

Resource Allocation on a Hybrid Cloud for Smart Grids

Alan Briones, Ramon Martin de Pozuelo, Joan Navarro, Agustín Zaballos

Engineering Department, La Salle – Ramon Llull University

Quatre Camins, 30, 08022, Barcelona (Spain)

E-mail: {abriones, ramonmdp, jnavarro, zaballos}@salleurl.edu

Received: December 14, 2015

Accepted: March 14, 2016

Published: March 31, 2016

DOI: 10.5296/npa.v8i1.8721

URL: <http://dx.doi.org/10.5296/npa.v8i1.8721>

Abstract

The use of hybrid clouds enables companies to cover their demands of IT resources saving costs and gaining flexibility in the deployment of infrastructures by paying under demand these resources. However, considering a scenario with various services to be allocated in more than one cloud, it is necessary to find the distribution of services that minimizes the overall operating costs. This paper researches on the resource allocation methodology to be applied in a multi-cloud scenario based on the findings derived from the framework used for the FINESCE project. The purpose of this work is to define a methodology to assist on the hybrid cloud selection and configuration in the Smart Grid for both generic and highly-constrained scenarios in terms of latency and availability. Specifically, the presented method is aimed to determine which is the best cloud to allocate a resource by (1) optimizing the system with the information of the network and (2) minimizing the occurrence of collapsed or underused virtual machines. Also, to assess the performance of this method and any alternative proposals, a general set of metrics has been defined. These metrics have been refined taking into account the expertise of FINESCE partners in order to shape Smart Grid clouds and reduce the complexity of computation. Finally, using the data extracted from the FINESCE testbed, a decision tree is used to come up with the best resource allocation scheme.

Keywords: Cloud Computing, Decision Tree, Hybrid Cloud, Resource Allocation, Smart Grids, Virtual Machines.

1. Introduction

Cloud Computing is a resource exploitation model that typically offers on-demand storage and computation services (and its derived applications) to third parties through a high capacity access network. The usage of server farms can provide different services on the

cloud (web, data storage, databases, etc.), allowing clients for a remote access to this information or application. These servers are able to handle different requests from anywhere at any time, featuring high mobility and scalability depending on the deployment model.

There are three different cloud deployment models [1]. First, the public cloud is a model in which resources for different companies are provided over a shared infrastructure. The objective is to amortize the global infrastructure cost with multiple clients. These resources can be accessed through data connections creating resilient platforms that can increase and decrease depending on the needs of the moment, paying only for the used resources on the selected cloud [2]. The main advantage is its flexibility, allowing on demand load peaks assimilation. The commercial public clouds are widely used platforms nowadays, providing their own services and platforms to deploy services above.

Another model is the private cloud, where the deployed platform is only exploitable by the company that has deployed it, having exclusive access to the resources. This platform is sized for an expected demand [3]. These solutions generate a feeling of greater safety for users in this type of deployment because of the use of technologies related to their own company and the lack of shared resources. However, the deployment may be out of reach for many companies because, despite its greater customization capabilities, it requires stronger economic investments at the beginning and during its operation. The update that the owner must take to avoid obsolescence is also high. This does not happen with the public cloud, where such costs are carried by the cloud provider [4].

In recent years, the use of cloud computing by enterprises has increased and organizations are deciding to adopt hybrid cloud architectures, looking for a mix of the advantages of private and public clouds [5]. This model combines resources of both. Companies can use its own infrastructure (private cloud), which is insufficient to successfully handle peaks workloads, and solve these peaks using the resources of an external provider (public cloud). The economic savings that offer public cloud services are higher than those of private clouds, although the level of privacy is lower [6]. Therefore, an organization can optimize its efficiency using public cloud services for not critical or lower sensitive data transactions, and use only the private cloud in those cases where it is necessary [7]. The main problem of this hybrid option is its complexity. It requires interconnect and integrate its own infrastructure with another one managed by a different institution. Therefore, hybrid clouds generally demand higher levels of management and administration [8], requiring a decision-maker system – an orchestrator of resources – for its allocation automation. In addition, since the different resources are allocated in different datacenters and are managed by different organizations, it is important to ensure the visibility of the whole through a single administration tool or dashboard [9].

Although there are many case studies for hybrid cloud computing, this study will refer to the case raised in the FINESCE (Future INtErnet Smart Utility ServiCEs) [10] project funded by European Union. The FINESCE project is the smart energy use case project of the second phase of Future Internet Public Private Partnership Programme (FI-PPP) [11] within FP7. It

was in charge of testing FIWARE [12] software architecture in the smart energy field. It was carried by a consortium that included leading energy and ICT operators, manufacturers and service providers as well as research organizations and SMEs from all over Europe. The project builds on and extended the results of FI-PPP FINSENY project to realize real-time smart energy services, organizing and running different field trials in 7 European countries.

One of the objectives of this project proposed a private cloud to store recent gathered data generated in a Smart Grid environment (smart meter data, electric vehicle charging stations, etc.), but there is not always enough storage capacity to keep all the historical data. Then, a public cloud is used to respond less restrictive queries or to handle peak demands, using an outsourcing burst. In the case of outsourcing burst, there is only an additional expense on demand when the private cloud cannot provide all the services, being the necessary additional resources provided by the public cloud [13].

Concretely, Software Define Utility trial (SDU) specifies data gathering from utility and data replication between nodes located in two different environments, in public and private clouds, creating a hybrid cloud. These nodes have the ability to replicate information through them and aim to store information in several allocations to have access from anywhere, regarding the cyber-security aspects, and allowing the users with the corresponding permissions to access the system.

However, the crucial factor for economic savings in IT by using a hybrid cloud is the optimal allocation of resources. If we imagine a scenario with different services to be allocated in more than one cloud, the distribution of these services is not trivial. Besides, a high time response is a detriment of the Quality of Service (QoS) offered by the cloud [14]. The fact of choosing a particular location without a defined strategy may entail not the best choice for a resource distribution to fulfill the defined requirements or can represent a cost much higher than the optimum cost [15]. It is necessary to design a set of rules that mark preferences, priorities and limits of cost, time, etc. in order to obtain the best possible location for services or data in a particular scenario [16]. The first step in this process is to analyze the different objective metrics that help to characterize a cloud.

The rest of this paper is structured as follows. Section 2 extends the explanation of the context in which the Hybrid Cloud for Smart Grids has been deployed and tested. In Section 3, this paper classifies hybrid cloud metrics in four categories and qualitatively evaluates these metrics depending on its importance for different actions that can be carried out in a cloud. In Section 4, it is determined which are the metrics that provide more relevant information about the proposed scenario of Hybrid Cloud for Smart Grids. This stage analyzes the metrics that are needed at the time to decide where data should be replicated or new resources launched. Moreover, it is defined how each metric is measured and its unit measures. Section 5 explains how to measure these metrics and how the data can be extracted from the testbed. Section 6 shows the results obtained from the gathered data and the generation process of the decision tree algorithm for resource allocation. Finally, Section 7 brings the results and conclusions of the paper to get an overview of the process.

2. Hybrid Cloud for Smart Grid deployment scenario

Relying on the data network infrastructure of the utility, the FINESCE Software Define Utility Trial [9][17] interconnects different FIDEVs (Finesce DEVICES) placed at different ESB facilities in Ireland (Electricity Supply Board is Ireland's premier electricity utility [18]) and La Salle R&D Lab in Barcelona, emulating the interconnection of spread locations (i.e. interconnecting separated secondary electrical substation). FIDEV is a platform built on commodity hardware, in which different software subsystems provide several communications and data concentrator functionalities. Among these new functionalities, a distributed storage system can be build over different interconnected FIDEVs, acting as a distributed data center. FIDEVs also provide seamless interaction between this FIDEVs-based private distributed storage system and the FIWARE Lab Cloud. In this sense, the system consists of a FIDEVs set (physical or virtualized) that will constitute a private cloud, plus public cloud storage capabilities by means of FIWARE Lab. Data can reside in any of the two clouds and be moved from one to another according to the decision of their owners.

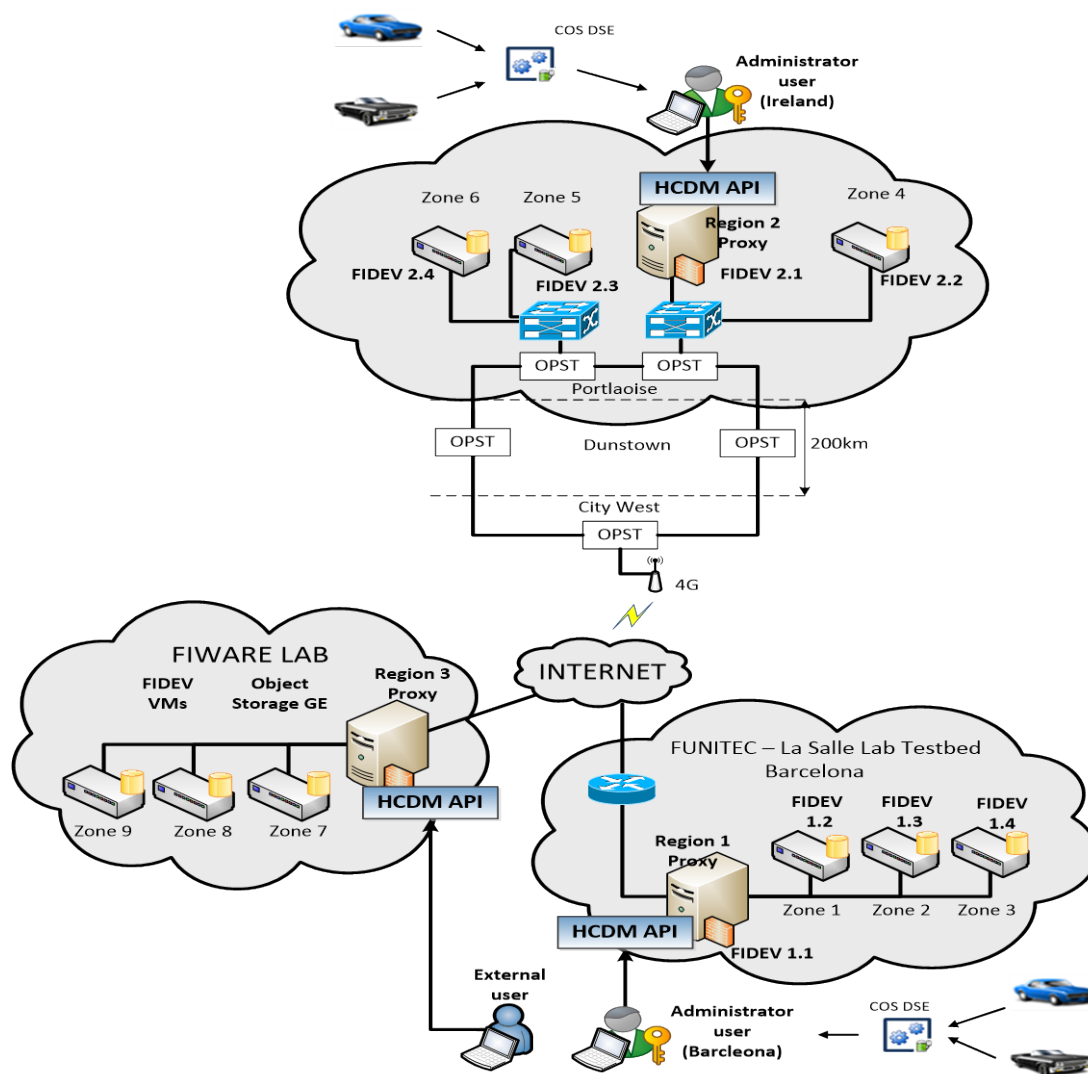


Figure 1. FINESCE Software Defined Utility trial scenario

In order to test this proof-of-concept scenario, data gathered from communications network monitoring system and electric vehicle charging points is collected and then injected into the Hybrid Cloud data storage system. It is proposed to be one of the main components in which a Software Define Utility system could rely on, providing a flexible data management system that will allow to maintain utility generated data locally replicated and also in the cloud (through FIWARE Lab), when needed (see Figure 1).

3. Cloud metrics

Several studies are focused specifically in how to classify metrics of a cloud [19][20][21][22] and computing a cloud service [23], with the objective of characterizing the cloud and extracting a unique value to determine the adequate behavior of the cloud.

As a first contribution, this paper presents a table (Table 1) with data obtained from the work carried out in the FINESCE project. In the vertical axis, this table provides a set of metrics classified in different categories: Computation, Storage and Network. The Miscellanea category is added to complement this classification, including economic, elasticity, scalability or security aspects. So, the importance of several metrics on a cloud is qualitatively defined depending on which action is being carried out, with a weight of 1 (low importance), 2 (medium importance) or 3 (high importance). It is important to emphasize that the assigned colors, red (1), yellow (2) or green (3), were defined through a process of cross opinion between different stakeholders involved in the project. The boxes in blue were subsequently modified in a second iteration, analyzing the contributions and opinions from other partners and cloud service providers.

The horizontal axis defines a set of processes and operations that are performed in a cloud, in order to characterize the behavior of the cloud. Depending on the type of services mostly used, and therefore the type of cloud demanded by the user, this table can determine which metrics should be selected if we want to evaluate and grade the available clouds where the user can allocate resources. If we want to use it in a more fine-grained view, we can apply it to select which metrics should be taken into account to evaluate in which cloud to allocate a resource before doing one of the specific operations defined in Table 1.

After reviewing several similar studies [24], it was found that there is a tendency to follow a specific strategy to deploy services in the most suitable cloud. This strategy is based on using metrics to evaluate the location of services in one cloud or another. Some studies [25][26] stand on the premise that the placement of resources will always be cheaper in the private cloud than in public clouds. This is because it is supposed to have available resources in the private cloud (the investment to deploy the infrastructure has already been performed previously). Thereby, the proposal is to place all the services in the private cloud, which shall not assume any additional cost unless the operation itself, relying on a threshold value. Beyond this threshold, the resources should be placed in the public cloud due to peak loads and their associated cost on demand. To set this threshold, the use of metrics that help to mark the boundary of the private cloud usage is necessary.

Table 1. Metrics analysis and evaluation

Process/Operation	Metrics																																		
	Miscellaneous												Network Metrics						Disk Metrics				Computation Metrics												
Economic costs	3	2	2	2	2	2	3	3	3	3	3	2	2	2	2	2	2	2	2	2	2	2													
Percentage of apps that meet SLA	3	2	1	1	1	1	1	2	2	2	2	1	3	2	2	2	2	2	2	2	3	1	1												
Total-Cost-of-Ownership	3	2	2	2	2	2	3	3	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2												
Infrastructure cost	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1												
Power Cost	3	2	2	3	2	2	2	3	3	3	3	1	3	2	2	2	2	2	2	2	3	3	2												
Server Cost	3	2	1	1	2	2	2	3	3	3	3	1	3	2	2	2	2	2	2	3	1	1													
Networking Cost	3	3	1	1	3	3	1	1	1	1	2	1	1	3	2	2	2	2	2	3	1	1													
Maintenance Cost	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2													
Cold Spares estimation	2	2	1	1	1	1	3	3	3	3	2	1	3	2	2	2	2	2	2	2	1	1													
Hot Spares estimation	2	2	1	1	1	1	3	3	3	3	2	1	3	2	2	2	2	2	2	2	1	1													
Generic costs	3	2	2	2	2	2	3	3	3	3	3	3	3	2	2	2	2	2	2	3	3	2													
EUE (kWh it) Energy Use Efficiency	2	3	2	2	2	2	3	3	3	3	3	3	3	3	1	1	1	2	3	2															
EUE (kWh) Energy Use Efficiency	2	3	2	2	2	2	3	3	3	3	3	3	3	3	1	1	1	2	3	2															
EUE (CPU) Energy Use Efficiency	2	3	2	2	2	2	3	3	3	3	3	3	3	3	1	1	1	2	3	2															
Power Usage Effectiveness (PUE)	2	3	2	2	2	2	3	3	3	3	3	3	3	3	1	1	1	2	3	2															
Data Center Compute Efficiency (DCcE)	2	3	3	3	1	1	2	2	2	2	2	3	2	3	1	1	1	1	2	3	3														
Service Compute Efficiency (ScE)	2	3	3	3	1	1	2	2	2	2	2	3	3	3	1	1	1	1	3	3	3														
Digital Service Efficiency	3	2	2	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	2	1	1														
Scheduling based on fairness	3	3	2	2	3	2	2	2	2	2	2	2	2	2	2	2	1	1	3	1	1														
Scheduling/Metric based on execution cost	3	3	2	2	3	2	2	2	2	2	2	2	2	2	2	2	1	1	3	1	1														
Spot price dynamics	3	3	1	1	3	2	3	3	3	3	2	2	2	3	2	2	2	2	3	1	1														
Resiliency (security/isolation)	3	3	2	2	3	2	2	2	2	2	2	2	3	3	2	2	2	2	3	1	2														
Reliability	3	3	2	2	3	3	2	2	2	2	2	1	3	3	3	2	2	2	3	1	2														
Availability	3	3	2	2	3	3	2	2	2	2	2	1	3	3	3	2	2	2	3	1	2														
Jitter	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1														
Latency	3	3	2	1	3	2	2	2	2	2	2	2	3	2	2	1	1	1	3	1	1														
Speed	2	3	1	1	3	2	1	1	1	1	3	1	3	3	3	2	2	2	3	1	1														
Migration Cost	1	3	1	2	3	2	1	1	1	1	1	1	1	1	1	3	3	1	3	2	2														
Hardware reliability	2	2	1	1	2	1	2	2	2	2	1	2	2	2	2	2	2	2	2	1	1														
Data throughput	1	3	1	1	3	2	2	2	1	1	3	1	2	3	3	3	3	3	3	1	1														
Average delivery time of new products or services	1	1	1	1	1	1	1	1	1	1	1	3	1	1	1	1	1	1	1	1	1														
Average time to deploy an application	3	3	1	1	3	2	1	1	1	1	1	3	1	1	1	1	1	1	3	1	2														
Time to Genesis	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1														
Recovery time	3	3	1	1	3	3	1	1	1	1	1	3	3	3	2	2	2	3	1	1															
Average time to provision a node	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1														
Average Weighted Response Time (AWRT)	3	3	1	1	3	3	3	1	3	3	3	1	3	3	3	2	2	2	3	1	1														
Instance Efficiency (% CPU Peak)	3	3	1	1	3	2	2	2	2	2	1	3	3	3	2	2	2	3	1	1															
ECU Ratio (Gflops/ECU)	3	3	1	1	2	2	1	1	3	1	1	1	2	2	2	2	2	2	3	1	2														
Maximum performance loss	3	3	1	1	3	2	2	2	2	2	2	1	3	3	2	2	2	2	3	1	1														
Degradation time	3	3	1	1	3	2	1	1	1	1	1	3	3	3	2	2	2	3	1	1															
Slowdown time/fairness	3	3	1	1	2	2	2	2	2	2	1	2	3	2	2	2	2	3	1	1															
Elasticity	3	3	2	1	3	1	3	3	3	3	2	3	1	1	1	1	1	3	2	2															
Density	2	2	1	1	2	1	1	1	1	1	1	3	2	2	1	1	1	3	1	1															
Time to deploy	3	3	1	1		1	1	1	1	1	1	2	1	1	1	1	1	3	1	1															
Runtime performance	3	3	1	1	3	2	2	2	2	2	2	1	3	3	3	2	2	3	1	1															
Workload	3	3	1	1	3	2	2	2	2	2	2	1	3	3	3	3	3	3	3	1	1														

4. Metric Selection

The usage and analysis of a single metric can help to see how the hybrid architecture behaves. The next step is taking into account more than one metric to design an optimal resource allocation. Considering the set of metrics presented in Table 1, a subset of them has been done based on the metrics that can provide relevant information about the features, performance or activity of a cloud in the Smart Grid field. It reduces the complexity of decision computation and discards some metrics that the partners considered less relevant in the scenario presented. These metrics can characterize the cloud behavior and provide information that can lead to decide in which cloud a resource should be allocated. This subset was composed by the following eleven metrics:

4.1. Workload

Workload is the increase of work that is generated when it is added a new virtual machine in a cloud. This metric determines which the cloud with better productivity is. It indicates the cloud that can manage more tasks without decrease its performance.

To measure the Workload [27] it has been found the following concepts and measure units:

- Memory utilization (MB, GB)
- CPU utilization (GHz)
- Disk Space utilization (MB, GB)
- SWAP utilization (MB, GB)

Using these four concepts, it is obtained enough information to know the status and activity of the virtual machine.

4.2. Hardware Reliability

Hardware Reliability indicates how much reliable is the server hardware of the data centers that provide the cloud service. This metric determines which cloud has the most secure hardware. It must be chosen the cloud with a better hardware reliability to store the most demanded information, that information that requires access many times and at any time, and especially the information that can create a critical situation if missing.

To measure the Hardware Reliability [28] two concepts are evaluated:

- Mean Time to Fail (MTTF) is the elapsed time since the beginning until a failure that is not repairable is produced. Usually represented in hours (h) and minutes (m).
- Mean Time between Failures (MTBF) is the elapsed time between two consecutive repairable failures. Usually represented in hours (h) and minutes (m).

4.3. Average Weighted Response Time (AWRT)

Average Weighted Response Time (AWRT) is the response time of the cloud, the elapsed

time since the user does a request until the user receives the response. This metric points out the most suitable cloud for specific tasks. AWRT depends on two concepts:

- The Average Request Time (ART) of a task will be calculated counting the average time of all the tasks with similar characteristics. Measured in milliseconds (ms).
- The Number of Instances Running is used to give a weight to request time.

The AWRT is the multiplication of these two parameters, which has no units. A list of ranges must be established to consider a good response time or not, according the AWRT value.

4.4. Average Time to Deploy an Application (ATDA)

Average Time to Deploy an Application (ATDA) is the average time invested in deployment of certain application. This measure determines which cloud can deploy a new application faster to avoid lose time.

It is only required to calculate the needed time to deploy an application. The average is calculated with the sum of all the deployment times, divided between the total of the number of deployments.

4.5. Data Throughput

Data Throughput is the amount of data per second transmitted through all the interfaces of a virtual machine. This metric determinates which cloud is able to send more data through the network. It can be used to select the fastest cloud to do specific tasks, for example, tasks that have to be done urgently or tasks that a high amount of data is required to transfer.

Data Throughput can be calculated with two variables related with the transfer bit/Byte speed.

- TCP/UDP/IP Transfer bit/Byte Speed (bps, Mbps, Gbps, MB/s, GB/s, TB/s).
- MPI Transfer bit/Byte Speed (bps, MB/s, GB/s).

Transfer bit is based in the communication evaluation metrics, meaning the speed of data transmitted.

4.6. Latency

Latency, or Transfer Delay, is the time between the information is sent by the source and it is received by the destination. This metric has to be considered depending on the type of services is necessary to be deployed as, for example, live stream server.

Latency can be calculated with two variables related with the transfer delay.

- TCP/UDP/IP Transfer Delay (s, ms).
- MPI Transfer Delay (s, μ s).

Transfer Delay is related with the communication evaluation metrics. One way to

calculate an approximation of the Transfer Delay is dividing the Round Trip Time (RTT) between two (considering the same delay in both ways). RTT is the elapsed time that a packet takes to arrive to the receiver, plus the elapsed time that it takes to come back to the sender.

4.7. Jitter

Jitter indicates the latency difference between different packets sent to the same receiver. This metric determines if the delay between packages is stable or the delay changes, indicating if there is a problem in the network.

Jitter can be evaluated calculating the Mean Packet to Packet Delay Variation (MPPDV):

$$MPPDV = average (abs (delay(i) - delay(i - 1))) \quad (1)$$

This formula calculates the variation of the latency of each packet, and is usually represented in milliseconds (ms). For this reason, to calculate jitter, it is necessary to measure the latency of each packet first.

4.8. Network Reliability

Network Reliability measures the amount of lost data. This metric determines which cloud is more reliable; the cloud that has many packets losses reduces its efficiency due to the necessity of sending the packets lost.

The Network Reliability can be calculated by the evaluation of the Connection Error Rate (%), taking into account those packets that have been lost, in relation to the total of transactions sent in a connection with the cloud.

4.9. Spot Price Dynamics

It is the price of the launched instances against the cloud. It measures the price per instance every certain time unit (for example, every hour or day). Using this metric, it can be chosen the instances available for a determinate user depending on the price he is willing to pay. This metric determines the virtual machine that can be afforded basing on its cost.

The standard to measure the price of each instance is €/h or \$/h, and this price is determined for the cloud service provider.

4.10. Total-Cost-of-Ownership (TCO)

Total-Cost-of-Ownership (TCO) determines the direct and indirect costs of a datacenter. It is the sum of all the costs of the system (infrastructures, power, server, network, and maintenance costs). With this metric can be decided if it is necessary to pay more to increase their system or not.

TCO [29] is the total cost of a whole system, is the sum of each cost (\$):

$$TCO = C_{infrastructure} + C_{power} + C_{server} + C_{network} + C_{maintenance} \quad (2)$$

It is required to know first the value of the costs to be able to calculate the TCO. With this value, we can divide the resulting economic cost between the total of instances running in that datacenter to obtain the total cost per instance.

4.11. Security

Security indicates how safe is a system in front of cyber attacks through the analysis of the mechanisms used to ensure safety. It determinates if the system is vulnerable or not.

To calculate this security metric, it is necessary to determine a security plan and its application. First, this security plan defines the different vulnerabilities and attacks that may occur in the system [30][31]. After listing vulnerabilities and attacks, it is necessary to weigh up it based on its criticality. Finally, the audition of the cloud grades the security of a cloud. An extensive explanation of the methodology to calculate this metric was considered out of the scope of this paper.

5. Metrics extraction

The next step is to take into account the abovementioned metrics for optimal resource allocation. It is necessary to define a threshold but the allocation of resources in a cloud or another is not trivial, especially if we are not using a single metric, but tackling a multi-criteria decision problem. In order to define a method that considers the different metrics, this paper proposes a decision tree. It can handle separately the different metrics, weighting and assigning them to higher or lower leaves depending on the relevance of the metric.

5.1. Parameters

Although the subset of metrics selected and presented in previous section, some of them required a greater level of delimitation. For example, the level of workload can be defined by many different aspects or measure units. Therefore, the different parameters that can be measurable and are related to the selected metrics were defined. These parameters are the ones that will be finally used to define a decision tree that allocates a specific resource in a cloud or another.

The chosen parameters for this calculation are related with the workload and performance of the virtual machines:

- *Used Memory*: Amount of main memory used for all the system processes in the moment of measure.
- *Total Memory*: Total memory of the virtual machine.
- *Used Swap*: Amount of memory swap used for all the system processes in the moment of measure.
- *Total Swap*: Amount of memory swap available in the virtual machine.
- *Used Disk Space*: Amount of space in disk used in the moment of measure.

- *Total Disk Space*: Total space of the virtual machine.
- *Processor*: Number of processors that the virtual machine has.
- *CpuMHz*: Speed of the system processors.
- *Load Average*: Load Average in the moment of measure, in %.
- *CPU*: Used CPU in the moment of measure.
- *Number Processes*: Number of processes working in the moment of measure.
- *Connected Users*: Number of users connected to the system in the moment of measure.
- *RX/TX Bytes*: Total number of bytes received/transmitted through all the physical interfaces of the virtual machine.
- *RX/TX Packets*: Total number of packets received/transmitted through all the physical interfaces of the virtual machine.
- *RX/TX Errors*: Total number of reception/transmission errors detected.
- *RX/TX Drop*: Total number of packets discarded at reception/transmission.
- *RX/TX FIFO*: Total number of FIFO's error at reception/transmission.
- *RX/TX Frame*: Total number of framing errors at reception/transmission.
- *RX/TX Multicast*: Total number of multicast frames received/transmitted.

The testbed scenario proposed in FINESCE project was a stable lab test, meaning that it does not have problems with the response time. Hence, in this specific case, it was decided not to use the metrics related with the response time: Average Weighted Response Time, Average Time to Deploy an Application, Latency, and Jitter because of the fact that there is not a high critical demand of new nodes deployment as it could be in other environments (e.g Cloud Providers). In addition, Security was not taken into account in the computation of cloud decision.

The partners agreed that the selected metrics are the most suitable for the project after the analysis of all of them, because of the feasibility to monitor and extract them at real-time or near real-time level in each of the virtual machines and the cloud as a whole. The next step was getting real data provided by the FINESCE scenario to generate the decision tree for cloud decision service location.

5.2. Getting real data

Before measuring data, it was needed to generate activity in the testbed scenario. In order to create this activity, three scripts were run in the six virtual machines that compose the scenario. The goal of these scripts is to maintain the scenario active, very close to the collapse, emulating a high activity environment. These scripts were executed in loop, creating random

binary files, sending them to other machines in the network and after sending them, deleting the files. They are designed to be simple, but perform tasks that mean a high load of computation and network transactions in order to stress the system. The intention is to emulate critical conditions where a high amount of data is generated by the Smart Grid, stored in the nodes in different substations and it is replicated among them.

In addition to that, a program that collects the subset of selected parameters from each virtual machine was needed. To get this information, we used a script containing the commands to measure the selected metrics, and *crontab* [32] command to call this script every minute and save the information in a *.csv* file.

To sum up the testing process, once we executed the load generation scripts and the six virtual machines are working and sending random files between them, we also activated the *crontab* command in each machine executing the parameter measurement script periodically, and generating the *.csv* file. Table 2 shows the status of virtual machine metrics in a particular instant of time after monitoring these metrics during one week.

Table 2. Metrics sample obtained from a virtual machine.

Hour	Min	Sec	Used Mem	Total Mem	Used Swap	Total Swap	Used Disk	Total Disk	Processor
18	37	1	401	489MB	0	507MB	4691752	15596512	2

cpuMHz	Load Average	CPU	Num Processes	Connected Users	RXbytes	RXpackets	RXerrs	RXdrop
4.018.211	0,67	131	1	77	229247755	261735	0	1042

RXfifo	RXframe	RXCompressed	RXmulticast	TXbytes	TXpackets	TXerrs	TXdrop	TXfifo
0	0	0	145	1510565165	228891	0	0	0

TXframe	TXCompressed	TXMulticast
0	0	0

If you look at Table 2 more closely, you will notice some relevant information about the virtual machine. For example, the 82% of the memory is used and the load average is around 67%. Moreover, you can check that there are no errors in transmission or reception. Finally, it is shown that there are 77 users connected to that FIDEV.

6. Results obtained and decision tree generation

The different generated .csv files were gathered and combined and formatted in a single *arff* file that can be processed by *Weka* [33] tool generating summarized statistics of the collected information. For example, Figure 2 shows the statistics of the used total swap memory of each virtual machine. Some analysis could be made from this simple capture. On the one hand, all the machines have the same number of swap memory (507MB), but some nodes (e.g. PROXY1/ FIDEV1 and PROXY2/FIDEV2) barely use their swap memory (less than 9MB). On the other hand, some storage-dedicated ones, only with storage functionalities, such as STORAGE2 and STORAGE3, used at least 100MB of their swap memory. It is concluded that the used memory in Storage nodes is greater (+10/15%) than the memory used in FIDEV nodes, showing the role of each type of node and how the memory is used by them.

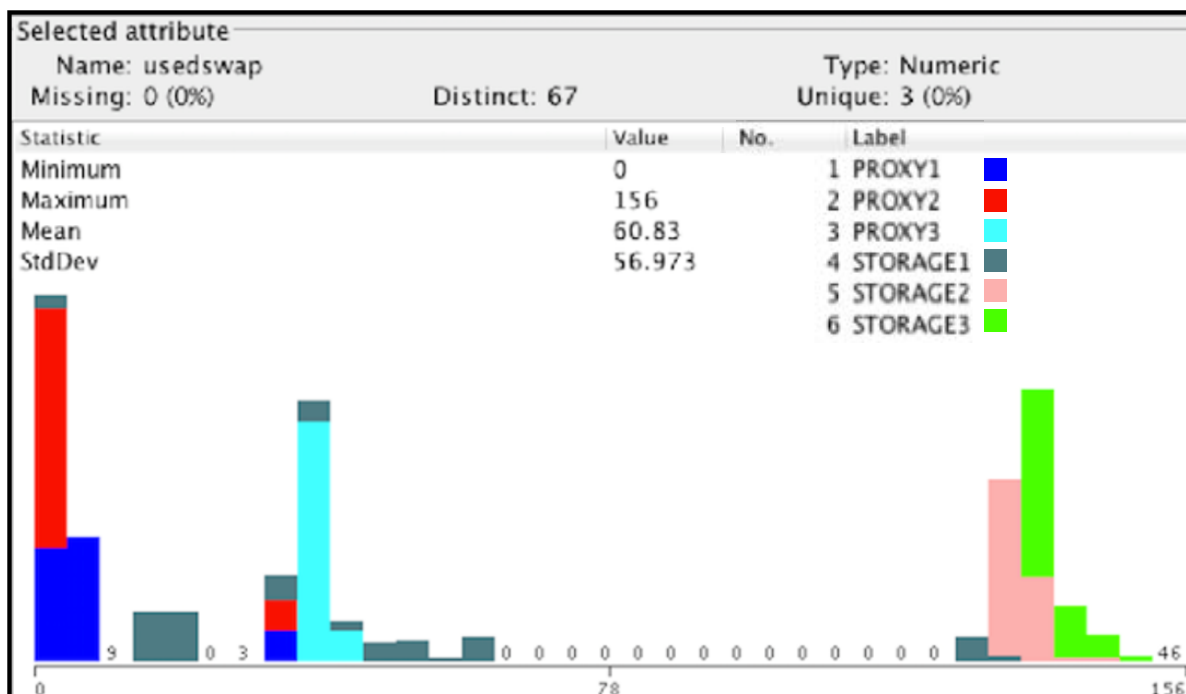


Figure 2. Swap Memory statistics gathered from the virtual machines of the scenario

Weka tool is also able to process all this collected data and generate a decision tree. Among the different options that it offers to generate it, C4.5 algorithm [34] was selected. It was preferred because it can withstand empty attributes and accepts discrete and continuous attributes, getting trees with a coherent size, and pruning the repetitive or pointless forks of the tree. In this way, based on the information collected, it automatically suggested discarding some metrics with less influence in the generation of the final decision tree. The six metrics taken into account by *Weka* were: *Used Swap Memory*, *TX Bytes*, *Connected Users*, *RX Bytes*, *Used CPU*, and *Used Disk Space*. These metrics provide the performance of each virtual machine and the information required to understand the scenario as well.

In Figure 3, it is shown the graphic generated by the *Weka* tool. It has established the threshold values for each metric that determines which is the most suitable cloud.

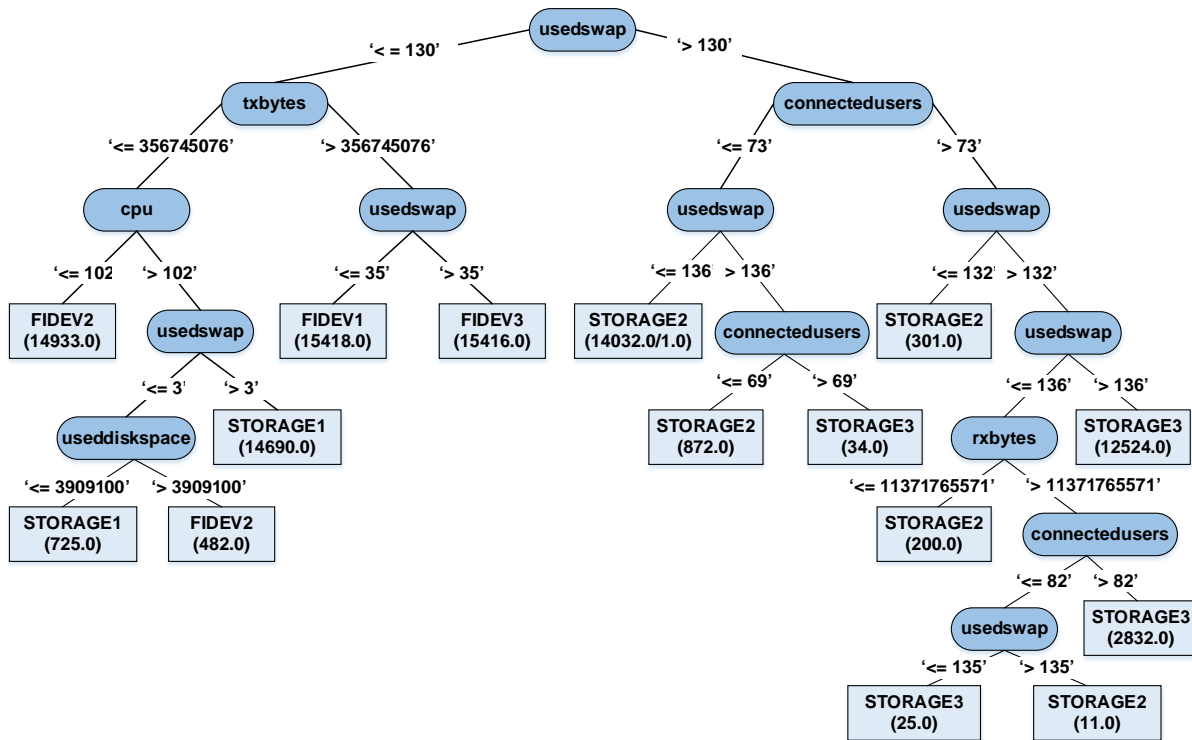


Figure 3. Decision tree generated for cloud and virtual machine selection in FINESCE testbed

Additionally, we can map the different thresholds to the different virtual machines, showing the different rules to be applied in order to the location of the resource in a virtual machine or another (Table 3).

Table 3. Values thresholds and ranges of the different metrics for resource allocation decision

Virtual Machines	Metrics					
	Used Swap	TXBytes	CPU	Used Disk	Connected Users	RXBytes
FIDEV 1	$35 > x$	$x > 356.745.075$	-	-	-	-
FIDEV 2	$130 > x$	$356.745.077 > x$	$102 > x$	-	-	-
	$3 > x$	$356.745.077 > x$	$x > 102$	$x > 3,909,788$	-	-
FIDEV 3	$130 > x > 35$	$x > 356.745.075$	-	-	-	-
STORAGE 1	$130 > x > 3$	$356.745.077 > x$	$x > 102$			
	$3 > x$	$356.745.077 > x$	$x > 102$	$3.909.788 > x$		

Virtual Machines	Metrics					
	<i>Used Swap</i>	<i>TXBytes</i>	<i>CPU</i>	<i>Used Disk</i>	<i>Connected Users</i>	<i>RXBytes</i>
STORAGE 2	$136 > x > 130$				$73 > x$	
	$x > 136$				$69 > x$	
	$132 > x > 130$				$x > 73$	
	$136 > x > 132$				$x > 73$	$11.371.765.571 > x$
STORAGE 3	$x > 136$				$73 > x > 69$	
	$x > 136$				$x > 73$	
	$136 > x > 132$				$x > 82$	$x < 11.371.765.571$
	$135 > x > 132$				$82 > x$	$x < 11.371.765.571$

With the decision tree and other statistics processed through *Weka*, the scenario is better understood and it is possible to optimize resources. Capturing the instant values of these parameters, the load that an operation will generate is approached. Therefore, applying the rules specified in Table 3, the virtual machine more suitable in a specific moment is selected.

7. Conclusions

This paper presents several contributions beyond the current state of the art. After summarizing the vantages and disadvantages of public, private and hybrid clouds, it presents the specific case of the Smart Grid, in which a cloud can provide an infrastructure to store data generated from different elements of the grid. The proposal of a hybrid cloud solution provides flexibility to use its private side for more sensitive data, with the possibility to move historical data to nodes in other clouds. Moreover, we present a more versatile approach for the private cloud, built by a set of distributed nodes or replicating that are spread in different locations from the utility (e.g. secondary substations). The access to this information must be carefully controlled with permissions, while the security to data migrated to other clouds could be much more relaxed.

The complexity of this solution is on the optimal resources allocation among the different clouds and nodes. It requires an orchestrator which decides the best location for services or data. This decision has to be based on a set of rules (preferences, priorities, costs, etc.) depending on the characteristics of the cloud.

This paper presented a characterization of the cloud based in a set of metrics, which give information about its behavior. Regarding to the result of the measurement of these metrics, the set of rules could be different. Therefore, a table that classifies and qualitatively evaluates multiple metrics is presented. It was build through the experience and expertise obtained from different FINESCE partners. Contrasted among different research groups, utilities and cloud providers, it allows cloud operators to characterize and evaluate a cloud performance. It can give guidelines about which metrics should be used and which strategy to follow for resource allocation, depending on the cloud operation, type of service or application required. This table helps to decide which metrics are the optimal metrics to evaluate a cloud, considering the services used in the cloud or demanded by a user.

After that, this paper also proposed a subset of metrics that were selected for the FINESCE project specific testbed, according the relevant information that can be gathered in the designed Smart Grid scenario (emulating a Smart Grid at electricity distribution level). An explanation and an analysis of each selected metric are also presented, including the measurement units for each one. For the final scenario, the metrics measured are mostly related to workload and performance of virtual machines.

To emulate an environment with high activity, it was necessary to generate this activity on the scenario. Three specific scripts were executed in loop to achieve this objective. With this activity generated, data was extracted from each machine of the scenario measuring the selected metrics. For the characterization of the testbed, *Weka* tool became a good option to generate and visually review statistics. It also allowed building a decision tree in order to select in which node a resource should be allocated. It is also proposed to apply C4.5 algorithm due to its characteristics (e.g. accepting empty, discrete and continuous attributes) using all the data gathered in the FINESCE specific testbed. Before applying for a specific operation in the cloud, it allows to decide which the most suitable cloud to allocate the resources. Although the decision tree and the associated allocation decision rules presented in the paper are valid for the FINESCE testbed scenario proposed, the general methodology can be applied for other scenarios, gathering previously a large and consistent amount of data, and recreating the decision tree and the table of rules. The paper can guide the own authors or other researchers in the future for facilitating and speeding up the design of other Hybrid Cloud resource allocation recommendation systems, using the experience and the results presented, and taking profit of the analyzed metrics, the ones finally selected, and the proposed methodology.

Acknowledgement

This work was carried out within the framework of FINESCE project, funded by the Future Internet Public Private Partnership Programme (FI-PPP) of the European Commission's 7th Framework Programme (ICT-2011, grant number 604677).

References

- [1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee et al. "A view of cloud computing." *Communications of the ACM* 53, no. 4, 50-58. 2010. <http://doi.acm.org/10.1145/1721654.1721672>
- [2] Smart Sm Chhetri, M.B.; Chichin, S.; Quoc Bao Vo; Kowalczyk, R., "Smart Cloud Broker: Finding your home in the clouds," in *Automated Software Engineering (ASE), 2013 IEEE/ACM 28th International Conference on*, vol., no., pp.698-701, 11-15 Nov. 2013. www.doi.org/10.1109/ASE.2013.6693136
- [3] P. Mell, T. Grance. "The NIST definition of cloud computing.", 2011. <http://doi.org/10.12691/jcsa-2-3-1>
- [4] Haag, S.; Eckhardt, A.; Kronung, J., "From the Ground to the Cloud -- A Structured Literature Analysis of the Cloud Service Landscape around the Public and Private Sector," in *System Sciences (HICSS), 2014 47th Hawaii International Conference on*, vol., no., pp.2127-2136, 6-9 Jan. 2014. <http://doi.org/10.1109/HICSS.2014.268>
- [5] J.Altmann, "Cost model based service placement in federated hybrid clouds", *Future Generation Computer Systems*, vol. 41, pp. 79-90, 2014. [doi:10.1016/j.future.2014.08.014](http://doi.org/10.1016/j.future.2014.08.014)
- [6] Kaur, R.; Kumar, P., "A Rule-Based Approach for Effective Resource Provisioning in Hybrid Cloud Environment" *International Journal of Computer Science and Informatics* ISSN (PRINT): 2231 –5292, Vol-1, Iss-4, 2012. http://doi.org/10.1007/978-3-642-35461-8_5
- [7] R. Van den Bossche, K. Vanmechelen, J. Broeckhove. "Cost-optimal scheduling in hybrid iaas clouds for deadline constrained workloads." In *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on*, pp. 228-235. IEEE, 2010. <http://doi.org/10.1109/CLOUD.2010.58>
- [8] F. Liu, J. Tong, J. Mao, R. Bohn, J. Messina, L. Badger, D. Leaf. "NIST cloud computing reference architecture." NIST special publication 500: 292. 2011.
- [9] Linh Manh Pham; Tchana, A.; Donsez, D.; de Palma, N.; Zurczak, V.; Gibello, P.-Y., "Roboconf: A Hybrid Cloud Orchestrator to Deploy Complex Applications," in *Cloud Computing (CLOUD), 2015 IEEE 8th International Conference on*, vol., no., pp.365-372, June 27 2015-July 2 2015. <http://doi.org/10.1109/CLOUD.2015.56>
- [10] FINESCE Project website. [Online: <http://www.finesce.eu/>, last accessed: 25/11/2015].
- [11] FI-PPP website. [Online: <https://www.fi-ppp.eu/>, last accessed: 25/11/2015].
- [12] FIWARE website. [Online: <https://www.fiware.org/>, last accessed: 25/11/2015].
- [13] Genez, Thiago AL, Luiz F. Bittencourt, and Edmundo RM Madeira. "On the performance-cost tradeoff for workflow scheduling in hybrid clouds." In *Proceedings of the 2013 IEEE/ACM 6th International Conference on Utility and Cloud Computing*, pp. 411-416. IEEE Computer Society, 2013. <http://dx.doi.org/10.1109/UCC.2013.82>

- [14] Song Li; Yangfan Zhou; Lei Jiao; Xinya Yan; Xin Wang; Lyu, M.R., "Delay-Aware Cost Optimization for Dynamic Resource Provisioning in Hybrid Clouds," in Web Services (ICWS), 2014 IEEE International Conference on , vol., no., pp.169-176, June 27 2014-July 2 2014. <http://www.doi.org/10.1109/ICWS.2014.35>
- [15] Chu, Hsuan-Yi, and Yogesh Simmhan. "Resource allocation strategies on hybrid cloud for resilient jobs." Small 1005.1 (2013): 0-065. [Online: <http://ceng.usc.edu/~simmhan/pubs/chu-usctr-2013.pdf>, last accessed: 25/11/2015].
- [16] Shifrin, M.; Atar, R.; Cidon, I., "Optimal scheduling in the hybrid-cloud," in Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on , vol., no., pp.51-59, 27-31 May 2013.
- [17] D5.7 – Trial Results FINESCE project [Online: http://www.finesce.eu/global/images/cms/Results/FINESCE_deliverable_%20D5.7_trial%20results_final.pdf, last accessed: 25/11/2015].
- [18] ESB website. [Online: <http://www.esb.ie>, last accessed: 25/11/2015].
- [19] Z. Li, L. O'Brien, H. Zhang, R. Cai. "On a catalogue of metrics for evaluating commercial cloud services". In Proceedings of the 2012 ACM/IEEE 13th International Conference on Grid Computing (pp. 164-173). IEEE Computer Society. September, 2012. <http://dx.doi.org/10.1109/Grid.2012.15>
- [20] Z. Li, L. O'Brien, R. Cai, H. Zhang, "Towards a taxonomy of performance evaluation of commercial Cloud services." In Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on (pp. 344-351). IEEE. June, 2012. <http://dx.doi.org/10.1109/CLOUD.2012.74>
- [21] L. Wang, U.S. Khan. "Review of performance metrics for green data centers: a taxonomy study." The Journal of Supercomputing, 63(3), 639-656. 2013. <http://dx.doi.org/10.1007/s11227-011-0704-3>
- [22] J.Y. Lee, J.W. Lee, S.D. Kim. "A quality model for evaluating software-as-a-service in cloud computing". In Software Engineering Research, Management and Applications, 2009. SERA'09. 7th ACIS International Conference on (pp. 261-266). IEEE. December, 2009. <http://dx.doi.org/10.1109/SERA.2009.43>
- [23] Z. Li, L. O'Brien, H. Zhang, R. Cai. "Boosting Metrics for Cloud Services Evaluation--The Last Mile of Using Benchmark Suites." In Advanced Information Networking and Applications (AINA), 2013 IEEE 27th International Conference on (pp. 381-388). IEEE. March, 2013. <http://dx.doi.org/10.1109/AINA.2013.99>
- [24] O. Mazhelis, "Role of Data Communications in Hybrid Cloud Costs", En 37th EUROMICRO Conference on Software Engineering and Advanced Applications, pp. 138-145, 2011. <http://dx.doi.org/10.1109/SEAA.2011.29>
- [25] M. Malawski, "Cost minimization for computational applications on hybrid cloud infrastructure", Future Generation Computer Systems, vol. 29, pp. 1786-1794, 2013.

<http://dx.doi.org/10.1016/j.future.2013.01.004>

[26] H. Zhang, "Proactive Workload Management in Hybrid Cloud Computing", IEEE Transactions On Network And Service Management, vol. 11, n. 1, pp. 90-100, 2014.

<http://dx.doi.org/10.1109/TNSM.2013.122313.130448>

[27] A. Khan, X. Yan, S. Tao, N. Anerousis. "Workload characterization and prediction in the cloud: A multiple time series approach", In Network Operations and Management Symposium (NOMS), 2012 IEEE, pp. 1287-1294. IEEE, 2012.

<http://dx.doi.org/10.1109/NOMS.2012.6212065>

[28] K. V. Vishwanath, N. Nagappan. "Characterizing cloud computing hardware reliability". In Proceedings of the 1st ACM symposium on Cloud computing, pp. 193-204. ACM, 2010. <http://dx.doi.org/10.1145/1807128.1807161>

[29] D. Hardy, M. Kleanthous, I. Sideris, A.G. Saidi, E. Ozer, Y. Sazeides, "An analytical framework for estimating TCO and exploring data center design space" Performance Analysis of Systems and Software (ISPASS), 2013 IEEE International Symposium on, vol., no., pp.54, 63, 21-23. April 2013. <http://dx.doi.org/10.1109/ISPASS.2013.6557146>

[30] Bhadauria R., Chaki R., Chaki N., Sanyal S., "A Survey on Security Issues on Cloud Computing", Acta Tehnica Corviniensis – Bulletin of Engineering, Tome VII, Fascicule 4, October – December 2014. <http://arxiv.org/abs/1109.5388>

[31] Venkata S., Padmapriya S., "A Survey on Cloud Computing Security Threats and Vulnerabilities", International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering, vol. 2, issue 1, January 2014.

[32] Crontab webpage. [Online: <http://crontab.org/>, last accessed: 25/11/2015].

[33] Weka is a software platform for machine learning and data mining written in Java. Weka Official Page. [Online: <http://www.cs.waikato.ac.nz/ml/weka/>, last accessed: 24/11/2015].

[34] Badr HSSINA, Abdelkarim MERBOUHA, Hanane EZZIKOURI and Mohammed ERRITALI, "A comparative study of decision tree ID3 and C4.5" International Journal of Advanced Computer Science and Applications(IJACSA), Special Issue on Advances in Vehicular Ad Hoc Networking and Applications 2014, 2014.

<http://dx.doi.org/10.14569/SpecialIssue.2014.040203>

Copyright Disclaimer

Copyright reserved by the author(s).

This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>).