaborative measures as the ones discussed. Section 5 presents the motivation, assumptions and idea behind our proposal, expected advantages. Section 6 identifies the major variables that need to be considered to measure such advantages and presents how we study their evaluation. Sections 7 and 8 discuss our experimental work: the tool employed, how it works, how we parameterized it and the results obtained. Section 9 presents general conclusions of this research, recommendations concerning the feasibility of the proposal and future directions.

## 2. Background and terminology

### 2.1. Data Centers (DCs)

A data center is an infrastructure built to provide IT services such as massive data storage, CDN, web, e-mail, server hosting, enterprise-class applications, or on-demand computing. Such services often consist of several components like databases, front-ends, application servers or middleware, increasingly being deployed as virtual machines. The business model varies, but typically a DC is either a service within a large corporation, or a business by itself that resells some of its infrastructure to third parties or "tenants".

While varying in size, purpose and structure, DCs typically consist of large farms of servers executing services for end users. Connectivity is provided by high-speed network equipment. Servers execute application software related to the offered services (e.g. web, e-mail, social network or video) as well as management software (e.g. backup or antivirus). Servers may host several VMs, controlled by a *hypervisor* software and managed either by the DC or, in a in cloud computing scenarios, indirectly by the users themselves. Apart from IT systems, DCs have facilities such as components used to deliver power (PDUs) and cooling to IT equipment. Most DCs use uninterruptible power supply (UPS) systems to protect the equipment and guarantee the supply in case of outages. In case of emergency, servers hosting secondary services can be switched off, and migration can be used to consolidate services. Hence, UPS systems must maintain power supply to keep the critical machines running while providing enough time to relocate critical services and orderly switch off the hosted machines. DCs use cooling systems and air flow strategies to keep hardware temperature below a critical value beyond which it may malfunction or damage. Lastly, some large DCs also have on-site renewable power sources.

Energy costs are becoming the major expense in DCs and this trend is increasing as hardware and software drop in price. Several studies show that most of the power consumed by a conventional DC is due to its servers (up to 50%), while, approximately, the other half of the power is due to cooling [8].

Many benchmark studies (e.g. [9]) have empirically found that the power consumed by servers increases quasi-linearly with the utilization of their cores, as depicted in Figure 1. Thus, the power consumed by a server can be accurately modeled as

$$W(u) = W_{min} + (W_{Max} - W_{min}) u \quad (1)$$

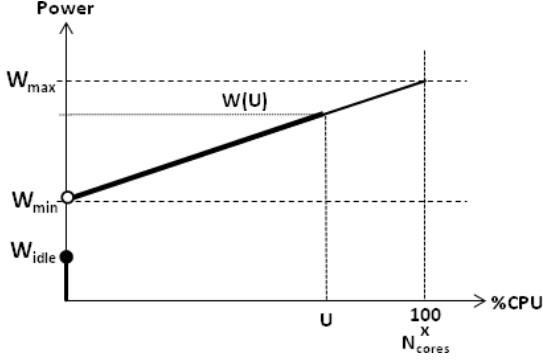


Figure 1: Typical power-utilization curve of a physical server.  $W_{min}$  refers to the *active-idle* power consumption, where server is in non-dormant state but with zero CPU utilization; while  $W_{idle}$  is the power usage in some dormant state.

where  $u$  refers to the % utilization of all its cores,  $W_{Max}$  is the *peak* power,  $W_{min}$  that at 0% utilization and  $(W_{Max} - W_{min})$  is the *dynamic power range*. The fact that servers consume even if idle –i.e. their lack of *proportionality*– is what precisely motivates the consolidation of workloads in other servers to turn some off, and has driven vendors to manufacture more efficient processors. In this regard, note that  $W'(u) \triangleq W(u)/W_{Max} = b + (1 - b)u$ , with  $b = \frac{W_{min}}{W_{Max}}$ , where  $b$  can be understood as a measure of proportionality (the lower the better). In the literature, other indices similar to  $b$  are used to quantify energy proportionality, such as  $b(1 - \frac{u}{2})$  or  $\frac{u}{W'(u)}$ . Similarly, several metrics are used to quantify the energy efficiency of DCs. A common one is the PUE (*Power Usage Effectiveness*), defined as the ratio between the total energy consumed over that spent solely by IT equipment. A lower PUE means higher efficiency. The PUE, which typically ranges in (1, 3) [10], does not account for carbon emissions. Another metric is the *Carbon Usage Effectiveness* (CUE) defined as the product of PUE and CEF; where the CEF or *carbon emission factor* is the amount of CO<sub>2</sub> grams released per kWh. CEF can be non-zero for renewable sources if the emissions due to manufacturing, transport, installation or recycling of their equipment are considered.

## 2.2. IT Customers, users and IT Services

We call *IT Customer* (ITC) the entity that contracts the provision, execution and maintenance of some application –*IT service* (ITS)– in a DC, as opposed to *users*, which are those employing the service. DCs provide a wide array of services, from consumer-oriented to enterprise-class offerings. Thus, ITCs can be single users, but also companies offering services to other individual users, businesses, or departments within themselves. An *IT service* is understood as a collection of servers (*physical* or *virtual*, hereafter called *virtual machines*, VMs) and software entities that jointly execute a task. IT services have a wide range of complexity: from single VMs started and controlled at the customers will, to complex cloud computing applications involving several physical servers, many VMs, and auxiliary sub-services such as antivirus or backup procedures. This model where customers pay to build a virtual infrastructure on top of a DC's physical one is called *Infrastructure-as-a-Service* (IaaS).

SLAs specify the conditions under which a certain type of IT service is to be provided, the tariff to be paid, and the penalties (usually discounts) in case the DC fails to deliver the service under the agreed conditions. The key performance indicators (KPIs) most often found in SLAs relate to availability (service uptime) and performance. Cloud computing services consisting of VMs can also include as KPI detailed parameters such as start-up delay, computing power<sup>1</sup>, memory, OS image, I/O performance or disk size (refer to [11] for examples). One way of quantifying the resources available to a virtual server (and thus its performance) is by a *consolidation ratio*  $C_R$ , defined as the number of *virtual* CPUs (vCPUs) of VM instances running per server or CPU core.

DC automation frameworks take care of provisioning the necessary resources (and monitoring them) for each IT Service, by mapping the high-level clauses and KPIs found in SLAs into low-level operational parameters. For instance, a web-hosting service may include a “*high-performance*” SLA clause that translates into 12 load balanced VMs (running in 3 servers) able to handle 1000 requests/sec; while a “*medium performance*” may refer to the same service running 6 VMs on 2 servers of lower capacity able to process 300 requests/s. In cloud/IaaS environments, service instances typically belong to a certain class (e.g. *small*, *medium* or *large*) with some predefined attributes (e.g. 1, 2 or 4 vCPUs, and 2, 4 or 8GB of RAM).

## 2.3. Energy Providers (EPs)

An EP is a company that produces and sells energy. *Supply-demand agreements* (SDAs) state the conditions under which power is delivered to customers. These include the price per kWh, a peak power demand and availability. Energy costs vary by region. According to [12], the average industrial rate per kWh is \$0.0677 in the US and €0.0836 in Europe.

Typically, EPs manage several power generator facilities like coal generation plants or gas turbines to satisfy the demands. It is very important to EPs that the power they produce matches the power that is being demanded and not feed the grid with more or less energy than needed (otherwise the grid could collapse). Therefore, it is crucial for EPs to correctly plan energy production ahead, and to rapidly react to unforeseen changes in demand. Prediction of power generation capabilities and power demands is critical for the EP to be able to plan economically (and also ecologically) which power sources it schedules. The reaction time of power sources varies significantly: for instance, coal power plants need several hours to shut down, while diesel generators can be turned on/off rapidly. And the availability of renewable energy can highly vary depending on wind speeds or solar irradiance.

EPs make use of customer load profiles and categorize them. For major customers (e.g. DCs), power consumption is measured directly to precisely tune prediction methods. To plan power generation, EPs take into account weather forecasts to predict the amount of renewable energy that will be available.

<sup>1</sup>Measured in abstract units such as Amazons Compute Unit, equivalent to a 1.0-1.2 GHz 2007 Opteron/Xeon processor (see [11]).

To quantify the extent to which production conforms to demand, EPs consider the ratio  $R$  between generated and demanded power. *Energy shortage* occurs when  $R$  deviates to values  $< 1$ ; while *surplus* when  $R > 1$ . Both types of episodes are critical for an EP if it is not able to timely compensate them with its own power sources. This is the reason to tighten the collaboration of DCs and EPs on a demand-response fashion.

The energy consumed by a DC typically accounts for a relatively small percentage of the energy produced by its EP. Nevertheless, the ability to quickly reduce the power demand even if in a small percentage can help an EP to avoid resorting to expensive and highly contaminating power sources. Thus, for an EP, a DR scheme constitutes an effective, practical means to dynamically influence the power demanded by customers. On the other hand, an EP may supply energy to multiple DCs, their joint demand constituting a larger fraction. Thus, the larger the number of DCs served and their share of the demand, the easier it can be for an EP to influence the demand (and mitigate peaks quicker) as this offers higher degrees of freedom.

### 3. Related work

There are diverse solutions to reduce DCs' energy consumption, such as techniques where servers' processor states are aligned with job demands (e.g. DVFS), system-wide approaches that reduce servers' idle powers [13], per-core power gating [14], consolidation of servers during low workload periods (e.g. [15]), virtualization techniques to obtain energy savings [16], [17] or the use of low-latency power state changes in servers to improve the impact of virtualization [18]. To complement these, there are many contributions that focus on the characterization of workloads in DCs, such as [19].

[20] considers the adaption of DC's power demand to fluctuations in electricity pricing considering the DC as a whole, disregarding the discrete elements inside, while [21] contemplates the use of on-site energy storage to avoid peak power costs. [3] studies the reduction of DCs' carbon footprints by using on-site renewable energy production. By managing resources across multiple DCs, [22] achieves power savings. Similarly, [23] contemplates the dynamic exchange of workloads among different sites, where services are moved to DCs with higher efficiency—both in terms of performance versus power consumption ratio and also depending on the carbon footprints of the energy generation at their EPs—.

As mentioned, the co-operation paradigm recently proposed in [5] is based on leveraging flexibility and dynamic collaboration among all the DC ecosystem players: the EP, the IT customers and the DC itself. Flexibility is enhanced by adding energy-aware extensions to SLAs—stating the conditions (QoS parameters, guarantee terms and associated tariffs) under which IT services are provided—which then become GreenSLAs. These extensions establish the degree of change in running conditions (e.g. performance or availability) that the DC and its customer agree to accept as a function of the *energy context* of the ecosystem, together with specific tariffs and guarantee terms. *Energy context* refers to the state of the ecosystem regarding energy, and it can be either calendar related or depen-

dent on the actual situation of the EP. Calendar-based clauses relate running conditions with the time-of-the-day. An example of a calendar related clause could be that a service enjoys a 99.95% availability for weekdays during working hours and a 95% – 99% availability for the rest of the time.

On the other hand, the collaboration paradigm, illustrated in Figure 2, contemplates interactions between EPs and DCs related to the current balance between the energy demand and production of the EP, according to extensions in the SDAs (called GreenSDAs), such that an EP can issue requests to a DC to reduce or increase the consumed power for a certain time period. If the DC accepts to serve the request, there will be clauses in the GreenSLA that allow the DC to modify its power consumption by changing running conditions of some IT services. An example of such a clause could be that, if a request from an EP to reduce power consumption for a time period  $T$  has been accepted by the DC, then a Web hosting service that is running in a *high performance* condition (say, 12 load-balanced VMs able to handle 1000 requests/s) can be set to run in *medium performance* (6 load-balanced VMs) during  $T$ .

The collaboration between the three DC ecosystem parties will also follow the rules established in the GreenSLAs (between the DC and the IT customers), and the GreenSDAs (between the DC and the EP), together with the associated economic impact for every change in running conditions of the IT services or the consumed power of the DC, respectively. An example of such rules could be that the EP may request the DC to reduce its power consumption for periods no longer than one hour, during which a certain discount in the electricity price will be applied.

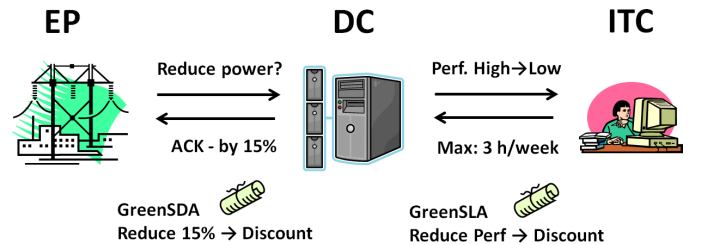


Figure 2: Interaction among the three DC ecosystem parties.

### 4. Current DC service models and barriers potentially hindering collaboration in the DC ecosystem

As discussed in §1, at least two conditions should be met for energy saving collaborative models to be adopted. First, the measures to save energy should not conflict or impair IT customers interests (e.g. businesses), who should still perceive adequate service levels relative to their cost. Second, energy-saving measures and their consequences (e.g. on pricing) should not affect DCs' economic viability (i.e., they should not cause DCs' profits to be reduced). To understand the motivation behind our proposal and its advantages, next we briefly discuss existing service models, pricing schemes, and the effect of energy costs in DC's economies through a simple model.

#### 4.1. Current service and pricing schemes

The service models and pricing schemes offered by public DCs (e.g. [11][24][25][26][27][28]) are diverse and rapidly evolve nowadays (particularly in cloud providers) as a result of the technology quickly advancing and competition. Generically, however, providers employ a billing scheme that depends on three dimensions: *performance* (committed resources), *up-time* and *resource utilization*. Customers can contract one or more instances of a certain class (specifying OS, amount of RAM, disk capacity, CPU power or network bandwidth) and pay accordingly; a monthly fee in traditional services (e.g. managed hosting), or only during the time that the service is active (in cloud/IaaS, typically on a hourly basis). Some providers also charge when the service is off, but at a lower tariff (e.g. 50% less) to cover the cost of storing customers' data. In others, pricing depends on the number of I/O operations performed, the amount of RAM available to VMs, or the network bandwidth used. To cope with increasing service's demands, some DCs have *elastic* offerings where the resources allocated to IT services can be scaled (and billed accordingly) depending on their loads, either automatically or by the customers themselves via a Web *dashboard* or some API.

#### 4.2. Data center economics and the effects of energy costs

To determine adequate tariffs (e.g. prices per unit time in pay-as-you-go schemes as the above), DCs need to find an equilibrium between two competing factors: their *profit* and their *competitiveness*. In a simple model, the *profit*  $P$  can be defined as the difference between the DC's *revenues*  $R$  (what customers pay) and running costs  $C$ , over some period of time (e.g. monthly). Costs include recurring costs (personnel salaries, licences, taxes, insurances, maintenance, leasing of buildings, etc.) and the amortization of facility or IT equipment and building costs. Part of these running costs are strictly due to energy consumption and the rest are independent of this factor. The costs due to energy consumption,  $C_e$ , are due to the energy consumed by servers and interconnects, lighting and cooling, and can be written as  $Er$ , with  $E$  the amount of energy consumed in a time period and  $r$  the price per kWh. We refer to the running costs that are independent of energy consumption as *fixed costs*.

Thus, in a simple economic model, we can express DC costs as  $C = C_f + C_e$ —with  $C_f$  the fixed costs and  $C_e = Er$  the energy related expenditure—and its profit as  $P = R - C = R - (C_f + C_e)$ , where, to be economically viable in the long term,  $R > C$ .

While well established cloud providers have reported significant profits in the past, new players continue to appear, offering services at tariffs often 40% lower [29]. In an increasingly competitive market as this, revenues  $R$  are expected to progressively approach costs  $C$  from above, as new providers emerge and DCs strive to capture/retain market share. Indeed, big providers have announced price reductions over time. Thus, in our model, we can assume that  $P \ll R$  (or  $\frac{R}{P} \gg 1$ ).

Since energy consumption represents a significant fraction of the total costs and it varies depending on IT workloads, finding the right tariff scheme is hard for a DC. A *conservative* approach (e.g. billing under the assumption that servers

or network devices consume peak powers  $W_{Max}$  even if they do not) would maximize profit; however, profit maximizing schemes are not always the optimal strategy as could be, for instance, capturing market share. Conversely, setting tariffs below the worst-case cost of energy can attract users, but also lead to lower profit or even economic loss. The point is that, if energy costs account for a large fraction of the OPEX, their variation can have a high impact on DCs total costs and profits. To illustrate this, let us see how a certain variation  $\Delta E$  in the energy spent by a DC may affect its overall costs. As  $C = C_f + C_e$ , if we assume that  $C_e = \alpha C$  for some  $\alpha < 1$ , and the fixed costs  $C_f$  to be independent of the  $E$ , we have  $\frac{\Delta C}{C} = \frac{\Delta C_e}{C} = \frac{\Delta E}{C} r = \frac{\Delta E r}{C} \frac{C_e}{C} = \frac{\Delta E}{E} \alpha$ , which says that a 20% variation in the energy consumption translates into a  $20\alpha\%$  variation in the total costs. Similarly, by definition, profit is  $P = R - C$ . Assuming that revenues  $R$  do not depend on  $E$  (as occurs in *XaaS*-services where customers pay fixed fees),

$$\frac{\Delta P}{P} = \frac{-\Delta C}{P} = -\frac{\Delta C}{C} \frac{C}{P} = -\alpha \frac{\Delta E}{E} \frac{R - P}{P} = -\alpha \frac{\Delta E}{E} \left( \frac{R}{P} - 1 \right)$$

which, under the assumption that  $R \gg P$ , is  $\frac{\Delta P}{P} \approx -\alpha \frac{\Delta E}{E} \frac{R}{P}$ , and says that a DC's profit can significantly vary depending on its energy consumption. For instance, for  $\alpha = 0.5$  and  $\frac{R}{P} = 6$ , a 10% increase in the energy bill will cut profit by 25%.

If finding a suitable pricing scheme is hard with conventional service offerings, the use of "green" SLAs will complicate the matter further, which may discourage their adoption. Suppose that a DC (employing a conservative billing approach) were to offer, according to some "greened" SLAs, two service performance levels abiding some calendar: a *high* one (for business hours), and a *low*, *energy-saving* one for the rest of time. During non-busy hours, the DC would save energy and see its energetic costs reduced. However, depending on how the price per unit time were set for those periods, it could see its interests impaired: too low of a price could cause the DC to see its revenue (and thus profit) reduced, while higher prices may not be accepted by customers if the price reduction did not pay off the performance one; causing the DC to loose market share (and again profit), compared to the case where no performance and price adjustments were made to save energy. Hence, given the expected increase in pricing sensitivity (due to higher competition), providers may be reluctant to adopt such measures.

### 5. Proposed service and pricing scheme

#### 5.1. Motivating observations and assumptions

As per the previous discussion, performance-centric pricing schemes can make it difficult for DCs to predict profits and establish competitive tariffs, given that fees do not necessarily match the costs incurred to run services, like those related to power consumption. Moreover, *customers have no stimulus to promote or claim for energy efficiency, but performance*.

Studies on DC's workloads have shown that servers are seldom idle but rarely operate at maximum CPU utilization [30]. Moreover, the load of many IT services may fluctuate and be, in several periods, far below the peak, for which resources are

provisioned (and customers billed irrespective of their usage). Under these circumstances, many customers could prefer a service whose tariff depended on the real utilization of their services, more than on the resources the DC provisions to them. Further, some could accept some performance degradation for the purposes of energy saving if this translated into a lower bill; particularly if they were given the assurance that, at peak or busy hours –when their services are mostly demanded– some prescribed performance metrics or resources are guaranteed.

## 5.2. Proposed pricing model

The pricing scheme that we propose is conceptually fairly simple. Instead of setting prices according only to performance, utilization or allocated resources, let *service fees include a fixed part plus a variable one that equals the monetary cost of the energy consumed by an IT service*. To exemplify how this scheme could work, consider a service consisting of a single VM charged only when the instance is running on a time slot basis, such as 10 minutes. The fee at the  $k^{th}$  slot could be  $p_k = f + e_k r_k$ , with  $e_k$  the energy spent by the instance and  $r_k$  the price per kWh. The value of  $f$  would depend on the instance properties (e.g. OS image) and be derived to apportion for DC's fixed costs  $C_f$ , including some quantity to achieve a certain *target profit*. If the IT service consisted of  $I$  components or instances, the fee for each would be  $p_{i,k} = f_i + e_{i,k} r_k$  and the overall service fee in a slot equal  $P_k = \sum_i^I p_{i,k} = \sum_i^I f_i + \sum_i^I e_{i,k} r_k$ . Thus, the bill after running the service over  $S$  slots would amount to

$$B = \sum_{k=1}^S P_k = \left( S \sum_i^I f_i \right) + \sum_{k=1}^S r_k \sum_{i=1}^I e_{i,k}$$

where the second term corresponds to the energy spent by the service throughout, weighted, at each slot, by the kWh rate. Note that if the  $f_i$  are identical (some  $f$ ) and the rate of energy constant  $r$ , the above is simply  $SI(f + \bar{E}r)$ , with  $\bar{E}$  the average energy consumed per instance over  $S$  slots.

Note that the above model is kept purposely open. For instance, the energy figures ( $e_i$ ) can include or not (part of) the costs of cooling or those of the network interconnects, which could, otherwise, be apportioned to the fixed terms  $f$ . The important fact is that a direct inclusion of energetic costs into fees can solve the issue with respect to the incentives that a DC has to offer variable-performance schemes to save energy; be these on a calendar basis or in response to explicit requests from EPs: if users pay proportionally to their energy consumption, DCs' profits can remain the same even when resorting to lower tariffs during energy-saving episodes, *because this is as if customers paid themselves the watts they consume directly to the EP*. In other words, during energy-saving periods, the DC's turnover can be reduced exactly by the same amount that its energy costs, hence keeping its profit roughly the same. Moreover, DC profits can be much less sensitive to energy expenditures.

Explicitly including energy expenditures in fees can increase customers' awareness of the environmental impact of their businesses, be fairer –not only environmentally, but also economically–, and makes particular sense given that energy

proportionality is becoming a primary design goal. Still, *would such a pricing scheme be positively perceived by customers?*

Next we propose a service model which, subject to the above pricing scheme, can be attractive for customers and DCs.

## 5.3. Elastic Energy-Aware Calendar-driven Services (EEACD)

As DCs may host IT services that are mission-critical for the tenants' businesses, these may demand stringent performance requirements. As argued, however, many would adhere to relaxing their performance indicators during low utilization periods –as long as certain levels (and the availability and uptime of their services) were guaranteed– in exchange for a cost reduction. One type of service allowing for this while promoting energy savings and letting DCs adjust power demand upon energy providers' requests can be as follows.

1. IT customers specify (in a *green* SLA) the minimum performance level (*min*) that the DC should grant on a calendar basis (e.g. at each time slot in a week), as well as a maximum one (*max*), *for each parameter and component of its service*. Thus *min*, *max* may be thought of as vectors, whose dimensions refer to measurable metrics as computing power, RAM, or network bandwidth, etc.
2. The customer is billed as described, depending on an estimate of the energy spent and the energy rate  $r$ .
3. The DC commits to offer *elasticity*: if service loads surpass their capacity, additional resources (e.g. CPU power) are allocated, within the *min*/*max* bounds.
4. For a DC to (further) cut its energy demand when requested by its EP, a second calendar specifies another set of *min*/*max* values for each IT service. We refer to the first calendar as the *regular calendar* (RC), and call the second, the *energy-saving calendar* (ESC). The DC commits to apply the ESC no more than a certain fraction of the time, e.g. 3.3% (1 day in a month, or 8h a week). During the time the ESC is obeyed, a lower energy rate  $r'$  applies. Thus, EP's rewards can come in the form of a rate reduction from which both a DC and its customers can benefit.

## 5.4. Can such a service suit customers, DCs and EPs?

The attentive reader may wonder about the purpose of a calendar if a DC can dynamically adapt services' resources to their demands. The role of the calendar is severalfold. First, while DCs employ monitoring tools to efficiently provision resources, migrating or consolidating VMs takes time and resources. The use of a calendar can ease the task of planning ahead and later rearrange computational resources. Second, it can be the basis against which SLA conformance is verified. Most importantly, the *min* values guarantee customers some minimum performance for their services on one side. On the other hand, these indicate energy-saving opportunities which, if exploited by a DC (e.g. if service loads follow the calendar), can reduce not only the overall power consumption, but also customers' bills. The role of the *max* values is that of bounding from above the cost per slot, by preventing the DC from allocating resources in excess. Lastly, *busy hours* can vary from service to

service, each of which can have distinct performance requirements. Specifying a calendar on a per service basis can help to optimize overall service performance and energy consumption.

Arguably, one possible downside of this kind of service could be that customers do not know the service fee in advance, as the price per unit time depends on the usage of the service, the DC decisions regarding consolidation and the activation of energy saving measures in response to requests from its EP. While slot prices are unknown, these are bounded from above (due to the max values) by a known quantity (probably close to the fee had the service been provisioned inflexibly for peak utilization). Thus, customers pay no more (and often less) than in case they were charged by allocation, while being granted a relatively high performance due to elasticity. Moreover, slot fees can be reported by the DC in nearly real time in services' *dashboards*. Thus, depending on the service, customers can modulate their computational loads. Lastly, in case of commercial customers, the more utilized a service is, the higher its bill, but also the higher their tenants' revenues, and vice-versa. Companies may positively perceive this service as a safe way to mitigate financial bleeding in case of business failure. *How about DCs?*

By including energy costs into service prices, data centers can keep their profits independent of the fluctuations in services activity and not be economically affected by energy-saving measures. It can also reduce their financial risk and ensure a predictable and measurable ROI. In addition, for reasons of environmental awareness, a growing number of conscious customers may choose green data centers in front of other options, even if at higher tariffs, altruistically. Thus, a DC adopting the above services may even see its customer base enlarged. On the other hand, if customers defrayed their energy costs, *what would be the incentives of DCs to save energy?*

First, in the proposed scheme, not all energy costs of a DC need to be charged to customers (e.g. lighting and part of cooling). Most importantly, customer bills will depend on the cost of the energy spent by their services, which depend on their load, *but also on how energy-efficient a DC is and the cost of energy*. Thus, DCs adopting the proposed service *may still strive to be as energy-efficient as possible to offer competitive prices*.

Hence, the scheme is such that all the participants of the DC ecosystem are induced to save energy. IT customers, to lower their costs (and widen their profit); DCs to stay as competitive as possible by purchasing efficient equipment, keeping up with eco-friendly technological advances and collaborating with EPs to get discounts and offer more economic tariffs to customers; and EPs by avoiding peak demands and thus maintaining the stability of the electricity distribution grid.

In the sequel, we aim at understanding *the extent* to which this scheme can be attractive for DCs and their customers and the energy be reduced. Next, we discuss additional benefits and the requirements from a technical perspective.

### 5.5. Additional benefits

There is an increasing environmental awareness of business activities, and the IT sector is not an exception. Corporate social responsibility, codes of conduct, and other forms of

self-regulation currently include energy consumption and CO<sub>2</sub> emissions. The ISO 14000 standards define the methodology for accounting the environmental impact of companies and help businesses comply with regulations or self-imposed goals. ISO 14604 [31], in particular, provides tools for quantifying GHG emissions, and is used by public or independent entities (e.g. the Carbon Trust in the UK [32]) to certify carbon footprints. EMAS (Eco-Management and Audit Scheme) [33], developed by the EU, is another example of greenness certification, and includes the ISO 14001 recommendations. Finally, international treaties such as the Kyoto Protocol have established limits on CO<sub>2</sub> emission volumes, whose rights can be traded; which establishes market-based incentives to reduce GHG. The proposed paradigm permits and incentivizes energy savings and cleanness. As all the participants could be required to meet such regulations, they all may benefit from it.

The main benefit for EPs may be the possibility of reducing peak demands at times when the energy production structure is under stress, due to a shortage or because the energy mix is biased towards fossil-based sources. This can result in a more reliable, stable and safer energy grid, and less penalties for an EP, which will then reach its environmental goals more easily.

### 5.6. Technical requirements

Deploying the service type proposed requires: a) measuring the amount of resources utilized by services 2) scaling physical resources to load (e.g. by migrating/consolidating VMs), and 3) measuring/estimating the energy consumed by services.

The first two requirements are readily covered in today's DCs. Many tools exist to monitor CPU loads, memory usage or network traffic. The time scales at which the monitoring should be performed in the above services (several seconds or minutes) are compatible with the performance of such tools, whose overhead is within acceptable levels at those timescales [34]. On the other hand, to consolidate and perform live VM migrations, cloud providers already employ open source and commercial products<sup>2</sup>, most of which come with monitoring functions built in. Typically, migration and consolidation times are on the order of a minute or less [35] [36], which may be acceptable for the above services. Also, VM migration: 1) may cause an interruption in the service or lengthen the execution time of jobs, and 2) causes extra load (and thus energy consumption) at the origin and destination servers. However, modern techniques minimize service interruption, and the increase of CPU load and power consumption of the destination server may be in the order of 20% according to [35][36]<sup>3</sup>.

Measuring the power and energy consumed by servers (and network equipment) can be performed directly with dedicated hardware, or indirectly from analytical models. Racks usually include PDUs able to report (via SNMP) the power consumed of each server at a second resolution [37]. Indirect methods use

<sup>2</sup>E.g. IBM's System Planing tool, VMware's Distributed Resource Scheduler (DRS), Rackwise's DCiM, or the OpenStack suite

<sup>3</sup>The figures refer to servers with Intel Xeon E5620 4-core CPUs at 2.40GHz, but the relative increase can be extrapolated to other models/vendors.



properly parameterized and trained models to estimate the consumption of servers from a measured figure of CPU utilization [38]. Thus, the power and energy spent by services can be estimated from such figures and models in conjunction with the measures of the resources they spent. For instance, in case of servers, if a certain VM has utilized  $\bar{u}\%$  of a CPU core on the average during a time slot of  $T$  seconds, the energy consumed can be estimated by  $\hat{e} = T \left( (W_{max} - W_{min}) \frac{\bar{u}}{N_{cores}} + \xi(W_{min}) \right)$ , where the first term corresponds to the fraction of the server's dynamic power range apportionable to the VM (as per (1)) and  $\xi(W_{min})$  the share of the server's active-idle power attributed to the VM, which may be derived in several ways. The logic we employed in our simulations was to split  $W_{min}$  among all the VMs hosted in a server proportionally to their minimum share of CPU reserved, which is the inverse of the consolidation ratio.

The above only accounts for server energy. Regarding other important energy sinks (network components and specially cooling), similar estimates can be derived. For instance, router and switches figures can be weighted by traffic volumes, readily measured. For simplicity, in our simulations we consider these to be proportional to that of servers.

## 6. Studying the feasibility and advantages of the approach

To assess the benefits and downsides of the proposed scheme, we need to understand the extent to which energy expenditure at the DC may be reduced due to calendaring and a DR collaboration with EPs, the impact on service performance and if the resulting prices may be worthwhile and competitive. To that end, we consider three factors (energy, power and performance indicators) to compute prices according to our scheme. Then, we compare the outcome against those obtained in case SLAs were inflexible, prices independent of energy expenses and the power demand not adapted. Of the above, experienced performance is the hardest to quantify, given the distinct nature and purpose of services, their variety, and the number of factors potentially influencing. Moreover, aspects like user satisfaction and perception come into play, which entail the use of *utility functions* to be assessed.

For modeling purposes, services can be categorized depending on whether they are up on a continuous basis or sporadically. Thus, a DC may host *always-on* services or *time-scoped* ones. Examples of the first kind abound and are those offered by traditional computing, such as e-mail, web or database, which, deployed in bare-metal or as VMs (i.e. as IaaS), are typically offered to a large set of users, have a direct or indirect revenue and require 24x7x365 uptimes. Services of the second kind may last for a limited period and could include from the execution of high-demanding simulations, batch processes, management tools (backup or antivirus), user-oriented remote desktops, to VMs instantiated to dynamically scale services, as computing platforms like MapReduce/Hadoop permit for data mining purposes, or Cassandra and alike for distributed databases.

Services can also be classified, depending on how load is serviced, into *workload-conservative* and *non-conservative*. In the

former, demands cannot be lost but delayed, as occurs with simulations, backups or an antivirus scans. In the latter, unsatisfied loads are lost. These include streaming, web, real-time collaboration systems. As a first approximation, our simulation study considers *always-on* and *time-scoped* services but only of the workload non-conservative type.

### 6.1. Measuring service performance

IT services may consist of several components, each conducting a distinct task. Deriving a generic metric that quantifies the performance exhibited by one such component is hard, and so is one that reflects that by a service as a whole. Indeed, the assessment performance in virtualized environments is an active area of research [39]. Several functions  $P$  may be used as *performance scores*. Next, we discuss the rationale behind the one we chose, which is suitable for workload non-conservative services and balances generality and simplicity. Recall that the purpose is not quantifying application performance in *absolute* terms but rather *comparatively*, to see the effect of adjusting service resources according to calendars or EP requests.

When a DC provisions some component (of some class), it *allocates a certain amount of resources* (e.g. disk space, memory, and a certain share of a CPU/core in VMs) which, apart from enabling the service, guarantee that some performance indicators fulfill those dictated by SLAs (e.g. the min vectors of calendars). For simplicity, suppose that a service is composed of a single VM. The performance experienced by a service  $P$  can be modeled as a function of three variables at each point in time: 1) the amount of resources (of some kind) *allocated* to a service  $a_n$ , 2) the amount of such resources *demand*  $d_n$ , and 3) the quantity of such resources *utilized*  $u_n$ .

In these terms, and irrespective of the performance score  $P$  employed, it is reasonable to accept that:

- I.  $u_n \leq d_n$ , i.e. utilization cannot exceed demand.
- II. If a service exactly utilizes what it requires,  $u_n = d_n$ , then performance  $P$  must be highest (e.g. 1 or 100%) and monotonically decrease as  $u_n$  deviates from  $d_n$ .
- III. In case the system provides resource allocation guarantees:
  1. If the demand does not exceed the allocation ( $d_n \leq a_n$ ), all the demanded resources are granted ( $u_n = d_n$ ), implying  $P = 1$ .
  2. If it does, at least the resources allocated are granted; and more in case spare resources exist and the system allows them to be borrowed ( $u_n \geq a_n$ ).

The above considerations mean that, under resource guarantees, in the worst case,  $u_n = \min(a_n, d_n)$  and  $P|_{d_n \leq a_n} = 1$ , regardless of how  $P$  is defined. A family of functions satisfying the above conditions is  $P_n^*(u_n, d_n) = \left( \frac{u_n}{d_n} \right)^\alpha$ ,  $\forall \alpha \geq 1$ , as

$$P_n^*(u_n, d_n) = \left( \frac{\min(a_n, d_n)}{d_n} \right)^\alpha = \min \left( 1, \left( \frac{a_n}{d_n} \right)^\alpha \right)$$

is 1 for  $d_n \leq a_n$  and, otherwise, decreases as  $\left( \frac{a_n}{d_n} \right)^\alpha$ , with  $\alpha$  governing the rate of decay as demands surpass guaranteed resources (or  $u_n$  deviate from  $d_n$ ). In our simulation study we take



$\alpha = 1$ , as then  $P^*$  represents a measure of *relative throughput*. We also compute the average performance of a service running over  $S$  slots as  $\bar{P}^* = \frac{1}{S} \sum_{n=1}^S P_n^*$ . In this regard, note that we can write  $\bar{P}^* = \frac{1}{S} \left( \sum_{d_n \leq a_n} 1 + \sum_{d_n > a_n} (a_n/d_n)^\alpha \right) \geq \frac{1}{S} \left( \sum_{d_n \leq a_n} 1 \right) = \beta$ . where  $\beta$  is the fraction of time where  $d_n \leq a_n$ . This has an important implication in our context. Whether services are sensitive or not to resource exhaustion (value of  $\alpha$ ), if calendars accurately reflect service loads (i.e.  $d_n$ ) or the DC can quickly scale resources ( $a_n$ ), service degradation may be small on the average, while achieving energy savings and lowering prices.

## 7. Simulation environment

Several DC/cloud simulators exist in the literature, which focus on distinct aspects of DC scenarios. CloudSim [40] is oriented to the dimensioning and modelling of cloud environments (and federations) and the evaluation of resource provisioning algorithms. DCWorms [41] is a recent tool able to compute the performance and energy consumption of DCs using diverse consumption models and workloads. It can be used also for the thermal optimization of the physical location of the equipment in the DC room. GreenCloud [42] is an extension to the ns-2 simulator and models accurately the DC's network traffic patterns. Although all of the aforementioned tools are able to calculate the power consumption of a DC, none of them fitted our needs (or were difficult to extend) regarding the key aspects of our approach: the calendars, the treatment of EP requests, elasticity in the resource allocation, and the economic costs involved. Thus, to evaluate the feasibility of our proposal, we developed our own simulator of the DC-EP ecosystem.

### 7.1. Our simulator: inputs, outputs and internal behaviour

Our simulator represents a DC by a collection of physical servers and simulates the execution of IT services placing some workload on them. Workload varies in two dimensions. First, the number of VM instances changes over time: there is a substrate of some number of *always-on* VMs (which may represent IaaS-type services or entire physical servers), plus a variable number of *time-scoped* ones, of distinct durations arriving spread in time. Second, the CPU demand placed by each at every time slot (i.e. our  $d_n$  in §6.1) varies according to some pattern. Slot durations are configurable. To emulate *GreenSLAs* and calendars, a file specifies, for each IT service, the min/max performance levels required at every slot, in regular regime (RC calendar) or *energy-saving* (ESC calendar), to apply for some time when a DC accepts to cut its consumption on an EP request. When that occurs, the tool can also simulate the effect of delaying tasks (e.g. the execution of backups or antiviruses) or the start of VMs, as additional measures.

Calendars' min/max values correspond to *consolidation ratios*  $C_R$ . Thus, a  $\min C_R = 4$  (of a certain server type) means that a VM should get  $\frac{1}{4^{\text{th}}}$  of a core; i.e. it should be granted  $a_n = 25\%$  of its capacity. The simulator employs these values to *assign services to servers* (as discussed next) and thus determine: the overall number of servers required, a performance score for each service at each slot (considering  $d_n$  and

the utilization  $u_n$  actually observed after the allocation), the overall CPU load of each server, its energy expenditure, and the energy apportioned to each service. To that end, the simulator models servers' power-utilization curves as per (1), and computes servers' power/energy expenditures per slot,  $W_n$  from their overall utilizations (equal to the sum of the  $u_n$  of the services hosted). Similarly, the energy spent by a service is derived from the  $u_n$  as discussed in §5.6.

To be as realistic as possible, server's parameters ( $W_{\min}$ ,  $W_{\max}$  and  $W_{\text{idle}}$ , and number of cores) were chosen from data in [9] for several platforms. In particular, the three types of servers used in the simulations were IBM x3250 M3, HP ProLiant DL360, and Dell PowerEdge R720. To account for other energy expenditures (e.g. cooling, network) we assumed that these spent power proportionally to servers, according to PUE=2. Note that this has no effect for comparison purposes.

The simulator underwent an extensive validation and tuning phase where its output was contrasted against the power/energy consumption measured in two DC testbeds with real servers and realistic load patterns. This revealed that the simulator can accurately reproduce the results found in real equipment.

### 7.2. Simulating service-to-server assignment

The simulator implements several strategies regarding VM placement for comparison purposes. As said, calendars' min/max values refer to consolidation ratios,  $C_R$ . From the  $C_R$  values of each VM, the simulator assigns VMs to servers by ordering  $C_R$  in ascending order and assigning them to the first server they fit in. This first-fit approximation to the bin-packing problem is not optimal and may require more servers than those strictly needed. However, it is easily implemented and known to require a number of servers below twice the number of the optimal case [43]. As the same method is used with and without our proposal, we expect this to hardly influence the comparison.

### 7.3. Simulating elasticity

To simulate elasticity (i.e. the ability of a DC to adapt resources according to services' demands), the simulator proceeds as follows. Note that, while in our discussion in §6.1, we spoke of demands  $d_n$  (e.g. CPU cycles), these may not be measurable metrics, particularly in workload non-conservative services. Instead, a DC may be able to monitor only the extent to which resources are utilized ( $u_n$ ) at each slot. Thus, our simulator tries to scale VM's  $C_R$  according to their observed CPU utilization levels  $u_n$ . However, since the burstiness of  $u_n$  samples may exhibit high variations from slot to slot, we compute a moving average of each service utilization as  $\hat{u}_n = \frac{\hat{u}_{n-1}W + u_n}{W+1}$  to filter out high frequency components. The smoothness (and reactivity) of this average is controlled by  $W$ , which defaults to 10 slots. Moreover, in practice, migrating/consolidating VMs very frequently may impose high penalties and is thus unrealistic. To account for this, the simulator is forbidden to adapt a VM's  $C_R$  more frequently than every  $K$  slots (e.g. 30 minutes).

Given these considerations, elasticity is implemented by comparing services'  $\hat{u}_n$  as follows. Recall that there is a 1-to-1 correspondence between consolidation factors  $C_R$  and shares

$U$  of CPU cores ( $U = \frac{1}{C_R}$ ) allocated to VMs. Thus, we keep a vector of feasible  $C_R$ 's and the corresponding vector of core shares or *levels*  $U = (U_1, U_2, U_j, \dots, U_h)$ , where the  $C_R$  vary at a granularity of 0.2 (i.e. 1, 1.2, 1.4, 1.6, etc.) and the  $U_j$  are in descending order. Let  $j$  be the index of the current level. If  $\hat{u}_n \geq 0.9U_j$ , we promote the service to level  $U_{j-1}$  (higher share; lower  $C_R$ ). If  $\hat{u}_n \leq 0.8U_{j+1}$ , we demote it to  $U_{j+1}$  and thus a larger  $C_R$ . If neither occur, the service retains its  $C_R$ .

Note that the above defines a *hysteresis* cycle and that we enforce such adaptation to respect the constraint that, at each slot  $n$ ,  $\min C_R(n) \leq C_R(n) \leq \max C_R(n)$ , as dictated by the RS and ESC calendars.

## 8. Experimental results

Next, we discuss some of the results obtained with our simulator. We start by discussing the setups and strategies simulated for comparison purposes.

### 8.1. Profiles, calendars and workloads

We considered two different profiles in our simulations. The first, **P1**, was derived from a real trace captured from a subset of customers of a commercial DC, corresponding to on-demand VM requests as typically offered by cloud IaaS services. Originally, the data corresponds to a 48 hour period (including a working day and a Saturday), during which the request time, the duration and the size of the VMs (small VM = 1 VCPU, medium VM = 2 VCPUs) were measured. The peak values were 112 simultaneous VM instances and 168 VCPUs. We extended the original data to a week by replicating the patterns of working days and weekends. We also artificially added a base of 200 always-on small VMs to simulate IaaS services<sup>4</sup>. A second workload profile, **P2**, was produced by randomly permuting the start time and duration of the VMs in P1, to study the influence of the distribution of the VM arrival time. Each IT service in these profiles obeys the calendars in table 8.1.

	Mo.-Sat. (9:00 AM - 21:00 PM) (busy hours)	otherwise non-busy
RC	$1 \leq C_R \leq 2$	$1 \leq C_R \leq 4$
ESC	$3 \leq C_R \leq 6$	$3 \leq C_R \leq 6$

Table 1: Performance levels defined in the calendars

At each slot, every VM demands a certain computational power, according to the following *workloads*. All of them refer to the CPU capacity of a single core: **W1**) 50% load plus two sinusoids with frequencies of 60 and 10 timeslots (minutes), with amplitudes of 30% and 20% respectively, and uniformly distributed phases (to avoid synchronization effects); **W2**) same as W1 during busy periods, but equal to  $0.6 \cdot W1$  during non-busy periods; and **W3**) uniform random demand in the range 25%-100% per slot, factored by 0.6 during non-busy periods.

<sup>4</sup>Although the total volume is not representative of a real-life DC, the key point is to capture the dynamics of the VM arrival. The traces can be scaled to higher volumes.

The energy rate  $r$  was considered constant throughout, whether the DC employed the RS or the ESC calendar.

### 8.2. Summary of results and their discussion

Figure 3 shows the evolution of the power consumed by our simulated DC during 7 days (measured at 1 minute slots) in several cases, all under profile P1 and workload W1. Case  $C_R = 1$  corresponds to the case where each VM was assigned to 1 core throughout. Consistently, power consumption is greatest in this case. Case *ideal* corresponds to a fictitious case where the DC exactly knew the amount of CPU power demanded by VM and could perfectly assign VMs to servers, freely at every slot. The power curve is lower in this case and it is the ideal as the scheme guarantees that all services' demands are coursed and thus a performance score of 100% –as shown by the bottom plot– at the minimum energy expenditure.

Curve *minRC* corresponds to assigning VMs to servers such that their minimum performance, as per the RC calendars, is guaranteed. In that case, no elasticity is attempted. Thus, the energy savings are greatest, but performance highly degraded. Case *elastic* is the one that would be obtained with elasticity, employing the hysteresis-based adaptation of §7.3 according to utilizations  $\hat{u}_n$ , and the limit of  $K$  slots regarding the frequency of  $C_R$  adjustments, but disregarding the calendar bounds. Case EEACD corresponds to the power consumed under the scheme that we propose; that is, when the DC adjusts VM's  $C_R$  as in the previous case, but respecting the calendar bounds in table 8.1. As can be seen, EEACD achieves a reduction in the power consumption comparable to that of the ideal case and an average performance score around 90%.

The shaded interval of time represents a 6-hour period (starting on Fri. 2:00 PM) where the DC switched to ESC calendars upon a request by its EP to reduce energy consumption. Without elasticity (min ESC), the DC achieved maximum power reduction at the expense of a large performance degradation. Case EEACD corresponds to our scheme where elasticity was employed but according to the ESC calendar. Note that, at the end of the shaded region, the power exhibits a peak. This is because, as an additional measure, we delayed the starting execution time of VMs arriving during the EP request window.

Figure 4 shows the results for the case when CPU demand is reduced by a factor 0.6 during non-busy hours (i.e. workload W2). The result illustrates that if the calendars are well chosen (in that they adapt to demands) then similar power reductions may be achieved while keeping high performance scores.

Figure 5 quantifies the energy saving and performance degradation, relative to the reference case  $C_R=1$  over the whole week and only during the EP request interval, with each of the schemes. The figures show that the strategies that save more energy degrade the performance excessively, and vice-versa, while the EEACD approach reaches a good trade-off among both. Notice that the savings with EEACD are higher than with the ideal case, because in the former, the system response is not instantaneous nor perfect, due to the adaptation delay and the inaccuracy of the moving average estimator of the utilization.

Note that attending an EP request greatly reduces the energy spent during the requested interval, but has a minimal effect

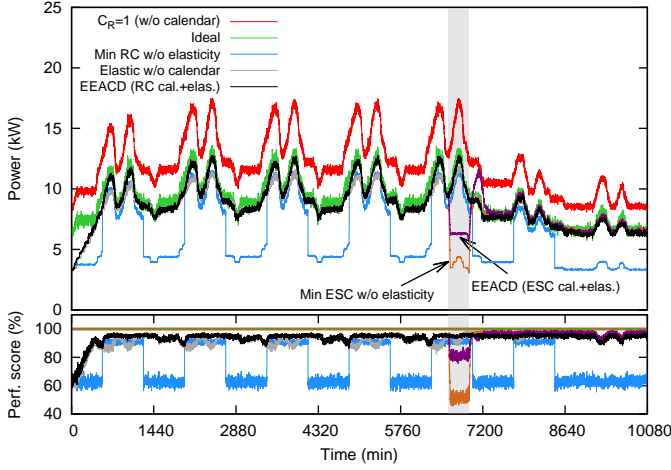


Figure 3: Top: Energy consumption of the DC for different power management strategies, with profile P1 and workload W1. Bottom: Mean relative performance of the IT Services.

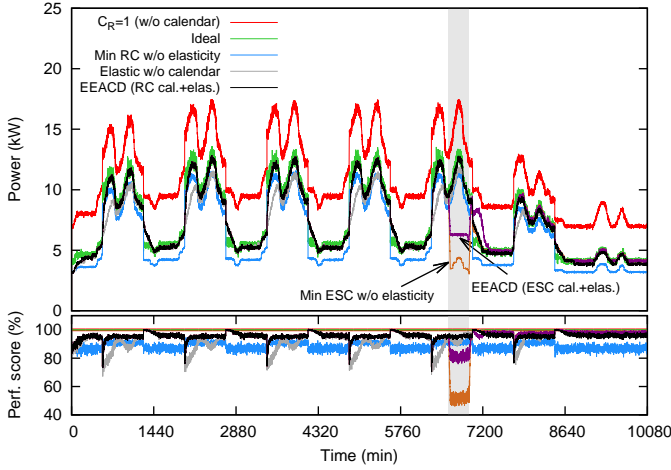


Figure 4: Top: Energy consumption of the DC for different power management strategies, with profile P1 and workload W2. Bottom: Mean relative performance of the IT Services.

on the weekly consumption. An important observation is that energy reduction outweighs the performance degradation. For instance, a 30% weekly energy reduction is achieved (top plot), at a mean performance degradation of 5% (bottom).

With workload W2, energy savings can be superior over the whole week as well as during the EP requests, as the demands are lower and follow the calendars more closely. Consistently, the performance score is also better.

Figure 6 quantifies the influence of workload patterns on the energy savings and the performance of services. We compare here the periodical (P1) and random (P2) arrival profiles (which have the same total amount of services, but distributed differently along the week) when combined with the sinusoidal (W2) and uniform (W3) workloads, both adjusted to a maximum 60% load during non-busy periods. The main conclusion here is that the service arrival pattern has a small influence, while the CPU load is much more influential. Again, in all the cases, the energy reduction outweighs the performance degradation.

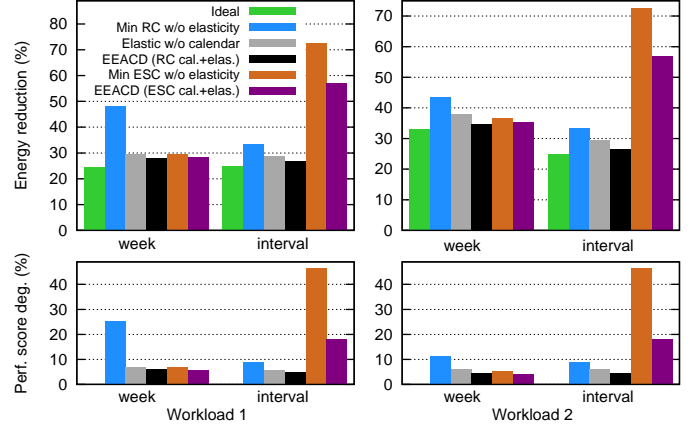


Figure 5: Energy saving (top) and performance reduction (bottom) for profile P1 and workloads W1 (left) and W2 (right).

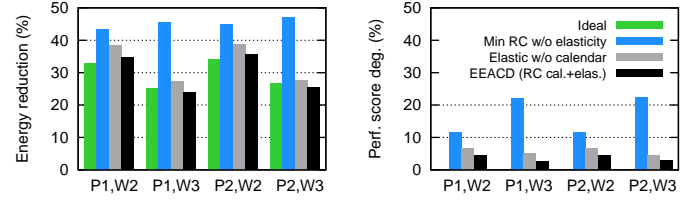


Figure 6: Energy saving (left) and performance reduction (right) for the combinations of profiles P1 and P2 and workloads W2 and W3.

We conducted experiments changing servers' properties. In many of the datasets we considered from [9], we saw that most servers  $b = \frac{W_{min}}{W_{max}}$  ratios ranged around 0.2. The number of cores affects the number of servers needed to satisfy the demands (assuming the same type of cores in all), as can be seen in Figure 7-top, which shows the number of servers required to course VM workloads. Note that the performance does not depend on servers' power properties. While the specific energy savings depend on the power curves considered, similar energy savings (in relative terms) are observed in all cases. The results with servers with 16 cores are slightly worse since the model considered was less energy proportional.

Figure 8 shows several things. The leftmost scatterplots show how the CPU utilization achieved relates to that demanded, for two services (VM1 and VM2). Each dot corresponds to a  $(d_n, u_n)$  pair. VM1 is an always-on VM and thus has activity in busy and non-busy hours. During busy hours,  $minC_R = 2$ . Thus it has 50% of a core guaranteed. When  $d_n > 50\%$ , the utilization depends on the availability of CPU cycles and the elasticity of the scheme to adapt to the demand. Thus,  $u_n \leq d_n$ . A similar effect is seen during non-busy hours, where  $minC_R = 4$ . VM2 is only active during busy hours, achieving  $u_n = d_n$  when  $d_n \leq 0.5$ . The plots in the middle show, at every slot, the price per slot as a function of the CPU demanded. Prices are proportional to the energy apportioned to the VM in each slot. For clarity, we normalized them by  $\frac{W_{max}}{N_{umcores}}$ , which is the maximum amount of energy that can be spent by a VM given assuming that the active idle power  $W_{min}$  is also equally divided by the number of cores. The relation is similar to that of the  $u_n, d_n$

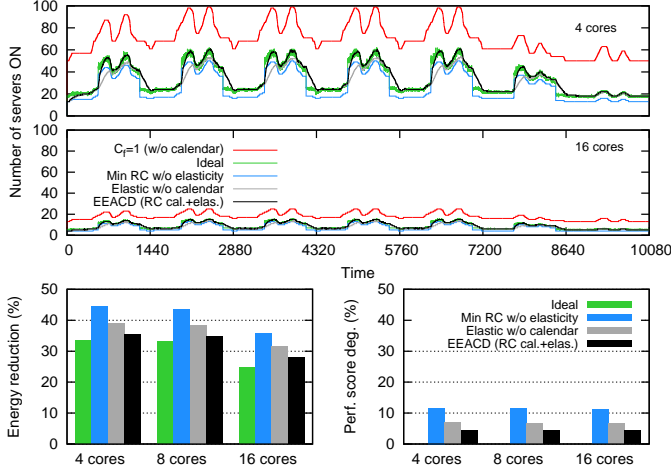


Figure 7: Top: number of active servers during the week, for the arrival pattern P1 and workload W2, for three models of servers with 4, 8 and 16 cores each. Bottom: Energy saving (left) and performance reduction (right).

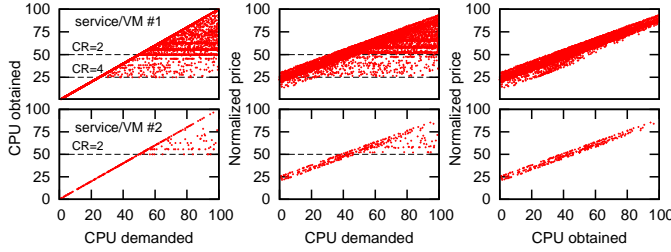


Figure 8: Scatterplots showing the relations among services utilization  $u_n$ , demands  $d_n$  and prices, per slot, for two sample services.

pairs, except for a fixed offset. This is confirmed by the right-most plots, which, consistently, show a linear correlation between price and utilization. Thus, the scheme is fairer than in the case that a fixed fee were employed. In this regard, note that we do not depict the fixed fee  $f$  per slot. Its inclusion would shift prices up by a constant offset. Obviously, if the fixed components  $f$  overexceed the energy costs, variations in the latter would not compensate the performance reduction. However, we expect the case to be the opposite.

Lastly, fig.9 shows, in the EEACD strategy, how prices and utilization relate *on the average* for each IT Service, in both P1 and P2 profiles, workloads W2 and W3. Note that, while the demand pattern determines the ranges on both axes, a linear correlation is observed.

## 9. Summary and conclusions

Advances in server and virtualization technologies, along with efficient workload consolidation methods, have greatly contributed to the reduction of the energy consumed by DCs. A step ahead toward a higher energy efficiency and carbon footprint reduction can consist, as recent work suggests, in a concerted approach where DCs adapt their power demands according to a demand-response scheme with their EPs and the performance of IT services is modulated, in a customer-consent

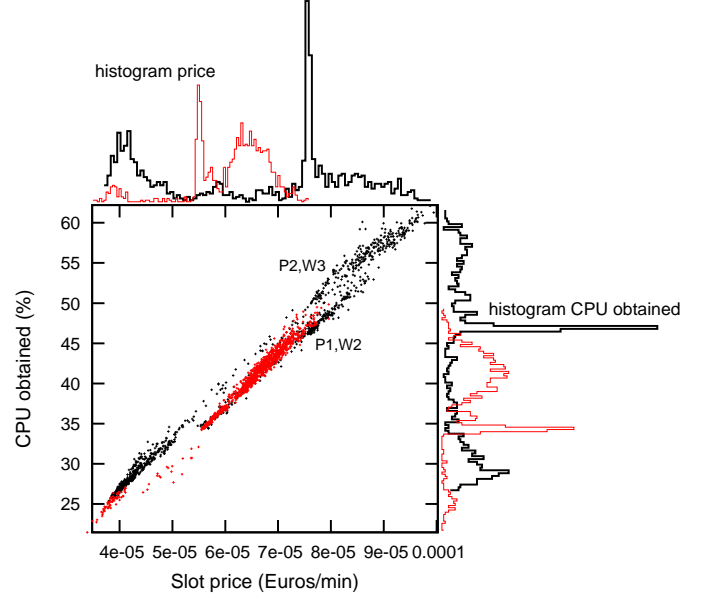


Figure 9: Price per IT service as a function of its average utilization.

manner, according to workload fluctuations, time of day, and the energy EPs can supply.

In this paper we argue that, while such a paradigm seems promising and the benefits to EPs are clear (as it can help EPs mitigate demand peaks and avoid shortages without resorting to non-renewable sources, or fit renewable ones in the energy mix), it will hardly be adopted by DCs or be well-perceived by customers unless some evidence exists that their economic interests are preserved. To overcome this, we propose a service model and pricing scheme that can be attractive for all the parties and is technically realizable given today's monitoring capabilities and the advances of elastic computing.

The idea behind our service type and pricing scheme is simple: dynamically adjust resources (thus power expenditure) depending to services' needs (load), performance bounds (calendars) and the availability of energy; and let customers pay according to the energy spent. By linking energy costs with service tariffs, customers may see their bills lowered while experiencing similar performance due to the elasticity. On the other hand, DCs can retain their profit, better plan their infrastructure (due to the calendars), reduce financial risks, and even enlarge their market share (given the advantages from the customer perspective), while still striving for energy efficiency and a tighter collaboration with EPs in order to offer attractive prices and stay competitive. Thus, the interests of all parties are protected while achieving higher energy efficiency.

Our simulation study suggests that customers may enjoy a performance similar to that in case they were given strict resource guarantees, but at a lower cost and overall power expenditure. Although the simulated scenario focuses on a cloud-computing environment and the performance score refers to CPU utilization (when other factors may influence), we note that the proposed service type is itself independent of the type of applications and energy-saving strategies used by DCs. Nat-

urally, it may not suit all types of applications, and its utility depend on the effectiveness of energy-saving measures. Nevertheless, we envisage that future advances in elastic computing will allow for a rapid and accurate provisioning of applications in multiple dimensions (CPU, memory, bandwidth), in which case the performance experienced by applications may be smoothly adapted, and our proposed service type suit a wide range of IT services.

## 10. Acknowledgements

This work was supported by the European Commission's FP7 programme (Objective ICT-2011.6.2, 28867 - All4Green Project) and by the Spanish government and FEDER under project TEC2013-47960-C4-1-P.

## References

- [1] Gartner Inc., Gartner estimates ict industry accounts for 2 percent of global co2 emissions., <http://www.gartner.com/it/page.jsp?id=503867> (2007).
- [2] J. Koomey, Growth in data center electricity use 2005 to 2010, Tech. rep. (2011).
- [3] C. Ren, D. Wang, B. Urgaonkar, A. Sivasubramaniam, Carbon-aware energy capacity planning for datacenters, in: IEEE 20th International Symposium on Modeling, Analysis Simulation of Computer and Telecommunication Systems (MASCOTS), 2012, pp. 391–400.
- [4] A. Berl, E. Gelenbe, M. Di Girolamo, G. Giuliani, H. De Meer, M. Q. Dang, K. Pentikousis, Energy-efficient cloud computing, *Comput. J.* 53 (7) (2010) 1045–1051.
- [5] D. Rincon, A. Agusti-Torra, J. F. Botero, F. Raspall, D. Remondo, X. Hesselbach, M. T. Beck, H. de Meer, F. Niedermeier, G. Giuliani, A novel collaboration paradigm for reducing energy consumption and carbon dioxide emissions in data centres, *Comput. J.* 56 (12).
- [6] Cisco Systems, Enterprise automates utility demand response, [http://www.cisco.com/c/en/us/products/collateral/physical-security/network-building-mediator/case\\_study\\_c36-543499.html](http://www.cisco.com/c/en/us/products/collateral/physical-security/network-building-mediator/case_study_c36-543499.html).
- [7] S. Klingert, A. Berl, M. Beck, R. Serban, M. di Girolamo, G. Giuliani, H. de Meer, A. Salden, Sustainable energy management in data centers through collaboration, in: *Energy Efficient Data Centers*, Vol. 7396 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2012, pp. 13–24.
- [8] S. Sams, Discovering hidden costs in your data centre - a cfo perspective, Tech. rep. (2010).
- [9] Standard Performance Evaluation Corporation, <http://www.spec.org>.
- [10] J. Bruschi, et al., Best practices guide for energy-efficient data center design, <http://datacenters.lbl.gov/sites/all/files/eedatacenterbestpractices.pdf> (2011).
- [11] Amazon Elastic Compute Cloud (EC2), <http://aws.amazon.com/ec2/>.
- [12] R. Dines, Build or buy? the economics of data center facilities, <http://www.io.com/wp-content/uploads/2013/04/build-or-buy.pdf> (2011).
- [13] D. Meisner, B. T. Gold, T. F. Wenisch, Powernap: Eliminating server idle power, Vol. 37, ACM, New York, NY, USA, 2009, pp. 205–216.
- [14] J. Leverich, M. Monchiero, V. Talwar, P. Ranganathan, C. Kozyrakis, Power management of datacenter workloads using per-core power gating, *IEEE Comput. Archit. Lett.* 8 (2) (2009) 48–51.
- [15] C. Ghribi, M. Hadji, D. Zeglache, Energy efficient vm scheduling for cloud data centers: Exact allocation and migration algorithms, in: 13th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid), 2013, pp. 671–678.
- [16] R. Birke, A. Podzimek, L. Chen, E. Smirni, State-of-the-practice in data center virtualization: Toward a better understanding of vm usage, in: 43rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2013, pp. 1–12.
- [17] B. Kantarci, L. Foschini, A. Corradi, H. Mouftah, Inter-and-intra data center vm-placement for energy-efficient large-scale cloud systems, in: *IEEE Globecom Workshops*, 2012, pp. 708–713.
- [18] C. Isci, S. McIntosh, J. Kephart, R. Das, J. Hanson, S. Piper, R. Wolford, T. Brey, R. Kantner, A. Ng, J. Norris, A. Traore, M. Frissora, Agile, efficient virtualization power management with low-latency server power states, *SIGARCH Comput. Archit. News* 41 (3) (2013) 96–107.
- [19] Z. Jia, L. Wang, J. Zhan, L. Zhang, C. Luo, Characterizing data analysis workloads in data centers, in: *Proceedings of IEEE International Symposium on Workload Characterization, IISWC*, 2013, pp. 66–76.
- [20] C. Wang, B. Urgaonkar, Q. Wang, G. Kesidis, A. Sivasubramaniam, Data center cost optimization via workload modulation under real-world electricity pricing, in: *IEEE/ACM 6th International Conference on Utility and Cloud Computing*, 2013, pp. 260–263.
- [21] B. Aksanli, T. Rosing, E. Pettis, Architecting efficient peak power shaving using batteries in data centers, in: *Proceedings of the IEEE 21st International Symposium on Modeling, Analysis Simulation of Computer and Telecommunication Systems (MASCOTS)*, 2013, pp. 1–12.
- [22] Y. Yao, L. Huang, A. Sharma, L. Golubchik, M. Neely, Data centers power reduction: A two time scale approach for delay tolerant workloads, in: *Proceedings of IEEE INFOCOM*, 2012, pp. 1431–1439.
- [23] C. Dupont, G. Giuliani, F. Hermenier, T. Schulze, A. Somov, An energy aware framework for virtual machine placement in cloud federated data centres, in: *Proceedings of the 3rd International Conference on Future Energy Systems: Where Energy, Computing and Communication Meet, e-Energy*, 2012, pp. 1–10.
- [24] Rackspace, <http://www.rackspace.com/>.
- [25] Flexiscale, <http://www.flexiscale.com/>.
- [26] HP, Public cloud services, <http://www.hpcloud.com/products-services>.
- [27] LunaCloud, Cloud server and cloud storage services, <http://www.lunacloud.com/>.
- [28] Google cloud platform, <https://cloud.google.com>.
- [29] Cloud service providers list 2014, [http://www.spamina.com/eng/cloud\\_hosting\\_providers\\_list.php](http://www.spamina.com/eng/cloud_hosting_providers_list.php).
- [30] L. Barroso, U. Holzle, The case for energy-proportional computing, *Computer* 40 (12).
- [31] International Organization for Standardization, ISO 14064-1:2006 Greenhouse gases – Part 1: Specification with guidance at the organization level for quantification and reporting of greenhouse gas emissions and removals (2006).
- [32] CarbonTrust, <http://www.carbontrust.com/>.
- [33] M. Wenk, *The European Unions Eco-Management and Audit Scheme (EMAS)*, Springer, 2005.
- [34] L. Cherkasova, R. Gardner, Measuring cpu overhead for i/o processing in the xen virtual machine monitor, in: *Proceedings of the Annual Conference on USENIX Annual Technical Conference, ATEC '05*, 2005, pp. 24–24.
- [35] C.-C. Lin, P. Liu, J.-J. Wu, Energy-efficient virtual machine provision algorithms for cloud systems, in: 2011 Fourth IEEE International Conference on Utility and Cloud Computing (UCC), 2011, pp. 81–88.
- [36] H. Liu, C.-Z. Xu, H. Jin, J. Gong, X. Liao, Performance and energy modeling for live migration of virtual machines, in: *Proceedings of the 20th International Symposium on High Performance Distributed Computing, HPDC '11*, 2011, pp. 171–182.
- [37] Px intelligent rack pdus, <http://www.raritan.com/products/power-distribution/intelligent-rack-pdus>.
- [38] A. Kansal, F. Zhao, J. Liu, N. Kothari, A. A. Bhattacharya, Virtual machine power metering and provisioning, in: *Proceedings of the 1st ACM Symposium on Cloud Computing, SoCC '10*, 2010, pp. 39–50.
- [39] A. Lenk, M. Menzel, J. Lipsky, S. Tai, P. Offermann, What are you paying for? performance benchmarking for infrastructure-as-a-service offerings, in: *IEEE International Conference on Cloud Computing (CLOUD)*, 2011, pp. 484–491.
- [40] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose, R. Buyya, Cloudsim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms, *Softw. Pract. Exper.* 41 (1) (2011) 23–50.
- [41] K. Kurowski, A. Oleksiak, W. Pitek, T. Piontek, A. Przybyszewski, J. Wglarz, Dcworms: A tool for simulation of energy efficiency in distributed computing infrastructures, *Simulation Modelling Practice and*

- Theory 39 (0) (2013) 135 – 151.
- [42] D. Kliazovich, P. Bouvry, S. Khan, Greencloud: a packet-level simulator of energy-aware cloud computing data centers, *The Journal of Supercomputing* 62 (3) (2012) 1263–1283.
  - [43] V. V. Vazirani, *Approximation Algorithms*, Springer, 2003.

## Author Biographies

**Anna Agustí Torra** received her M.Sc. degree (2001) and her Ph.D. degree (2009) in Telecommunication Engineering from the Universitat Politècnica de Catalunya – Barcelona Tech (UPC). In 2001, she joined the Department of Telematics Engineering at UPC, where she is currently an Assistant Professor and member of the Broadband Networks and Services (BAMPLA) research group. Her research interests include optical networks, dynamic resource allocation mechanisms, performance engineering, and energy-aware techniques applied to networks and services.

**Frederic Raspall** received a M.Sc. (Feb.'01) and a Ph.D. (Jul.'09) degree in telecommunication engineering from the Universitat Politècnica de Catalunya – Barcelona Tech (UPC). From 2000 to 2004, he was with NEC Network Laboratories Heidelberg (Germany), where he developed network prototypes and participated in several research projects funded by the European Union. Since Sept. 2004, he has been a lecturer at two engineering schools within the UPC, where he has participated in research projects funded by the EU and the Spanish Ministry of Science and Innovation. His current research interests include network measurements, algorithms, estimation problems, and energy-awareness.

**David Rincón** received a M.Sc. in telecommunication engineering and a Ph.D. (2007) degree in Computer Networks from the Universitat Politècnica de Catalunya – Barcelona Tech (UPC). In 1998 he joined the Department of Telematics Engineering at UPC, where he is currently a Lecturer and member of the Broadband Networks and Services (BAMPLA) research group. He was a visiting researcher at the Teletraffic Research Centre (University of Adelaide, Australia, 2007) and at the Institute of Pure and Applied Mathematics (IPAM) at UCLA (2008). His interests include network traffic modelling, multi-resolution analysis techniques, cloud computing, and energy consumption in communication systems.

**David Remondo** holds a Tenured Associate Professorship at the Department of Telematics Engineering of the Universitat Politècnica de Catalunya – Barcelona Tech (UPC), Spain. He graduated in physics, with a major in electrical engineering, at the Universidad de Cantabria, Spain, in 1990, and he obtained a Master in Electronics Engineering at the Technische Universiteit Eindhoven, the Netherlands, in 1992. Until 1995, he was a research engineer with CEIT, in Spain. He obtained his PhD in the area of telecommunications from the Universiteit Twente, the Netherlands, in 2000. Dr. Remondo held an Assistant Professorship at the Computer Networks chair of the same university since 1999 until 2001. Until 2006 he was a



Ramón y Cajal fellowship holder at UPC. David Remondo has published 12 full papers in international research journals and has participated in more than fifteen evaluations of research projects co-funded by the European Commission. His research interests lay in the area of communication networks, mainly in heterogeneous networks, energy saving issues and performance evaluation.

**Giovanni Giuliani** graduated in Electronics Engineering with honors in 1982, and has over 30 years of working experience in R&D, IT Consulting and System Integration in major Computer Companies (Olivetti, Digital, Compaq, HP). He has covered various positions in SW Engineering until 1997, working on Operating System development and Manufacturing Framework products; he has been nominated Consulting Engineer by Digital Corporate Engineering Board in 1992. In parallel to his job in the industry, Giovanni performed university teaching and research activities from 1988 to 1993 at Engineering and Computer Science universities in Italy (Milan and Como). Starting from 1998 - as Solution Architect in Consulting and System Integration organization in Digital, and then, due to the mergers, Compaq and finally HP – he delivered several complex projects in Manufacturing, Government, Finance market, then exploring the area of Mobility for various sectors. Back in 2005 he joined the HP Italy Innovation Center, where he's been leading the Cloud Computing Initiative since 2009. Starting from 2013 he serves as Lead Architect of the HP LIFE platform (HP worldwide program offering a cloud based e-learning initiative for entrepreneurs). During the last decade, he has been involved in several research projects funded by European Commission in the area of Mobility, e-Collaboration, Cloud Computing and more recently he is technical coordinator in a sequence of 3 Green-IT research projects (FIT4Green, All4Green and now DC4Cities).