Estimation and forecasting of ecological efficiency of Virtual Machines

Gregory Katsaros[c], Pascal Stichler[c], Josep Subirats[a,b], Jordi Guitart[b,a]

[a] *Barcelona Supercomputing Centre (BSC), 08034 Barcelona, Spain*
[b] *Universitat Politecnica de Catalunya - Barcelona Tech (UPC), 08034 Barcelona, Spain*
[c] *FZI - Research Center for Information Technology, Berlin, Germany*

Abstract

The massive development of the cloud marketplace is leading to an increase in the number of the Data Centers (DCs) globally and eventually to an increase of the CO_2 related footprint. The calculation of the impact of Virtual Machines (VMs) on the environment is a challenging task, not only due to the technical difficulties but also due to the lack of information from the energy providers. The ecological efficiency of a system captures the relationship between the performance of the system with its environmental footprint. In this paper we present a methodology for the estimation and prediction of the ecological efficiency of VMs in private cloud infrastructures. We specifically focus on the information management starting from the energy resources in a region, the energy consumption and the performance of the resources and finally the calculation of ecological efficiency of a VM. To this end, we have designed and implemented a framework through which the ecological efficiency of a running VM can be assessed and the ecological efficiency of a VM to be deployed can be forecasted. The presented framework is being evaluated through several private cloud scenarios with VM deployments in hosts located in Germany.

Keywords: Ecological efficiency, cloud computing, Virtual Machine, energy consumption, monitoring

1. Introduction

The rapid growth of ICT application services goes along with an increase in number and size of data centers (DCs) that host these services. Because data centers are massive energy consumers, the carbon footprint of application services is moving more and more into focus. It is considered that ICT presently accounts for approximately 2% of global carbon emissions (more than 830 million tons of carbon dioxide) [1][2]. The advent of the cloud computing paradigm

Email addresses: katsaros@fzi.de (Gregory Katsaros), stichler@fzi.de (Pascal Stichler), josep.subirats@bsc.es (Josep Subirats), jordi.guitart@bsc.es (Jordi Guitart)

gave an enormous boost to the ICT services sector and the prediction towards 2020 is that the market (SaaS, PaaS, IaaS) will quadruple from what it is today.

NESSI, in the Strategic Research and Innovation Agenda for 2013 [3], clearly defined research priorities and recommendations: we must find new ways to increase software performance and energy-efficiency, by engineering energy-aware software to improve power-efficiency of software systems and services. Apart from the pure environmental and technological perspective, one should consider the current market demands. There is a significant part of the consumers community who enjoys above-average income and spending, and their buying decisions reflect environmental considerations [4]. The market of green consumers has an estimated global potential of 200 billion to 400 billion. Approximately 10% of German consumers and 12% of U.S. consumers are willing to pay 10% more for "green" products that require less energy to operate or are manufactured by companies with a "green" reputation. A recent report [5] also shows that 28% of IT decision-makers at companies in Europe, the Middle East and Africa consider environmental criteria to be very important when purchasing IT products and services. Another 67% of those surveyed attach relatively high importance to environmental criteria.

Until now the efforts of Green ICT in relation with cloud computing were restricted at the level of the Data Center and mainly in the form of increasing the energy efficiency of the infrastructures (PUE). While this activity represents only a small part of the cloud marketplace, mainly among the IaaS providers, the challenging question is whether we can introduce ecologically friendly policies and concepts in the level of applications and service providers. In that context, the element that is the main building block of all modern, cloud-enabled applications is the Virtual Machine (VM). Thus, the starting point of a relevant study and investigation should be at the level of virtual infrastructure.

Ecological efficiency of a system is the amount of work that is delivered in relation to its CO_2 emissions. To assess the eco-efficiency of cloud-based services, it becomes increasingly important to investigate the eco-efficiency of the cloud resources that the service utilizes. To this end, there are several technical constraints and challenges [6]: virtualization technology, which is a major characteristic of cloud computing, introduces an abstraction layer between the consumers of cloud resources and the physical infrastructure. The energy consumed by a service or a VM cannot be directly metered and therefore must be estimated through certain modeling methodology. What is more, the CO_2 footprint of the cloud resources is directly related with the energy-mix that the respective data center consumes at the time of the VM operation. Considering that the needed energy is provided by the local power providers, the calculation of ecological efficiency in cloud computing is therefore a location and time relative figure.

The above-mentioned issues harden the effective calculation of ecological efficiency of cloud services. To ecologically evaluate the resources that a cloud-enabled application utilizes, providers require appropriate tools and methods that are still not there. Thus in this paper we aim at investigating the environmental impact of cloud resources and specifically estimate the ecological efficiency at the granularity of the VM. In particular, we propose and implement a methodology that allows assessing the ecological efficiency of running VMs in the provider and forecast the ecological efficiency of VMs to be deployed or in operation. This work constitutes an updated and significantly extended version of the concepts presented in [7].

As has been presented also in [8] there are many parameters that affect the cost (energy or financial) of cloud resources. We, though, selected to focus on the ecological efficiency of the VM as it is one very interesting variable parameter that directly relates performance with power consumption of cloud infrastructures, resulting in a highly dynamic metric. In our work, we have not considered the cost related with the static infrastructure (cooling of Data Center, management cost, etc.), because these figures can be calculated separately and added on top of the parameters that we calculate dynamically. In addition, we assume that contemporary cloud application topologies consist of multiple application components installed in different VMs. Therefore, the power consumption and eventually the ecological efficiency should be investigated at the level of VMs.

In the following sections we present the proposed methodology and corresponding implementation to calculate and forecast the ecological efficiency of a VM considering the location and time constraints of its operation. In Section 2 we present the state of the art and the related work in the respective fields of research. In Section 3 we elaborate on the proposed solution: a monitoring system that allows assessing and forecasting the eco-efficiency of VMs. We describe in detail the architecture of the solution as well as the theoretical methodology and concept.Finally, in Section 4 we proceed with the evaluation of the implemented solution with several deployment scenarios within Germany, while in Section 5 we conclude and summarize our findings.

2. State of the art and related work

In order to reduce the carbon footprint of a system or a service, first of all we need to monitor and analyze the performance as well as the energy related information of our computing infrastructure. In cloud computing, it is important to know how much energy a specific service or VM consumes, rather than the consumption of the whole physical infrastructure. However, measuring the energy consumption of a single or even several VMs is a challenging task. From a consumer point of view, VMs are a black box whose energy consumption can only be estimated [9]. In order to do so, power usage models are normally used where performance characteristics are being used for the modeling of the energy consumption. In [10] [11] [12] [13] it is pointed out that CPU utilization is the main factor driving energy consumption of a computing system, with memory and disk resource utilization playing a secondary role.

What is more, in [14] they make use of power usage metering to calculate and forecast the energy efficiency level of VMs in order to optimize VM deployments in private clouds. The methodology for the calculation of the energy efficiency of all cloud entities has been proposed using the CPU utilization of the VMs as the parameter to define the useful work performed and linear regression technique for the forecasting. To this end, there have been many discussions about measuring the useful work of a computer accurately [15]. Several benchmarks have been proposed [16], however, since every application has different requirements no universal formula can be determined. In the same context, power consumption modeling has been used for power-aware VM allocation using genetic algorithms [17] or through heuristic algorithms [18]. The definition of energy efficiency in cloud computing has been playing an important role also for the application of VM consolidation strategies [19] [20].

The term ecological efficiency is a rather generic description of the efficient use of ecological resources. In computing systems, eco-efficiency can be seen as computing power delivered compared to the environmental resources needed to do so. And again, in this context the complete product or service life-cycle and its related impacts on the environment have to be considered. In the context of cloud computing, there have been different approaches to define eco-efficiency. For example, Google publishes the carbon emissions per query (0.2 g CO_2) and per watched minute on YouTube (0.1 g of CO_2) [21]. Similarly, in [22] a charge-back model is presented, where the environmental impact of providing data center services to the service consumers is traced back to the consumer. The consumer receives information about the CO_2 intensity of each transaction as well as the overall CO_2 emissions produced by his transactions. The eco-efficiency is calculated by CO_2 emissions per data transaction executed on the service.

In [23], the authors have formulated the cost of VM migrations between private clouds aiming at the reduction of the carbon footprint of a cloud network. Furthermore, in [24] a routing methodology for user placement in data centers is presented, that generates minimum carbon footprint and therefore optimizes the ecological efficiency. Both research studies point out the significance of the geographic location of the cloud infrastructure towards the increasing of the eco-efficiency of a service deployment. In addition, in [25], a framework for optimizing the carbon efficiency in clouds is presented, which is based on the installation of a registry with offers from cloud providers including data about their CO_2 emission rate, average DCiE (Data Center infrastructure Efficiency), VM power efficiency, prices, etc.; all this information has to be updated by the provider though.

While the benchmarks described in [26] are used to measure the offline energy efficiency of a node or set of nodes, the algorithm described in [27] considers real-time metrics, but not the energy efficiency itself. It actually considers metrics which potentially affect the energy efficiency. In this publication, we evolve and combine the above-mentioned techniques concerning the modeling of the VM performance and the estimation of energy consumed to calculate the ecological efficiency in real-time. Both the delivered performance and power consumption are measured using real-time metrics and power-measuring devices. We also introduce a methodology for the mapping of energy consumption with the amount of CO_2 emitted and therefore calculate the ecological efficiency of VMs based on the location of deployment, the time of execution and the performance characteristics of each instance.

The ecological efficiency assessments and predictions presented in this publication can be used by placement algorithms to maximize the ecological efficiency of a given service and therefore reduce the CO_2 associated with it. For instance, [28] presents two task consolidation algorithms that decide the destination node for a batch application based on resource utilizations and power consumptions. In [29], the presented solution also migrates and consolidates VMs while maintaining the QoS as high as possible, in order to minimize the energy consumption. Similar approaches could be done by considering the potential ecological efficiency as the target to maximize, instead. The policy in [30] maximizes the provider's benefit by considering multiple facets to decide the placement of VMs: energy efficiency, virtualization overheads and SLA violation penalties. It also supports the outsourcing of VMs to external providers. In this context, the provider could also consider the possibility of outsourcing the execution of VMs to those providers providing a given level of ecological efficiency (it could be a requirement in the SLAs), which depends on the location of such providers. Finally, the authors in [31] propose an algorithm that conveniently schedules the execution of batch tasks in order to maximize the use of photovoltaic energy, which is locally produced in the datacenter. Indirectly, their algorithm tries to schedule the task execution when the ecological efficiency is at the maximum value. This publication also reflects the concern of current providers at trying to minimize their emissions while providing the desired QoS to their clients.

3. Monitoring ecological efficiency in clouds

Any type of energy optimization or improvement in the resource management of a system first of all requires energy information awareness. Thus, the goal of the cloud ecological efficiency monitoring system which we present in this paper is to calculate the eco-efficiency of the VMs taking into consideration their performance requirements as well as energy related information. The system includes two independent operations (Figure 1) that are necessary for the calculation of the eco-efficiency of each VM:

- Creation of the CO_2 Emission Registry: the amount of CO_2 generated when using a service is based on the energy resources mix (coal, wind, solar, nuclear etc.) and therefore is a location-specific (for example, 10kWh electricity produced in Athens results in 2kg of carbon emission, while in Berlin it results in 3kg of carbon emission) and a time-related (the energy-mix changes by the time) figure. The CO_2 Emissions Registry will maintain the emission factors (how much CO_2 is generated for the production of 1kWh electricity) per location and for every hour of the day. The required information will be gathered by utilizing public energy data streams (e.g. European Energy Exchange[1]) and historical data-stores, transformed and placed in a database from where they will become available to the rest services.

[1] http://www.eex.com/en/

- Calculation of the eco-efficiency: the system monitors the cloud infrastructure in terms of energy as well as performance metrics. The interaction with the private cloud infrastructure is done through a plug-in mechanism that can support multiple cloud middlewares. The implementation used during the evaluation in this article includes a plug-in for the VMWare ESXi cloud middleware which could be easily replaced with a different one for OpenStack, OpenNebula etc. The service can calculate the ecological efficiency of each VM in real-time or forecast it for a given VM to be deployed in the cloud infrastructure. The prediction of the ecological efficiency during monitoring allows for optimizing the VM placement process.

The first operation represents the data aggregation phase in which the data regarding the CO_2 emissions from the energy resources of all regions and countries is collected and analyzed. On the other hand, the second operation represents the monitoring phase in which the performance and energy data from the cloud infrastructure is gathered and the eco-efficiency of each VM is calculated or forecasted.

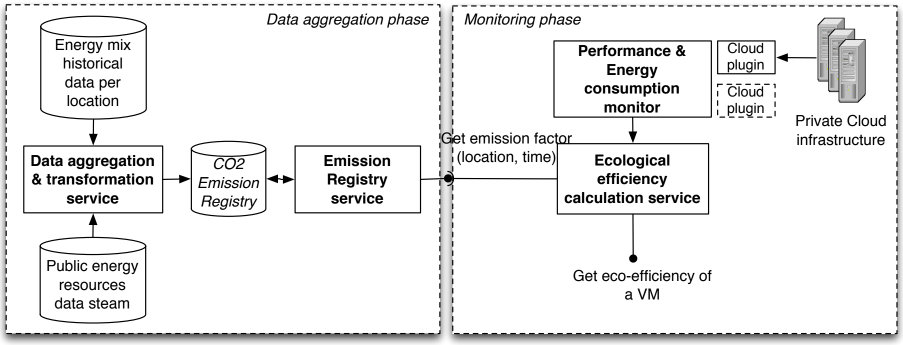
Data aggregation phase
Energy mix historical data per location
Data aggregation & transformation service
CO2 Emission Registry
Emission Registry service
Public energy resources data steam
Monitoring phase
Performance & Energy consumption monitor
Cloud plugin
Cloud plugin
Private Cloud infrastructure
Get emission factor (location, time)
Ecological efficiency calculation service
Get eco-efficiency of a VM

Figure 1: High level architecture of the ecological efficiency monitoring system for clouds

In the following subsections we will describe the design, implementation and operation of these two phases in detail.

3.1. Aggregation phase: Creation of the CO_2 Emission Registry

This phase aims at creating the CO_2 emission registry with region and time specific information about the carbon emission factor. As it has been pointed out in [32], the location of a DC is of great importance. A DC, and therefore the cloud services (IaaS, PaaS, SaaS, etc.) that it offers, is powered by energy sources of the local region. The long distance energy transfer is usually expensive and inefficient. Thus, contemporary DCs are being built close to the energy sources and specifically close to "green" energy sources. Examples are Facebook's and Spotify's new DCs in Sweden where there is a big percentage of

$E_{total}(region)$	Total amount of energy produced in a region
$E\_Share\_Local(resource, region)$	Share of energy produced from a resource in a region with regards to the total amount of energy produced in that region
$E_{country}(resource)$	Energy produced from a resource in a country
$E\_Share\_Country(resource, region)$	Share of energy produced from a resource in a region with regards to the total amount of energy produced in the country
$ERDData(resource, time)$	Hourly energy production from a resource in the country
$CO_2Factor(resource)$	Grams of CO_2 emitted to produce a kWh of energy from a resource
$CO_2EmissionFactor(region, time)$	Grams of CO_2 emitted to produce a kWh of energy in a given region and time
$EcEf$	Energy efficiency
VMs	Set of virtual machines, index j
VMs_i	Subset of virtual machines being executed in node i, index j
$U_{VM_j^i}$	CPU utilization of VM j in node i
$Perf_{max_i}$	Maximum performance of node i in Computing Units (CU)
R_i	Real power consumption of node i
$\widehat{U}_{VM_j^i}$	Forecasted CPU utilization of VM j in node i
$\widehat{R}_i$	Forecasted real power consumption of node i
$\widehat{U}_0^i$	Forecasted CPU utilization of Domain-0 of node i
$nodes$	Set of nodes, index i
$Perf_{VM_j}$	Current performance of VM j in Computing Units (CU)
Pwr_{VM_j}	Current power consumption of VM j
U_0^i	CPU utilization of Domain-0 of node i
n_i	Number of VMs being executed in node i
$EcEf_{VM_j}$	Ecological efficiency assessment of VM j
$\widehat{Perf}_{VM_j}$	Forecasted performance of VM j in Computing Units (CU)
$\widehat{Pwr}_{VM_j}$	Forecasted power consumption of VM j
$\widehat{EcEf}_{VM_j}$	Forecasted ecological efficiency of VM j
$\overline{Perf_{CPU}}$	Average performance per CPU in the whole IaaS provider infrastructure
$U_{CPU_{def}}$	Default initial utilization per CPU
CPU_{node_i}	Total number of CPUs in node i
N	Number of nodes of the whole Iaas provider infrastructure
Pwr_{min_i}	Minimum power consumption of node i (when U=0%)
$\overline{Pwr_{incr}}$	Average incremental power consumption per CPU in the whole IaaS provider infrastructure
Pwr_{incr_i}	Incremental power consumption per CPU in node i
CPU_{alloc_i}	Total number of CPUs allocated to VMs in node i
$\delta_{cert.}$	Boolean variable: equals to 1 if the provider owns all the certifications required in the Service Manifest
CPU_{VM_j}	Number of CPUs required by VM j

Table 1: Summary of symbols used in this section

renewable energy production. In the following analysis, we assume that regardless of the existence of different power providers or utilities, all the DCs in a region are consuming the same energy mix. Therefore, for simplifying the development of our methodology, we have eliminated the from our study the power providers entities and we consider the regional energy mix when calculating the ecological impact of a DC and therefore a VM.

The challenge in this task was to discover public data streams that provided energy production information per country, region and per resource in a reliable manner. To achieve that, we had to combine data from energy information providers such as the European Energy Exchange (EEX) with historical dataset depicting the energy mix of every local region. The first data source (EEX) offers a well defined API and a hourly refresh rate information stream but it gives out information in a per-counrty level. On the other hand, the historical datasets that we discovered present the energy mix rates for each region in Germany for the last three years. By associating those two sources of information we come up with a real-time energy analysis of every region. We collected, analyzed and transformed this data by introducing a certain data structure and transformation logic which allows calculating the CO_2 emission factors per region.

3.1.1. Data structure

In order to efficiently manage the available information we designed a data model that captures all the above mentioned information and their interrelations. The gray boxes in Figure 2 represent the information that is collected from different sources, which after certain processing and transformation is aggregated in the CO_2 Emission Registry. The energy resources that we considered are split into the following categories: Coal, Gas, Lignite, Oil, Uranium, Water/Wind, Photovoltaic (sun), Others. The resource category Others combines the energy produced from resources, not covered in the list, such as geothermal energy. In order to transform the energy mix to carbon emissions data, we need information about the CO_2 emissions associated with each energy category.

For the calculation of the different CO_2 emission factors we must also consider the complete life cycle of the respective power plants (construction, operation, maintenance, disposal). In the literature there have been different attempts to estimate these factors accurately. One common approach is the process chain analysis in which the energy production process is divided into different steps and for each step the input and output factors are calculated. Wagner et al. [33] have performed a comparison of the major methodologies and summarized them into a minimal reasonable, maximal reasonable and average CO_2 emission factor for each resource. In order to make all the emission factors (like CH_4, N_2O etc.) comparable, we transformed them into CO_2 equivalents. To do so, one unit of the respective gas is transformed into the amount of CO_2 which would have the same impact on the environment as one unit of that gas. In our work we adopted these factors as well as data from Lübbert [34] during the aggregation phase of our system.

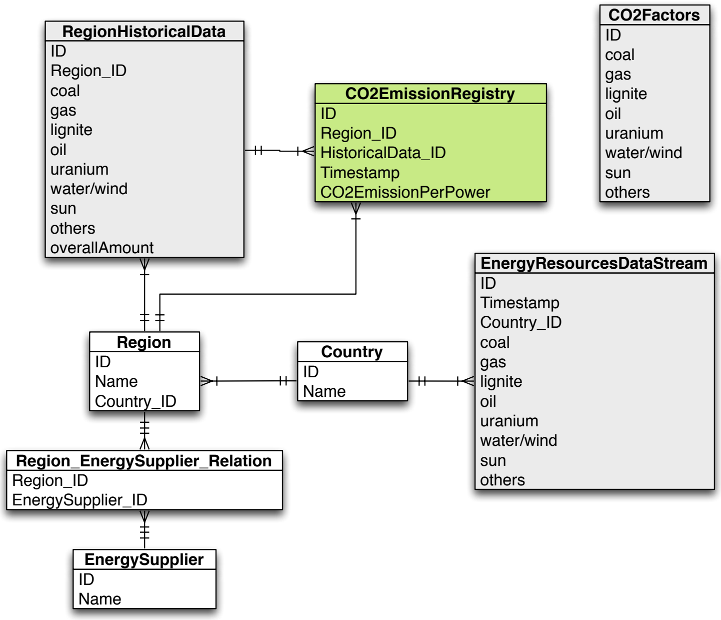

Figure 2: Data model that captures the information used during the aggregation phase.

3.1.2. Transformation logic

The objective of this section is to determine a region and time specific CO_2 emission factor. This factor indicates the amount of CO_2 emitted to the atmosphere in grams to generate a kWh of energy, in a given region for a specific time instant. The units for this factor are, therefore, $\left[\frac{g}{kWh}\right]$.The raw information from the data streams is transformed and combined to calculate this factor. The process is split into several steps, detailed below.

Firstly, we work with the historical datasets and we calculate the annual amount of energy produced from a given resource in the whole country (Germany, in our case). This is done by multiplying the share of the resource in the energy-mix of each region ($E\_Share\_Local(resource, region)$) with the overall amount of produced energy in that respective region ($E_{total}(region)$) and adding up the values from all the regions in the country. This calculation is presented in Equation 1.

$$E_{country}(resource) = \sum_{region \in country} E\_Share\_Local(resource, region) * E_{total}(region) \tag{1}$$

For example in 2009 in Baden-Württemberg, a region of Germany, 23 % of the energy was produced from coal ($E\_Share\_Local(coal, BaWu)$) and the overall amount of energy produced was 61.792 million kWh ($E_{total}(BaWu)$). Hence, the amount of energy produced from coal in Baden-Württemberg was 14.212 million kWh. Added up with the corresponding values of the other regions, the overall amount of power produced from coal was 99.673 million kWh in 2009 in Germany ($E_{Germany}(coal)$).

As a second step, we calculate the energy share for a particular region and resource (energy source) with regards to the overall energy production for that same resource in the whole country (calculated as in Equation 1). Note that while $E\_Share\_Local(resource, region)$ referred to the energy share for a particular region and resource with regards to the total energy produced in that region, $E\_Share\_Country(region, resource)$ calculates it with regards to the energy produced in the whole country from that same resource.

$$E\_Share\_Country(resource, region) = \frac{E\_Share\_Local(resource, region) * E_{total}(region)}{E_{country}(resource)} \tag{2}$$

Following the previous example, for Baden-Württemberg this is $E\_Share\_Germany(BaWu, coal) = \frac{14\ 212\ million\ kWh}{99\ 673\ million\ kWh} = 0.1426 = 14.26\%$. As a result, from all the power produced from coal in Germany, 14.26% is generated in Baden-Württemberg.

In the final step, the historical data is combined with the up-to-date data from the *EnergyResourcesDataStream* (ERDData [2]) to calculate the actual CO_2 emission factor for each region of a country. The ERDData provides information about the hourly energy production for a given resource in the whole country. By multiplying it by the historical country-wide energy share in power production from a specific resource for a region (Equation 2), we obtain the current amount of energy produced from this resource in that particular region. In order to transform it into actual CO_2 emissions, it is multiplied by the CO_2 emission factor associated to that resource (grams of CO_2 emitted to produce a kWh of energy from that resource). At this point, we have calculated the emissions caused by the "combustion" of one of the resources of the energy share of the region of interest. The emissions caused by each of the resources in the share are then summed up to get the overall emissions. Finally, to get the average CO_2 emission factor for the region, the overall CO_2 emissions of the region are divided by the power generation in that region.

$$\begin{aligned} &CO_2EmissionFactor(region, time) \\ &= \frac{\sum_{r \in resources} E\_Share\_Country(r, region) * ERDData(r, time) * CO_2Factor(r)}{\sum_{r \in resources} E\_Share\_Country(r, region) * ERDData(r, time)} \end{aligned} \tag{3}$$

Following with the example, from 10 am to 11 am of February 13, 2013, the overall energy production in Baden-Württemberg was of 5.38 million kWh,

[2] It represents the European Energy Exchange (EEX) which we mentioned before.

resulting in an overall amount CO_2 emissions of 2.14 million kilogram. This leads to an average CO_2 factor for this time period in the region of Baden-Württemberg of $397\frac{g}{kWh}$.

The detailed process is described in the pseudocode in Algorithm 1.

Algorithm 1 Algorithm 1: Transformation

```
Require: $ERDData, HistoricalData, CO_2Factors$
Ensure: $CO_2\ emission\ factor\ for\ each\ region\ of\ a\ country\ by\ energy\ sources$

STEP 1
for $r \in resources$ do
    for $reg \in regions$ do
        $E_{Country}[r] = E_{Country}[r] + E\_Share\_Local[r][reg] * E_{total}[reg]$
    end for
end for

STEP 2
for $r \in resources$ do
    for $reg \in regions$ do
        if $E_{total}[r] > 0$ then
            $E\_Share\_Country[r][reg] = \frac{E\_Share\_Local[r][reg]\%*E_{total}[reg]}{E_{Country}[r]}$
        end if
    end for
end for

STEP 3
for $t \in EEXData$ do
    for $reg \in regions$ do
        for $r \in resources$ do
            $CO_2Emission[reg][t] = CO_2Emission[reg][t] + E\_Share\_Country[r][reg] * ERDData[r][t]) * Co2Factor[r];$
            $energyProd[reg][t] = energyProd[reg][t] + E\_Share\_Country[r][reg] * ERDData[r][t]);$
        end for
        $CO_2EmissionFactors[reg][t] = \frac{CO_2Emission[reg][t]}{energyProd[reg][t]}$
    end for
end for
```

At this point we should also mention that our methodology does not include the energy imports or exports that are taking place between the regions of a country or even between counties. Energy losses during the long-distance transfers, energy transformations as well as other issues are included in the real-world scenarios which shape the energy market as a very dynamic and flexible system. Germany for example, after the nuclear phase out of March 2011

(Fukushima incident) increased significantly the import of inexpensive nuclear power from the Czech Republic, stores it as pumped hydro-power and sells it to its neighbors at a much higher rate. Such complex market dynamics are not captured by our data structure and we leave this analysis for a later investigation.

3.2. Monitoring phase (i): Assessment of eco-efficiency for running VMs

In this operational phase of our system the monitoring of the cloud resources and the calculation of the eco-efficiency is performed. As presented in Figure 1, the monitoring phase involves two separate processes: (a) the monitoring of the cloud infrastructure, and (b) the calculation of the eco-efficiency for each active VM resource of the private cloud environment. As we mentioned before, we are examining the ecological footprint and efficiency of a VM from a computing system perspective, in the sense that we do not consider environmental, maintenance or installation costs. These parameters are not directly related with the performance of the application that is being executed within the VM and therefore such static figures can be calculated and considered independently.

For the first process, the *Performance and Energy Monitor* interacts with the hypervisor of the cloud infrastructure and collects the list of the active VMs. For each VM, our system monitors the status, the performance metrics (such as CPU, disk, and memory utilization) as well as power consumption of the cloud server that hosts these VMs. In this paper we are not discussing the methodology of energy metrics collection as in [14] we have presented more details about that issue. The methodology that we followed for the calculation of eco-efficiency in clouds is based on a more low level approach from the ones we presented in Section 2 to compare the useful work to the amount of carbon dioxide emitted.

The work performed by a computing system relies on three performance modules: CPU, memory, and disk. The performance capacity of the CPU can be measured either by million instructions per second (MIPS) or in order to make it even more comparable by million whetstone instructions per second (MWIPS). In the same context, the memory capacity by the amount of allocated memory (measured in bytes), and the disk performance by the number of input and output operations per second (IOPS).

However, the absolute numbers are not comparable since they are calculated in different units. In addition, literature research as well as experimentation proved that the main factor that relates with the energy consumption of a computing system is the CPU utilization [9] [10]. Hence, in this work we calculate the useful work performed by a VM as the product of its CPU utilization on the cloud host by the maximum capacity (in BWIPS, that is, billion whetstone instructions per second) of the same cloud host. For the calculation of the Pwr for the VM we consider that at 0% of the VM's CPU utilization there is a minimum power consumption Pwr_min at the node, which we use as our control value. As a result, the formula for the real-time calculation of the eco-efficiency is:

$$EcEf_{VM_l} = \frac{Perf_{max_i} \cdot \frac{U_{VM_l^i}}{100}}{CO_2EmissionFactor(region, now) \cdot Pwr_{VM_l} \mid l \in VMs_i} \tag{4}$$

In our implementation, the eco-efficiency is calculated using the information collected from the cloud infrastructure (CPU utilization and power consumption of every VM), the CO$_2$ Emission Registry (CO$_2$EmissionFactor) and the static number of BWIPS capacity which is provided by each node (calculated by benchmarking the infrastructure beforehand).

To determine the approximate power consumption incurred by a VM, its proportional part of the total power consumption R_i of the node where it is being executed is derived based on the VM's CPU utilization, including overheads. This proportional distribution with regards to the CPU utilization is accomplished with the second term of Equation 5. As in [35], a linear relationship between a node's real power consumption and its total CPU utilization is assumed. Since Domain-0's CPU utilization (privileged domain in the Xen hypervisor's nomenclature) is required to run all of the n_i VMs being executed in node i, it is considered as an overhead which is uniformly distributed between them.

$$Pwr_{VM_l} \mid l \in VMs_i = R_i \cdot \frac{U_{VM_l^i} + \frac{U_0^i}{n_i}}{U_0^i + \sum\limits_{j \in VMs_i} U_{VM_j^i}} \tag{5}$$

The unit of the eco-efficiency is defined as $\frac{billion\ whetstone\ instructions(BWI)}{CO_2(g)}$. The billion whetstone instructions represent the useful work that can be done per gram of carbon dioxide emitted. In the following the transformation steps to get to this unit is shown. As it can be seen the result has to be multiplied by 3600 to transform the unit into the desired format.

$$\begin{aligned} \frac{1\ MWIPS}{1\ Watt \cdot 1\ \frac{CO_2(g)}{kWh}} &= \frac{1\ \frac{MWI}{s}}{1\ Watt \cdot 1\ \frac{CO_2(g)}{1000\ Watt \cdot 1\ h}} \\ = \frac{1000\ \frac{MWI}{s}}{1\ \frac{CO_2(g)}{h}} &= \frac{1000\ \frac{MWI}{s}}{1\ \frac{CO_2(g)}{3600\ s}} = \frac{3600\ BWI}{1\ CO_2(g)} \end{aligned} \tag{6}$$

In order to calculate the eco-efficiency of a VM as accurate as possible, we consider the CO$_2$ emission factor available in the CO$_2$ Emission Registry (calculated as in Equation 3), that has the closest timestamp with the current time of monitoring. While the variation of the CO$_2$ emission factor has a certain pattern within a day for every region, we use the value for the same time period of the previous day. For example, if it is 11:30 a.m., the eco-efficiency of a VM is calculated considering the emission factor of the 11:00 to 12:00 a.m. of the previous day for that specific region. Hence, the eco-efficiency is calculated based not only on region but also time specific values for the CO_2 emission factors. In order to avoid inaccuracies during usage profile changes (e.g. from Friday to Saturday, or Sunday to Monday) a more sophisticated prediction mechanism that takes into account the weekly patterns along with the values of the previous day could improve the calculation of the emission factors. However, this extended functionality is out of the focus of this study.

3.3. Monitoring phase (ii): Prediction of eco-efficiency for potential VM deployments

When a client wants to deploy a new VM in an IP (Infrastructure Provider), whether he is an end user who wants to instantiate a single VM or a SP (Service Provider) who wants to deploy a service comprised of several VMs, he must decide to which IP it will forward the deployment request to ensure that his requirements are fulfilled. In some cases, this client may have a minimum desired eco-efficiency level (either energy and/or ecological efficiency) for that VM. Moreover, he may also have specified a given set of environmental-aware certifications which the IP where the VM will be deployed must possess (i.e. LEED, BREEAM, CASBEE, ISO14000, etc.). In these situations, the client must be able to determine whether the minimum required eco-efficiency for that VM can be guaranteed by the IP where the VM will be deployed, as well as if it meets the specified certification requirements.

For this reason, the IP must provide their clients with the forecasted ecological efficiency of a VM before it is deployed in their infrastructure. With this information, clients can decide whether a given IP is appropriate to deploy it or not. To predict the ecological efficiency at VM level, an IP needs to know the performance this VM will require, the power consumption that it will incur and the associated CO_2 emissions.

The calculation of these aspects is heavily conditioned by the placement of the VM in the IP resources. In the following subsections, we present two approaches to calculate the predicted ecological efficiency of a VM if it is deployed in a given IP node that both approaches can be used seamlessly, depending on the way the IP operates.

In the first approach, the potential placement of the VM has already been determined by the IP (for instance, because the IP makes a reservation of the resources to be allocated to this VM), and the provider provides a single precise forecast of the eco-efficiency level of the new VM.

In the second approach, the placement of the VM is not known (the IP will not decide the actual placement until the VM has been accepted). Consequently, the provider informs the client about the minimum and maximum eco-efficiency levels which can be achieved by the VM depending on its placement, as well as the mean value.

3.3.1. VM deployment with known placement

In this section, we predict what the ecological efficiency of a new VM (VM_l) will be if it is deployed into a given destination node ($node_{dst}$). When the client issues the VM creation, he sends a descriptor specifying the features required by the VM (number of CPUs, amount of memory, disks, etc.), as well as the amount of performance it will require ($\widehat{Perf}_{VM_l} \mid VM_l\ depl.$) for its execution. The delivered performance is equal to the normalized CPU utilization incurred by the VM in the destination node by the maximum performance which can be delivered by it, as depicted in Equation 7.

$$\widehat{Perf}_{VM_l} \mid VM_l\ depl. = \frac{\widehat{U}_{VM_l^{dst}}}{100} \cdot Perf_{max_{dst}} \tag{7}$$

Therefore, from this formula the additional CPU utilization caused by the new VM in the destination node can be calculated as in Equation 8. In our formulation, the CPU utilization of a given node ranges from 0 to 100 regardless of its number of CPUs. Therefore, it has to be guaranteed that the CPU utilization of the VM is at most the proportional part of 100% corresponding to the CPUs used by the VM. Otherwise, it means that the destination node is not able to provide the required performance to the new VM. This condition is also depicted in Equation 8. Note that, additionally, the VM scheduler has to guarantee that the destination node has enough available CPUs and RAM memory to allocate the VM.

$$\widehat{U}_{VM_l^{dst}} = \frac{100 \cdot \widehat{Perf}_{VM_l} \mid VM_l\ depl.}{Perf_{max_{dst}}} \leq \frac{100 \cdot CPU_{VM_l}}{CPU_{node_{dst}}} \tag{8}$$

The future power consumption of a VM corresponds to the proportional part of the future real power consumption of the node where the VM will run, as shown in Equation 9. Note that the Domain-0 CPU utilization is considered as an overhead equally distributed among the running VMs, including the new one to be deployed. To forecast the power consumption of the destination node, the offline power model described in [36] is used, which only needs to know the forecasted CPU utilization in the host as an input. In this case, it will consist of the future CPU utilization of the privileged domain (Domain-0) and those of the already running VMs, plus the additional CPU utilization caused by the new VM. The CPU utilization of the Domain-0 and the already running VMs will follow its original trend, and are forecasted using the variable estimator described in [36]. The CPU utilization of the new VM is calculated as in Equation 8.

$$\begin{aligned} \widehat{Pwr}_{VM_l} \mid VM_l\ depl. = \widehat{R}_{dst} \Bigg(\widehat{U}_0^{dst} + \sum_{j \in VMs_{dst}} \widehat{U}_{VM_j^{dst}} + \\ + \widehat{U}_{VM_l^{dst}} \Bigg) \cdot \frac{\widehat{U}_{VM_l^{dst}} + \frac{\widehat{U}_0^{dst}}{n_{dst}+1}}{\widehat{U}_0^{dst} + \sum_{j \in VMs_{dst}} \widehat{U}_{VM_j^{dst}} + \widehat{U}_{VM_l^{dst}}} \end{aligned} \tag{9}$$

Finally, having the predictions of the performance and the power consumption of a VM_l if it is deployed into a given destination node, the ecological efficiency can be calculated as in Equation 10. The term $\delta_{cert.}$ is a boolean variable which is equal to 1 if all the required environmental-aware certifications specified in the service manifest (service contract specified by the service owner) are in possession of the IP where the service is to be deployed, and 0 otherwise. Therefore, if the IP has the required certifications, the forecasted ecological efficiency of the service will be different from 0 and the service might be deployed in that provider (if the specified minimum ecological efficiency requirements are met).

$$\widehat{EcEf}_{VM_l} \mid VM_l\ depl. = \delta_{cert.} \cdot \frac{\widehat{Perf}_{VM_l} \mid VM_l\ depl.}{CO_2EmissionFactor \cdot \widehat{Pwr}_{VM_l} \mid VM_l\ depl.} \tag{10}$$

3.3.2. VM deployment with unknown placement

In this section, we explain how to predict the ecological efficiency of a new VM (VM_l) if its placement in the IP is not known beforehand. The required VM performance is specified by the client, so only the power consumption needs to be calculated in this case. However, given that the destination node is not known, it is impossible to determine what the exact power consumption will be. Therefore, the fairest approach is to inform the client about the range where the final eco-efficiency value will reside, as well as the mean of all the values, so the client can have an idea of the tentative final value. With this approach, the client has a clear view of the range of the future eco-efficiency and is able to determine if it will surely fulfill his requirements (if the minimum value fulfills them) or if it might (if the maximum value fulfills them but not the minimum).

To provide the aforementioned values, the power consumption is estimated by conditioning the placement to each of the possible destination nodes. The same formula as in Equation 9 is used in each calculation, and the term $\widehat{U}_{VM_l^{dst}}$ is calculated as in Equation 8. Those nodes which do not accomplish the conditions commented in Section 3.3.1 are discarded from the calculations. From all the obtained values, the minimum, maximum and mean values are taken, corresponding to the maximum, minimum and mean of the returned eco-efficiency values, respectively. For each returned value, the eco-efficiency is calculated as in Equation 10.

4. Evaluation

For the evaluation of the proposed system we performed a series of experiments using private cloud infrastructures located in Germany. The information sources that we used in order to build the CO_2 Emission Registry were on the one hand the statistics platform *Statista* [3] through which we acquired data regarding the energy mix of each federated state in Germany (historical data for the regions' energy mix), and from the European Energy Exchange Transparency Platform (EEX) which provided us the daily production and energy mix for every country(ERDDAta stream as described in the previous algorithms).

By triggering the data aggregation and transformation service the CO_2 Emission Registry is populated with information for every state in Germany, using the refresh interval that is available in the EEX dataset. In Figure 3 we present the intra-day CO_2 emission factors for the German states based on the EEX dataset of 23rd of April, 2013. As we can see, there is significant variation between the values of the German states because of the type of the energy resources available in each state. For example, in Bavaria where the CO_2 emission factor is around 131 g/kWh, the 56% of the energy is produced by nuclear plants, while in Saxony where the CO_2 emission factors is around 1087 g/kWh, 80% of the energy is produced by coal plants. In addition, we notice an important variation of the values within the day. In that context, and as we present later, we could benefit from that phenomenon and provision VM allocations depending on the eco-efficiency level that we desire. The nuclear energy impact in terms of ecological concerns is not considered in this study. We base our analysis only on the CO_2 emission levels.

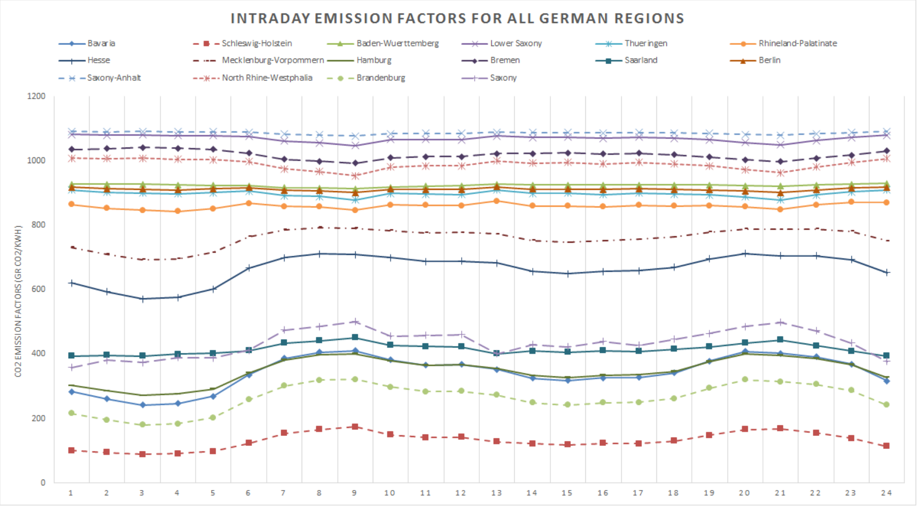

Figure 3: Intraday CO_2 emission factors for all German states.

[3] http://www.statista.com/

4.1. Evaluation of Eco-efficiency assessments

For the the testing of the monitoring functionality we instantiated one small VM instance (VM1) of 1vCPU and 1024MB memory and one big instance (VM2) of eight vCPUs and 2048 MB memory. The CPU capacity on the physical host (cloud server) has been benchmarked through the UnixBench tool at 3560 MWIPS, that is, 3.56 BWIPS. Assuming that within the VMs we execute some CPU intensive applications and considering also the fact that CPU is the driver of energy consumption into computing systems [37] [38] [9], in the following experimentation we have monitored and analyzed the CPU workload in comparison with the eco-efficiency.

We applied variable CPU load to VM1 when it was deployed in a host in Berlin and the same load when it has been deployed in Baden-Würtemberg state. Based on the collected results shown in Figure 4, and as expected, the eco-efficiency of the VM in Berlin is worse than the one in Baden-Würtemberg. There was an average 65 units improvement of the ecological efficiency, a significant 42% improvement when operating the VM in Baden-Würtemberg. The variation of both lines during the experimentation period is following the pattern of the CPU utilization, while the more distinctive spikes in the beginning and the end of the chart are caused due to the small latency of the energy metric that is reported (big and fast changes of CPU utilization cause increased energy consumption that is reported with a small latency) and therefore the changes in the CPU load are reported faster than the changes in energy metrics.

In Figure 5 we compare the ecological efficiency of the two different VMs, which are deployed in the same infrastructure (Berlin). Even though they have different specification and are subjected into different CPU load, the eco-efficiency level during operation is very similar. Here we can note that the ecological efficiency of the bigger (in computational capacity) VM presents better efficiency (around 8-9% improvement for the bigger VM).

In our last validation experiment, we plot the association of CPU utilization of a VM with it's eco-efficiency (Figure 6). We observe a non-linear relationship, with a steep increase of the efficiency from 0-30% CPU utilization, and a more stable behavior above the threshold of 30%. The threshold of 40% CPUutil seems critical while above that the eco-efficiency seems stable at around 1.5 $\frac{billion\ whetstone\ instructions(BWI)}{CO_2(g)}$. Based on this finding we note that the CPU capacity and eventually CPU workload of a VM (independent of the workload of the host) affects the ecological efficiency in a non-linear way. Thus, this could be exploited by a VM resource manager policy that would resize the VM resources depending on the workload they serve, trying to keep the CPU workload of the VM above 40%.

4.2. Evaluation of Eco-efficiency forecasts

In this section, we evaluate the precision of the tool at performing estimations of the future eco-efficiency value of a given VM. In particular, in Section 4.2.1 we make such prediction before the VM is deployed, using the approaches described in Sections 3.3.1 and 3.3.2. In Section 4.2.2, we consider a VM during its

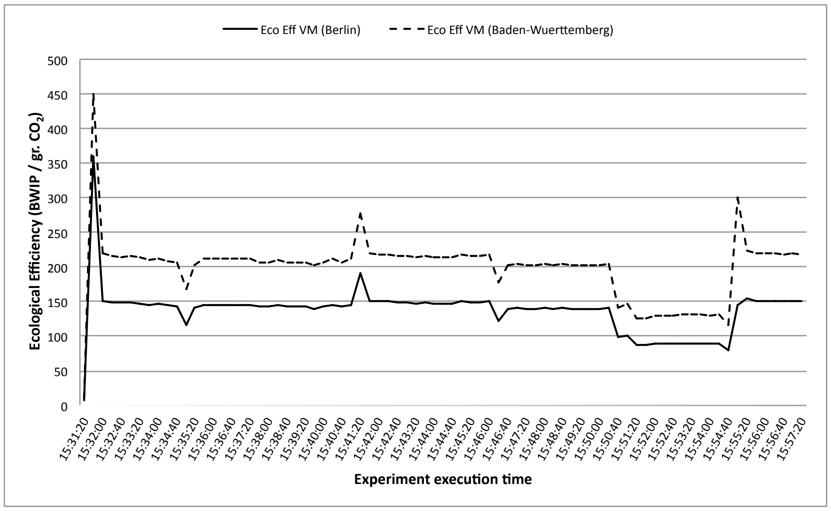

Figure 4: Ecological efficiency comparison of a VM in Berlin and in Baden Württemberg state.

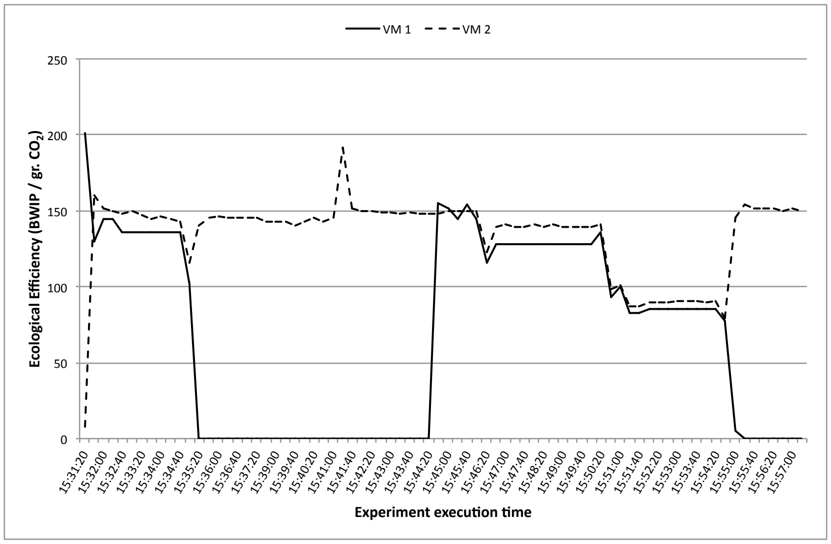

Figure 5: Ecological efficiency comparison of two VMs in Berlin state.

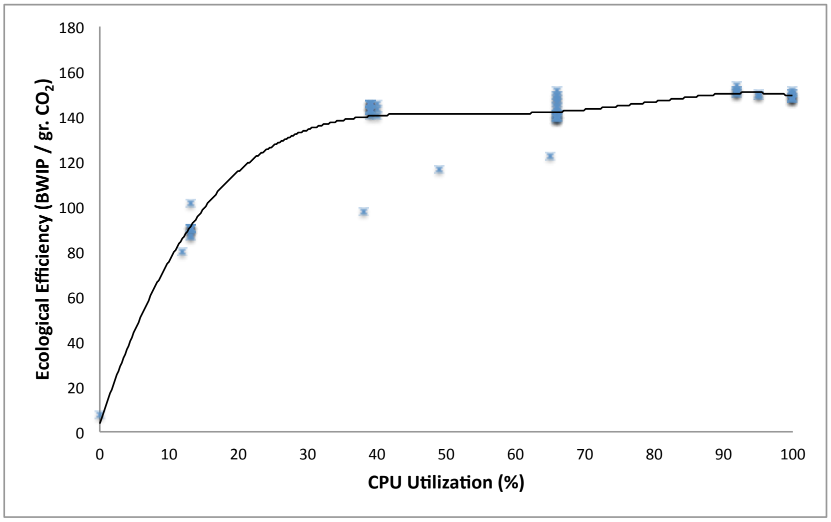

Figure 6: Relation of ecological efficiency with CPU utilization during operation of VM1 in Berlin state.

operation and we continuously predict the eco-efficiency for a given time instant in the future, comparing such forecast with the real eco-efficiency level at that particular instant, when it takes place. This process is repeated for the whole experiment execution, which simulates the behavior of a webserver operating during a week.

4.2.1. Eco-efficiency forecast at VM deployment

In this experiment, the client asks the infrastructure provider for the potential eco-efficiency of a 4-CPU VM which he wishes to deploy in its infrastructure. To this end, the client submits a description of the VM to be deployed in the OVF format [39], specifying the expected performance required by the VM as a "Property" field in the "ProductSection" section of the OVF. Depending on the way the IP provider operates, it performs such forecast following the approaches described in Section 3.3.1 or 3.3.2. Consequently, the client will receive an answer consisting of a single forecasted value or multiple values describing a range (minimum, maximum, and mean values). Finally, we deploy such VM in the infrastructure and evaluate if the performed forecast corresponds to the real eco-efficiency values achieved by the new VM.

The nodes that were used to perform this experiment are described in Table 2. The *stress* [40] and *cpulimit* [41] utilities were used to generate a constant load once the VM was fully deployed, requiring the amount of performance specified in the VM's OVF descriptor.

Table 3 displays the forecasted eco-efficiency of a VM deployment requiring different performance levels, with an unknown placement (approach of Section 3.3.2) and with a known placement in node R710 (approach of Section 3.3.1),

	R710	**R210**
CPUs	16	8
Performance	8.20 BWIPS	4.77 BWIPS
Pmin	216.70 W	89.10 W
Pmax	382.57 W	153.21 W
Location	Berlin	Berlin

Table 2: Node characteristics

when there are no VMs running in the infrastructure. As it can be observed, the minimum eco-efficiency level reported by the unknown placement approach corresponds to the one returned by node R710 (as it coincides with the one with a known placement to node R710), whereas the maximum one corresponds therefore to node R210. The potential eco-efficiency increases as the required performance is also increased. Notice that when requesting 2.2 BWIPS of performance, the returned minimum, maximum and mean eco-efficiency levels (with an unknown placement) match. This is due to the fact that only node R210 can cope with this amount of performance for a 4-CPU VM and R710 is left out of the calculation. This fact can be deduced from Table 2. As R710 provides 8.20 BWIPS with all of its 16 CPUs working at full occupation, this means that it will be able to provide a maximum of $\frac{8.20}{16} \cdot 4 = 2.05$ BWIPS when only using 4 CPUs for this particular VM, which is less than the requested amount. Similarly, R210 can provide a maximum of $\frac{4.77}{8} \cdot 4 = 2.38$ BWIPS for a 4-CPU VM. Given that this amount is less than 2.4 BWIPS, this justifies the fact that all the returned eco-efficiency values for this performance level are 0, meaning that it will be impossible to place such VM in the datacenter.

	Unknown placement			**Known**
BWIPS	**Minimum**	**Maximum**	**Mean**	**(R710)**
1	110.23	252.45	181.34	110.23
2	182.50	402.36	292.43	182.50
2.2	432.44	432.44	432.44	-
2.4	0	0	0	-

Table 3: VM eco-efficiency forecasts (no VM running in the infrastructure)

After performing the previous forecasts, a 4-CPU VM was deployed in node R710, with two performance profiles: 1 and 2 BWIPS. Then, the eco-efficiency was measured in both cases every minute during 6 minutes after its deployment. The results are displayed in Table 4. In both cases, the first assessment did not match the expected eco-efficiency, because the VM was already booting. In subsequent assessments, the VM eco-efficiency level is very close to the forecasted one. Most of the assessments are slightly below the forecasted level. This phenomena is explained after presenting the results of the next experiment.

BWIPS	1	2
Expected	110.23	182.50
Assessment 0	1.10	147.96
Assessment 1	106.34	181.38
Assessment 2	108.12	180.65
Assessment 3	115.55	175.39
Assessment 4	111.88	180.66
Assessment 5	113.72	179.18

Table 4: VM eco-efficiency assessments after deploying the VM (no VM running in the infrastructure)

The same experiment was repeated again, but in this case a 4-CPU VM using 2 BWIPS was already running in the node R710. Table 5 displays the forecasted eco-efficiency levels of the VM deployment, with an unknown and with a known (R710) placement. Given that a VM was already running in the node R710, the power consumption of this node was proportionally distributed between the running VM and the one to be deployed (see Equation 9). Because of this, the VM is expected to consume less energy than in the previous experiment and therefore the returned eco-efficiency levels for node R710 are greater. Note that the eco-efficiency levels if the VM is placed in node R210 are the same (small difference because of Domain-0 CPU utilization fluctuation) as in the previous experiment, because no VM is running in that node. The fact that there is a VM running in R710 does not affect the forecast performed when considering R210 as the destination node. The small difference between the R710 forecast with an unknown and a known placement is because the forecasts were performed in different moments and there were fluctuations in the resource usage of the running VM.

In this experiment, for a new 1 BWIPS VM, node R710 was found to be the one providing a greater potential eco-efficiency, because of the "power savings" (due to power distribution with the running VM). However, for a new 2 BWIPS VM, despite the power savings in node R710, node R210 still consumes less energy if the VM is placed there, thus providing a greater eco-efficiency level. As before, only R210 can cope with a 2.2 BWIPS VM, which explains why all the returned values match. A 2.4 BWIPS VM cannot be hold by any of the nodes and thus the returned eco-efficiency is 0 (impossible deployment).

As in the first experiment, two 4-CPU VMs requiring 1 and 2 BWIPS were deployed in the infrastructure, which has a 4-CPU VM consuming 2 BWIPS already running in node R710 at the time of each deployment. After being deployed, the VM eco-efficiency level is monitored. The first assessment corresponds to the VM booting process and is not considered to be representative. The rest of the assessments are close to the forecasted ones in both cases, but with slightly lower values.

This difference is caused by two factors: the accuracy of the power sensor and the accuracy of the power model (described in [36]) used to predict the

	Unknown placement			Known
BWIPS	**Minimum**	**Maximum**	**Mean**	**(R710)**
1	252.68	282.24	267.46	282.09
2	316.07	402.45	359.26	316.07
2.2	432.56	432.56	432.56	432.56
2.4	0	0	0	0

Table 5: VM eco-efficiency forecasts (4-CPU VM already running in the infrastructure)

power consumption. The power sensor provides lectures with 10W of resolution, which constitutes an error source for subsequent calculations based on the values returned by the sensor. The off-line power model was built using these lectures, hence the inaccuracy of the sensor directly affects the accuracy of the power model. Moreover, the used power model assumes a linear relationship between the CPU utilization and the power consumption of the server, which is not completely true and also is a source of error. This last aspect could be improved by using a more sophisticated power model.

BWIPS	1	2
Expected	282.09	316.07
Assessment 0	168.98	78.26
Assessment 1	267.08	294.15
Assessment 2	270.00	294.15
Assessment 3	269.24	291.77
Assessment 4	268.53	293.56
Assessment 5	241.67	294.15

Table 6: VM eco-efficiency assessments after deploying the VM (4-CPU VM already running in the infrastructure)

Despite the inaccuracies previously described, the tool gives a good approximation of what the eco-efficiency level of a new VM will be when deployed under different environment situations.

4.2.2. Eco-efficiency forecast during VM operation

In this section we have evaluated the precision of the tool when predicting the future eco-efficiency of a 4-CPU VM during its operation. The conducted experiment continuously predicts the eco-efficiency for a particular time in the future using the formulation in Equation 4 and estimating the future value of its variable terms. This prediction is stored in a time-series named “Forecasts”. When the aforementioned time arrives, the eco-efficiency is evaluated also using the formulation of Equation 4 but with the real data provided by the power sensor and the monitoring scripts at that particular time. This evaluation is stored in a time-series named “Assessments”. This capturing procedure is repeated during all the experiment, and finally, the “Assessments” and “Forecasts” time-series are compared by means of the Concordance Correlation Coefficient (CCC) in order to evaluate the precision of the forecasts.

The CCC, described in [42], measures the agreement between two variables and can be used to evaluate the degree of agreement among raters. In the context of this publication, it has been used to evaluate the degree of agreement between the performed assessments and forecasts (“Assessments” and “Forecasts” time-series, respectively).

As described in [42], the CCC between two vectors of length N can be calculated as:

$$\rho_c = \frac{2 \cdot s_{xy}}{{s^2}_x + {s^2}_y + (\overline{x} - \overline{y})^2} \tag{11}$$

Where:

$$\overline{x} = \frac{1}{N} \cdot \sum_{n=1}^{N} x_n \tag{12}$$

$${s^2}_x = \frac{1}{N} \cdot \sum_{n=1}^{N} (x_n - \overline{x})^2 \tag{13}$$

$$s_{xy} = \frac{1}{N} \cdot \sum_{n=1}^{N} (x_n - \overline{x}) \cdot (y_n - \overline{y}) \tag{14}$$

We have generated a synthetic workload which emulates the behavior of a webserver hosting a university web page, using the web traces of the Barcelona School of Informatics web page of the week from 5th to 11th of April 2010. In a nutshell, we generate a CPU utilization inside the VM proportional to the number of requests served during an hour by the real webserver. As the experiment progresses, the CPU utilization is changed according to the number of requests served by the real webserver during each hour of the simulated week. Each iteration of the experiment lasts 2 minutes (T_{hour}), therefore each hour of the original traces is simulated in 2 minutes in the experiment execution. The simulation of the whole week, therefore, is done in 5.6 hours of execution.

The predictions are performed, each time, for 5 different time periods in the future: 15, 30, 45, 60 and 75 seconds, corresponding to 7.5, 15, 22.5, 30 and 37.5 minutes in the “real time” of the web traces. This leads to the generation of 5 pairs of “Assessments”-“Forecasts” time-series, that are compared with the CCC to evaluate what is the precision degradation as the forecasts are performed for longer periods of time ahead.

The obtained results are presented in Table 7. For 15-second predictions (7.5 minutes in “real time”), the precision is very high: 95.54%. As it can be observed in Figure 7, the “Forecasts” time-series follows very closely the “Assessments” one during all the simulation.

Time ahead (s)	CCC (%)
15	95.54
30	94.08
45	92.51
60	89.84
75	87.47

Table 7: Obtained CCC for different prediction time-spans

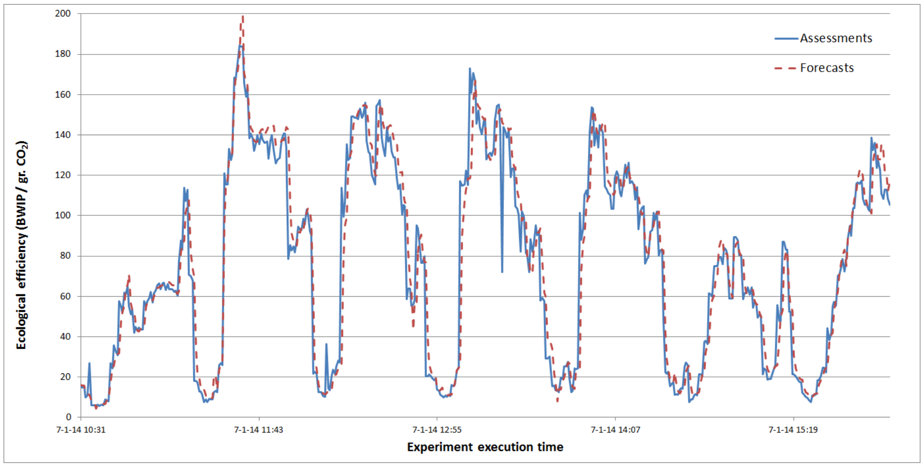

Figure 7: VM ecological efficiency assessments and 15-second forecasts during the experiment

From Table 7, it is straightforward that the longer the time span in the predictions, the lower the precision of the forecasts, which is quite obvious. Figure 8 shows how for 75-second predictions even though the "Forecasts" time-series follows the "Assessments" one during the simulation, it is not as precise as when performing 15-second predictions. In this case, the "Forecasts" time-series is delayed with regards to the changes in the "Assessments" one. Moreover, some individual predictions suffer from a greater deviation from the real observed value than before.

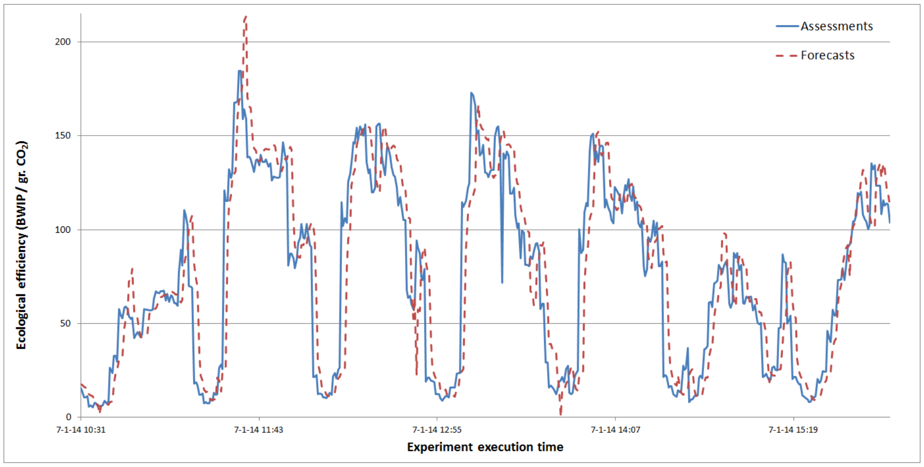

Figure 8: VM ecological efficiency assessments and 75-second forecasts during the experiment

5. Conclusions

In this paper we investigated the environmental impact of cloud resources and specifically about techniques to measure and monitor the CO_2 footprint of virtual machines in cloud computing infrastructures. We identified ecological efficiency of VMs as the metric that effectively defines such impact and we modeled this metric using energy and performance related parameters of the hosting system. To this end, we proposed a service architecture which allows monitoring the eco-efficiency of VMs in private cloud scenarios. The designed system aggregates information from public energy data-stores and historical data for each country in order to define the CO_2 emission factors for each region of interest. Through this methodology we are able to calculate the eco-efficiency of VMs in a location and time specific manner.

The proposed framework has been validated against VM deployments in cloud infrastructures within Germany. The results gathered from the executed experiments demonstrated the intra-day variation of CO_2 emission factors in each state of Germany and therefore the potential for an eco-efficient resource management of VMs. In addition, we captured the eco-efficiency levels for: (a) identical VMs deployed in different locations (Berlin, Karlsruhe), and (b) different size of VMs deployed in the same infrastructure.

We have also presented two different methods to estimate the eco-efficiency of a VM to be deployed in the infrastructure, when it specifies the amount of performance in BWIPS that it requires for its execution. These two methods can be used seamlessly depending on the way the provider operates. The results have demonstrated that the tool performs such predictions with a good accuracy, with different scenarios (with an empty infrastructure or with a VM already running on it). We have observed that the precision directly depends on the precision of the used power model. Given that the used power sensor has a granularity of 10W, this has affected the precision of our power model in the experimentation. Moreover, we have used a linear power model, but the usage of a more sophisticated one could also improve its precision.

Finally, we have also evaluated the accuracy of the tool to predict the eco-efficiency of a VM during its execution. We have stated that the precision of the tool is very high (95.54%) for short-term predictions (7.5 minutes ahead), and it degrades as predictions are performed for longer periods of time in the future.

The targeted users of the toolkit are cloud providers that can use services developed to monitor and predict the eco-efficiency of their own infrastructure. Such methodology could be integrated into cloud middleware, and therefore allow the providers for better internal management or offering new green services to the costumers. Thus, the establishment of ecological efficiency as an important factor of ICT services and cloud resources does not only have an environmental impact but also innovative market potentials.

As future steps of this research we are aiming at extending the ecological efficiency calculation formula by considering also the activity of the memory and disk resources when estimating the performed work of a VM. Contemporary production servers [4] that have been developed with the last generation CPU cores have significantly improved the processor power consumption and therefore resources like memory, network and disk must be considered as well. In addition, we would like to extend the data model and methodology of calculating the CO_2 Factors taking into account the energy providers too. Even if the energy resources of a region are the ones that the power providers are consuming, there can be significant variations in the energy mix that each company uses and therefore that could affect eco-footprint of the consumers. We also aim to improve the precision of the power predictions by using a more sophisticated power model than the current one. It should consider other aspects apart from the CPU utilization, such as memory usage, disk usage and processor performance counters. Finally, having completed with this work an analysis of the energy and ecological efficiency at the level of the VM, we will focus our efforts towards a holistic ecological modeling of cloud resources considering the cooling costs, environmental parameters and migration capabilities.

6. Acknowledgments

This work has been partly funded by the European Commission's 7th Framework Program under contracts 608679 (RenewIT), 610874 (ASCETiC), and 610456 (EuroServer). This work expresses the opinions of the authors and not necessarily those of the European Commission. The European Commission is not liable for any use that may be made of the information contained in this work. It is also supported by the Ministry of Science and Technology of Spain

[4] e.g. http://www.supermicro.com/products/system/1U/

and the European Union (FEDER funds) under contract TIN2012-34557, by the Generalitat de Catalunya under contract 2014-SGR-1051.

References

[1] CEET anual report, Tech. rep., Centre for Energy-Efficient Telecommunications (CEET) (2013).
URL http://www.ceet.unimelb.edu.au/publications/downloads/ceet-annualreport-2013.pdf

[2] Gartner, Gartner Report for ICT Industry (2009).
URL www.gartner.com/it/page.jsp?id=503867

[3] NESSI, Strategic research and innovation agenda (2013).
URL http://www.nessi-europe.com/Files/Private/NESSI_SRIA_Final.pdf

[4] Booz&Co, Going green addressing untapped consumer potential in the telecom industry (2008).
URL http://www.booz.com/media/file/Going_Green.pdf

[5] Forrester, Market update: The state of green it adoption (2010).
URL http://www.forrester.com/Market+Update+The+State+Of+Green+IT+Adoption+Q2+2010/fulltext/-/E-RES56554?docid=56554

[6] A. Berl, E. Gelenbe, M. Di Girolamo, G. Giuliani, H. De Meer, M. Q. Dang, K. Pentikousis, Energy-efficient cloud computing, The Computer Journal 53 (7) (2010) 1045–1051. doi:10.1093/comjnl/bxp080.

[7] G. Katsaros, P. Stichler, Quantifying ecological efficiency in cloud computing., in: J. Altmann, K. Vanmechelen, O. F. Rana (Eds.), International Conference on Economics of Grids, Clouds, Systems, and Services (GECON), Vol. 8193 of Lecture Notes in Computer Science, Springer, 2013, pp. 74–89. doi:10.1007/978-3-319-02414-1_6.

[8] A. Greenberg, J. Hamilton, D. A. Maltz, P. Patel, The cost of a cloud: research problems in data center networks, SIGCOMM Comput. Commun. Rev. 39 (1) (2008) 68–73. doi:10.1145/1496091.1496103.

[9] Q. Chen, P. Grosso, K. V. D. Veldt, C. D. Laat, R. Hofman, H. Bal, Profiling Energy Consumption of VMs for Green Cloud Computing, in: 2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing, IEEE, 2011, pp. 768–775. doi:10.1109/DASC.2011.131.

[10] A. Kansal, F. Zhao, A. A. Bhattacharya, Virtual Machine Power Metering and Provisioning, in: Proceedings of the 1st ACM symposium on Cloud computing, Indianapolis, IN, USA, 2010, pp. 39–50. doi:10.1145/1807128.1807136.

[11] J. Stoess, C. Lang, F. Bellosa, Energy management for hypervisor-based virtual machines, in: Proceedings of the USENIX Annual Technical Conference, Santa Clara, CA, USA, ATC'07, USENIX Association, Berkeley, CA, USA, 2007, pp. 1:1–1:14.

[12] A. E. Husain Bohra, V. Chaudhary, VMeter: Power modelling for virtualized clouds, in: 2010 IEEE International Symposium on Parallel & Distributed Processing, Workshops and Phd Forum (IPDPSW), IEEE, 2010, pp. 1–8. doi:10.1109/IPDPSW.2010.5470907.

[13] R. Basmadjian, N. Ali, F. Niedermeier, H. de Meer, G. Giuliani, A methodology to predict the power consumption of servers in data centres, in: Proceedings of the 2nd International Conference on Energy-Efficient Computing and Networking, e-Energy '11, ACM, New York, NY, USA, 2011, pp. 1–10. doi:10.1145/2318716.2318718.

[14] G. Katsaros, J. Subirats, J. O. Fitó, J. Guitart, P. Gilet, D. Espling, A service framework for energy-aware monitoring and VM management in Clouds, Future Generation Computer Systems 29 (8) (2013) 2077–2091. doi:10.1016/j.future.2012.12.006.

[15] H. Vandierendonck, K. De Bosschere, Many benchmarks stress the same bottlenecks, in: Workshop on Computer Architecture Evaluation Using Commercial Workloads, 2004, pp. 57–64.

[16] A. Phansalkar, A. Joshi, L. Eeckhout, L. John, Measuring program similarity: Experiments with SPEC CPU benchmark suites, in: 2005 IEEE International Symposium on Performance Analysis of Systems and Software, Austin, TX, USA, 20-22 March 2005, 2005, pp. 10–20. doi:10.1109/ISPASS.2005.1430555.

[17] N. Quang-Hung, P. D. Nien, N. H. Nam, N. H. Tuong, N. Thoai, A genetic algorithm for power-aware virtual machine allocation in private cloud, in: ICT-EurAsia'13: Proceedings of the 2013 International Conference on Information and Communication Technology, Yogyakarta, Indonesia, March 25-29, 2013, Springer-Verlag, 2013, pp. 183–191. doi:10.1007/978-3-642-36818-9_19.

[18] A. Beloglazov, J. Abawajy, R. Buyya, Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing, Future Generation Computer Systems 28 (5) (2012) 755–768. doi:10.1016/j.future.2011.04.017.

[19] C.-H. Hsu, S.-C. Chen, C.-C. Lee, H.-Y. Chang, K.-C. Lai, K.-C. Li, C. Rong, Energy-Aware Task Consolidation Technique for Cloud Computing, in: CLOUDCOM '11: Proceedings of the 2011 IEEE Third International Conference on Cloud Computing Technology and Science, Athens, Greece, IEEE, 2011. doi:10.1109/CloudCom.2011.25.

[20] C. Lin, P. Liu, J. Wu, Energy-efficient virtual machine provision algorithms for cloud systems, in: 2011 Fourth IEEE International Conference on Utility and Cloud Computing (UCC), Victoria, Australia, 2011, pp. 81–88. doi:10.1109/UCC.2011.21.

[21] Google, Google Green Products, March 2013 (2013).
URL `http://www.google.com/green/bigpicture`

[22] E. Curry, S. Hasan, M. White, H. Melvin, An environmental chargeback for data center and cloud computing consumers, in: J. Huusko, H. de Meer, S. Klingert, A. Somov (Eds.), Energy Efficient Data Centers, Vol. 7396 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2012, pp. 117–128. doi:10.1007/978-3-642-33645-4_11.

[23] F. Moghaddam, M. Cheriet, K. K. Nguyen, Low carbon virtual private clouds, in: 2011 IEEE International Conference on Cloud Computing (CLOUD), Washington, DC, USA, 2011, pp. 259–266. doi:10.1109/CLOUD.2011.36.

[24] P. X. Gao, A. R. Curtis, B. Wong, S. Keshav, It's not easy being green, in: Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication, Helsinki, Finland, SIGCOMM '12, ACM, 2012, pp. 211–222. doi:10.1145/2342356.2342398.

[25] S. K. Garg, C. S. Yeo, R. Buyya, Green cloud framework for improving carbon efficiency of clouds, in: Proceedings of the 17th international conference on Parallel Processing, Bordeaux, France, Euro-Par'11, Springer-Verlag, 2011, pp. 491–502. doi:10.1007/978-3-642-23400-2_45.

[26] M. Poess, R. O. Nambiar, K. Vaid, J. M. Stephens, Jr., K. Huppler, E. Haines, Energy benchmarks: a detailed analysis, in: 1st International Conference on Energy-Efficient Computing and Networking, 13-15 April 2010, Passau, Germany, e-Energy '10, ACM, 2010, pp. 131–140. doi:10.1145/1791314.1791336.

[27] G. Dhiman, G. Marchetti, T. Rosing, vGreen: a system for energy efficient computing in virtualized environments, in: 14th ACM/IEEE International Symposium on Low Power Electronics and Design, 19-21 August 2009, San Francisco, CA, USA, ISLPED '09, ACM, 2009, pp. 243–248. doi:10.1145/1594233.1594292.

[28] Y. C. Lee, A. Y. Zomaya, Energy efficient utilization of resources in cloud computing systems, The Journal of Supercomputing 60 (2) (2012) 268–280. doi:10.1007/s11227-010-0421-3.

[29] A. Beloglazov, R. Buyya, Energy efficient resource management in virtualized cloud data centers, in: 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGrid), 2010, pp. 826 –831. doi:10.1109/CCGRID.2010.46.

[30] Í. Goiri, J. L. Berral, J. O. Fitó, F. Julià, R. Nou, J. Guitart, R. Gavaldà, J. Torres, Energy-efficient and multifaceted resource management for profit-driven virtualized data centers, Future Generation Computer Systems 28 (5) (2012) 718–731. doi:10.1016/j.future.2011.12.002.

[31] Í. Goiri, R. Beauchea, K. Le, T. D. Nguyen, M. E. Haque, J. Guitart, J. Torres, R. Bianchini, Greenslot: scheduling energy consumption in green datacenters, in: Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis, Seattle, WA, USA, ACM, 2011, p. 20. doi:10.1145/2063384.2063411.

[32] Greenpeace, Campain Report, How Clean is Your Cloud? (2012).
URL `http://www.greenpeace.org/international/en/publications/Campaign-reports/Climate-Reports/How-Clean-is-Your-Cloud/`

[33] H.-J. Wagner, M. Koch, J. Burkhardt, T. Böckmann, N. Feck, P. Kruse, CO 2 -Emissionen der Stromerzeugung, BWK 59 (10) (2007) 44–52.

[34] D. Lübbert, CO 2 -Bilanzen verschiedener Energieträger im Vergleich, Tech. rep. (2007).

[35] S. Rivoire, M. A. Shah, P. Ranganathan, C. Kozyrakis, Joulesort: a balanced energy-efficiency benchmark, in: Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data, Beijing, China, SIGMOD '07, ACM, 2007, pp. 365–376.

[36] J. Subirats, J. Guitart, Assessing and Forecasting Energy and Ecological Efficiency on Cloud Computing Platforms, Tech. Rep. UPC-DAC-RR-2013-48, Universitat Politecnica de Catalunya, Departament d'Arquitectura de Computadors (09 2010).
URL `http://www.ac.upc.edu/app/research-reports/html/RR/2013/48.pdf`

[37] Z. Zhang, S. Fu, Profiling and analysis of power consumption for virtualized systems and applications, in: 2010 IEEE 29th International Performance Computing and Communications Conference (IPCCC), 2010, pp. 329–330. doi:10.1109/PCCC.2010.5682290.

[38] H. Viswanathan, E. Lee, I. Rodero, D. Pompili, M. Parashar, M. Gamell, Energy-Aware Application-Centric VM Allocation for HPC Workloads, in: 2011 IEEE International Symposium on Parallel and Distributed Processing Workshops and Phd Forum (IPDPSW), 2011, pp. 890–897. doi:10.1109/IPDPS.2011.234.

[39] DMTF, Open virtualization format specification (2 2009).
URL `http://www.dmtf.org/sites/default/files/standards/documents/DSP0243_1.0.0.pdf`

[40] stress Workload Generator (01 2012).
URL `http://weather.ou.edu/~apw/projects/stress/`

[41] cpulimit Tool (07 2009).
URL `https://github.com/opsengine/cpulimit`

[42] I. Lawrence, K. Lin, A concordance correlation coefficient to evaluate reproducibility, Biometrics 45 (1) (1989) 255–268.