

## Characterizing Cloud Federation for Enhancing Providers' Profit

Íñigo Goiri, Jordi Guitart, and Jordi Torres

*Universitat Politecnica de Catalunya and Barcelona Supercomputing Center*

*Jordi Girona 31, 08034 Barcelona, Spain*

*{igoiri, jguitart, torres}@ac.upc.edu*

**Abstract**—Cloud federation has been proposed as a new paradigm that allows providers to avoid the limitation of owning only a restricted amount of resources, which forces them to reject new customers when they have not enough local resources to fulfill their customers' requirements. Federation allows a provider to dynamically outsource resources to other providers in response to demand variations. It also allows a provider that has underused resources to rent part of them to other providers. Both things could make the provider to get more profit when used adequately.

This requires that the provider has a clear understanding of the potential of each federation decision, in order to choose the most convenient depending on the environment conditions. In this paper, we present a complete characterization of providers' federation in the Cloud, including decision equations to outsource resources to other providers, rent free resources to other providers (i.e. insourcing), or shutdown unused nodes to save power, and we characterize these decisions as a function of several parameters. Then, we demonstrate in the evaluation section how a provider can enhance its profit by using these equations to exploit federation, and how the different parameters influence which is the best decision on each situation.

**Keywords**-Cloud provider, outsourcing, federation, economic

### I. INTRODUCTION

Cloud Computing is an emerging style of computing in which applications, data, and IT resources are provided to users as services over the Internet rather than being locally on the user's machine. Users can access these services anytime and anywhere, avoiding in this way hardware acquisition costs, software licenses or upgrades management, etc. These services are typically deployed and executed on third-party companies that act as *service providers*. As stated in [1], the Cloud allows a service provider to virtualize its resources and dynamically provision them as unified computing resources based on a Service Level Agreement (SLA) established through negotiation between the service provider and the consumer.

In order to be profitable, service providers tend to share their resources among multiple concurrent services owned by different customers, but at the same time, they must guarantee that each service has always enough resources to meet the agreed performance goals. This requires of complex resource management mechanisms that could dynamically manage the provider's resources in the most cost-effective

way (e.g. maximizing their utilization or reducing their power consumption), while satisfying the QoS agreed with the customers.

Nevertheless, former resource management approaches for Cloud providers hindered their market potential by considering a limited amount of resources. In these approaches, if a service provider had not enough local resources to fulfill its customers' requirements, it should start denying the acceptance of new customers or canceling low priority services that were already running on the system. This has further implications than just losing the revenue from some services, because it also implies a loss of reputation and therefore a loss of future customers [2].

This problem can be overcome via Cloud federation [3]. Different providers running services that have complementary resource requirements over time can mutually collaborate to share their respective resources in order to fulfill each one's demand. For instance, a provider could outsource resources to other providers when its workload cannot be attended with its local resources. In this way, the provider would obtain higher profit because it can attend more customers. Of course, the expected revenue from these customers should be higher than the cost of outsourcing the additional resources in order to be worth doing it. Similarly, a provider that has underused resources could rent part of them to other providers. We refer to this situation as *insourcing* resources. In this way, the provider would improve its benefit, better exploit its resources, and compensate the cost of maintaining them. Again, the expected benefit from renting its resources should be higher than the cost of maintaining them operative. Otherwise, it would be preferable to shutdown them in order to save power (and thus reduce costs) [4].

From previous discussion, one could realize that the profitability of using Cloud federation for a service provider highly depends on a number of parameters, such as the provider's incoming workload, the cost of outsourcing additional resources, the revenue for renting unused resources, or the cost of maintaining the provider's resources operative. All these parameters must be taken into account to decide the most adequate resource management action for the provider depending on current (and foreseen) environment conditions. Depending on their value, the provider could decide at every moment whether to outsource resources, insource resources,

or shutdown them.

In order to obtain the maximum benefit from Cloud federation, and given the described complexity of federation decisions, it is important that the provider has a clear understanding of the potential of each federation decision, in order to choose the most convenient depending on the environment conditions. In this paper, we analyze the impact of federation as a mechanism for maximizing Cloud providers' revenue in a scenario that federates Private and Public Clouds [2]. By Private Clouds we mean, essentially, a Cloud computing capability dedicated to one organization having a limited capacity. We refer as Public Cloud the utility computing made available in a pay-as-you-go manner to the general public. We present an analytical model that characterizes Cloud federation and can be used to drive provider's decisions about resource outsourcing, insourcing, and node shutdown. We also study the effect of these decisions on the provider's profit and we evaluate the most appropriate provider's configuration depending on the environment conditions.

The remainder of this paper is organized as follows: Section II introduces the equations we propose for characterizing Cloud federation. Section III describes the experimental environment and the evaluation. Section IV presents the related work. Finally, Section V presents the conclusions of the paper and the future work.

## II. EQUATIONS CHARACTERIZING A FEDERATED CLOUD

In order to analyze the impact of using federation in Cloud providers, we envision a global *Scheduler* on each provider that is responsible for deciding the placement and the allocated resources for all the VMs running in that provider. This includes both the movements among the different nodes in that provider and between that provider and other federated Cloud providers. This *Scheduler* is in charge of deciding where a service will be executed and managing its location during the execution (e.g. migrations, cancellations, etc.). It decides about the placement of services in the nodes of the provider, and the amount of allocated resources to each of them in order to guarantee that they meet the agreed performance goals and trying to maximize provider's utility. This can include allocating additional resources from a federated Cloud provider, insourcing unused resources to other federated providers, or shutting down provider's unused nodes.

### A. Allocation within the provider

Currently, researchers are seeking to find effective solutions to make Cloud data centers reduce power consumption while keeping the desired quality of service. One approach consists of consolidating the maximum number of tasks in a single node in order to maximize its usage. In addition, this allows applying different techniques for reducing the

power consumption of the provider, such as Dynamic Voltage/Frequency Scaling (DVFS) and node shut down [5], [6].

In particular, the *Scheduler* can shut down nodes that remain unused in order to reduce power consumption in the provider, thus saving the costs of keeping them in idle state. Nodes can be shut down and restarted when needed using Wake-on-LAN. The *Scheduler* continuously monitors the node utilization in the provider, and shuts down or restarts nodes (or outsources new nodes to federated providers) in order to fit the customers' demand.

The expected profit for the provider drives the allocation decisions of the *Scheduler*. We define the profit obtained from executing tasks in a provider  $p$  during in a certain period  $\Delta t$  as  $Profit_p(\Delta t) = Revenue_p(\Delta t) - Cost_p(\Delta t)$ .

$Revenue_p(\Delta t)$  is obtained by multiplying the number of VMs running in the provider during that period of time  $VM_p(\Delta t)$  with its corresponding price (e.g.  $Price\_VM\_Hour$  of a small instance in Amazon EC2 is €0.085 per hour). Notice that  $VM_p(\Delta t)$  will depend on the provider's incoming workload.

$$Revenue_p(\Delta t) = VM_p(\Delta t) \cdot Price\_VM\_Hour \cdot \Delta t \quad (1)$$

$Cost_p(\Delta t)$  is defined as the cost of maintaining all the nodes in the provider up ( $Nodes_p \cdot Cost\_Node\_Hour_{var}$ ) during a certain period  $\Delta t$ . In addition, since shutting down idle nodes would reduce the costs for the provider, we add a factor to the formula ( $C_p(\Delta t)$ ), which indicates the capacity of the system (understood as the ratio of nodes that are up), in order to reflect this. If all the nodes in the system are up, capacity is 1. If the provider shuts down half of the nodes, capacity is 0.5. Finally, we add also some fixed costs per node ( $Cost\_Node\_Hour_{fix}$ ), which include the costs of acquiring the nodes and the physical space they occupy, taking into account their amortization.

$$Cost_p(\Delta t) = C_p(\Delta t) \cdot Nodes_p \cdot Cost\_Node\_Hour_{var} \cdot \Delta t + Nodes_p \cdot Cost\_Node\_Hour_{fix} \cdot \Delta t \quad (2)$$

In order to normalize the provider's incoming workload (i.e. the number of VMs to be executed), we define the provider's utilization ( $U_p(\Delta t)$ ). It is calculated using as reference the maximum number of VMs that the provider can host, which is given by  $Nodes_p \cdot VM\_Node$ , in the following way:

$$U_p(\Delta t) = \frac{VM_p(\Delta t)}{Nodes_p \cdot VM\_Node}$$

As discussed before, a single provider is profitable when  $Revenue_p(\Delta t) > Cost_p(\Delta t)$ . Using previous equations and operating on this formula, we obtain Equation 3, which establishes the relationship between the utilization (i.e. the

$$C_p(\Delta t) < \frac{U_p(\Delta t) \cdot VM\_Node \cdot Price\_VM\_Hour - Cost\_Node\_Hour_{fix}}{Cost\_Node\_Hour_{var}} \quad (3)$$

$$C_p(\Delta t) < \frac{U(\Delta t) \cdot (1 - \alpha + U_{ratio}(\Delta t) \cdot \alpha) \cdot VM\_Node \cdot Price\_VM\_Hour - Cost\_Node\_Hour_{fix}}{Cost\_Node\_Hour_{var}} \quad (4)$$

amount of VMs to execute) and the capacity (i.e. the ratio of nodes that are operative) for provider's profitability. Obviously, this and the subsequent equations require  $C_p(\Delta t)$  to be greater or equal than  $U_p(\Delta t)$ . This equation will allow the *Scheduler* to determine the number of nodes to shutdown ( $C_p(\Delta t)$ ) given the current workload ( $U_p(\Delta t)$ ) to get the best profit.

### B. Outsourcing to federated Clouds

As described in [2], outsourcing resources to federated providers can be preferable to overprovisioning a private data center when demand varies over time. In addition, it also allows the provider to insource its resources to other providers if these are not being used. The decision of using these capabilities is based on their economic viability. [2] introduces an equation that evaluates whether outsourcing resources to an external provider is profitable or not. It essentially compares the profit (resulting from  $Revenue - Cost$ ) for the provider when outsourcing external resources with respect to executing in its own resources.

Our analysis starts from this formula in order to decide grabbing additional resources when there is a resource demand that cannot be fulfilled using local resources. In particular, the additional revenue obtained when outsourcing resources  $Revenue_o(\Delta t)$ , which is shown in Equation 5, is calculated in the same way as Equation 1 and depends on the number of VMs that are outsourced ( $VM_o(\Delta t)$ ). Notice that, in this scenario, the total revenue for the provider is  $Revenue(\Delta t) = Revenue_p(\Delta t) + Revenue_o(\Delta t)$ .

$$Revenue_o(\Delta t) = VM_o(\Delta t) \cdot Price\_VM\_Hour \cdot \Delta t \quad (5)$$

The total cost for the provider in this scenario is  $Cost(\Delta t) = Cost_o(\Delta t) + Cost_p(\Delta t)$ . The cost of outsourcing ( $Cost_o(\Delta t)$ ) could be calculated also from Equation 5. However, we assume that the provider can buy these VMs cheaper than the revenue it obtains for selling them. According to this, we apply a factor  $\alpha$  to the cost of the VM, obtaining Equation 6.

$$Cost_o(\Delta t) = VM_o(\Delta t) \cdot \alpha \cdot Price\_VM\_Hour \cdot \Delta t \quad (6)$$

In this scenario, the provider is profitable when  $Revenue_p(\Delta t) + Revenue_o(\Delta t) > Cost_p(\Delta t) + Cost_o(\Delta t)$ . Notice that in this case  $U(\Delta t) = U_p(\Delta t) + U_o(\Delta t)$ , that is, it includes both the VMs executed in

the provider and the VMs outsourced to other providers. According to this, we define  $U_{ratio}(\Delta t) = \frac{U_p(\Delta t)}{U(\Delta t)}$ , which represents the ratio of incoming workload that is executed locally in the provider. Using previous equations, we can derive Equation 4, which allows to determine the number of nodes to shutdown ( $C_p(\Delta t)$ ) and the distribution of local and outsourced VMs ( $U_{ratio}(\Delta t)$ ) to get the best profit.

### C. Insourcing from federated Clouds

As commented before, in a federated environment the provider can offer its unused resources to other providers (i.e. insourcing). In this case, the total cost for the provider does not vary ( $Cost(\Delta t) = Cost_p(\Delta t)$ ). This means that there are not additional costs if the provider rents its free resources. However, the total revenue is expected to increase ( $Revenue(\Delta t) = Revenue_p(\Delta t) + Revenue_i(\Delta t)$ ). For calculating  $Revenue_i(\Delta t)$ , we use again the  $\alpha$  factor and, in addition, we include another factor ( $\beta$ ) that represents the ratio of free resources that can be offered. This serves to model the market demand of resources, since not all the resources can be always sold to external providers. This factor also allows the provider to reserve some free resources in order to be able to react to variations in its workload. According to this, the number of potential VMs that could be sold is defined as follows:

$$VM_{free} = (C_p(\Delta t) - U_p(\Delta t)) \cdot Nodes \cdot VM\_Node$$

Using this parameter,  $Revenue_i(\Delta t)$  can be calculated using Equation 9.

$$Revenue_i(\Delta t) = \beta \cdot VM_{free}(\Delta t) \cdot \alpha \cdot Price\_VM\_Hour \cdot \Delta t \quad (9)$$

Having the option to offer unused resources to other providers, or to shut down them to reduce power consumption, the provider could doubt on which is the more profitable decision. The answer comes from resolving the following inequation:  $Revenue_p(\Delta t) + Revenue_i(\Delta t) > Cost_p(\Delta t)$ . Again, using previous equations and operating on this formula, we obtain Equation 7, which allows to determine the number of nodes to shutdown ( $C_p(\Delta t)$ ) and the ratio of free resource to insource ( $\beta$ ) given the current workload ( $U_p(\Delta t)$ ) to get the best profit.

$$C_p(\Delta t) < \frac{U_p(\Delta t) \cdot (1 - \alpha \cdot \beta) \cdot VM\_Node \cdot Price\_VM\_Hour - Cost\_Node\_Hour_{fix}}{Cost\_Node\_Hour_{var} - \alpha \cdot \beta \cdot VM\_Node \cdot Price\_VM\_Hour} \quad (7)$$

$$C_p(\Delta t) < \frac{U(\Delta t) \cdot (1 - \alpha + U_{ratio}(\Delta t) \cdot \alpha \cdot (1 - \beta)) \cdot Price\_VM\_Hour \cdot VM\_Node - Cost\_Node\_Hour_{fix}}{Cost\_Node\_Hour_{var} - \alpha \cdot \beta \cdot VM\_Node \cdot Price\_VM\_Hour} \quad (8)$$

#### D. Insourcing and outsourcing in federated Clouds

The final step is putting all together. In this case, profitability occurs when  $Revenue_p(\Delta t) + Revenue_o(\Delta t) + Revenue_i(\Delta t) > Cost_p(\Delta t) + Cost_o(\Delta t)$ . After operating on this formula, we obtain Equation 8. Using this equation, the *Scheduler* can decide whether outsourcing resources, renting free resources to others providers, or shutting down nodes is profitable for the provider.

### III. EVALUATION

This section presents a simulation study about the potential benefit in a service provider when doing outsourcing and insourcing and shutting down nodes by means of the equations described in this paper. We follow the pricing idea for Grids presented in [7]. However, the numbers in this paper should not be interpreted absolutely. In fact, we use reference values for revenues, costs, and virtualization parameters just to demonstrate how our equations can drive resource allocation decisions, though the particular values of these parameters will highly depend on the real provider's characteristics. Anyway, the presented equations remain valid. As base node, we use a mid-range server with a direct consumption of 638 W in mean (see [8] and [9]). These mid-range servers support in mean a maximum amount of 6 VMs per node, assuming small EC2 instances, which have a cost of 0.085 €/hour (EC2 pricing in Europe).

Nevertheless, power consumption is not just the server direct consumption. It must also take into account all the related energy costs such as cooling and other infrastructure consumptions. This is evaluated using the Power Usage Effectiveness (PUE), which is defined as the ratio of data center power to IT power draw [10]. According to historical trends, site infrastructure consumes 50% of all data center energy, which corresponds to a PUE of 2.0 (this means that the data center must draw 2 Watts for every 1 Watt of power consumed by the IT equipment). Therefore, we assume an average consumption of 1276 W per node. The pricing used for the electricity is the Spanish one, which corresponds to 0.09 €/KWh [11].

Finally, in order to calculate the cost of the nodes, we also take into account the amortization of the servers (in 3 years) and the space (in 10 years) required to deploy them using a price of 4000 €/node and 2000 €/m<sup>2</sup>, respectively.

#### A. Profitability analysis in a federated Cloud

Taking into account previous information, we first calculate how a service provider should be dimensioned (i.e. its capacity) according to its utilization in order to be profitable by applying Equation 3. The results are displayed in Figure 1, showing the capacity as a function of the price of the VMs and the provider's utilization. In particular, high capacities are only profitable when the utilization is greater than 0.4 and the price per VM is higher than 0.05 €. Being below these values, the provider is not profitable since the fix costs are too high.

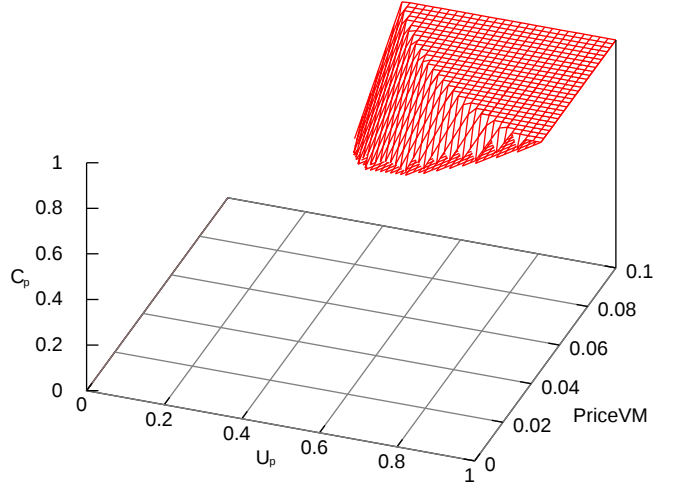


Figure 1. Relation between Utilization and Capacity in a single provider (Equation 3)

Figure 2 shows the maximum capacity a provider should have in order to be profitable when using outsourcing according to Equation 4. This figure relates the capacity with the provider's global utilization (i.e. the incoming workload) and the ratio of VMs that are locally executed, assuming that  $\alpha = 0.75$ . As shown in the figure, the higher the workload and the number of VMs locally executed are, the higher capacities are allowed.

Figure 3 shows the maximum capacity a provider should have to be profitable if it is able to insource resources according to Equation 7. It relates the capacity with the provider's utilization and the factor  $\beta$  (the ratio of free VMs that it sells), assuming an  $\alpha$  factor of 0.75. If the provider sells less than the 30% of its free resources, high capacities are only profitable when the utilization is greater than 50%.

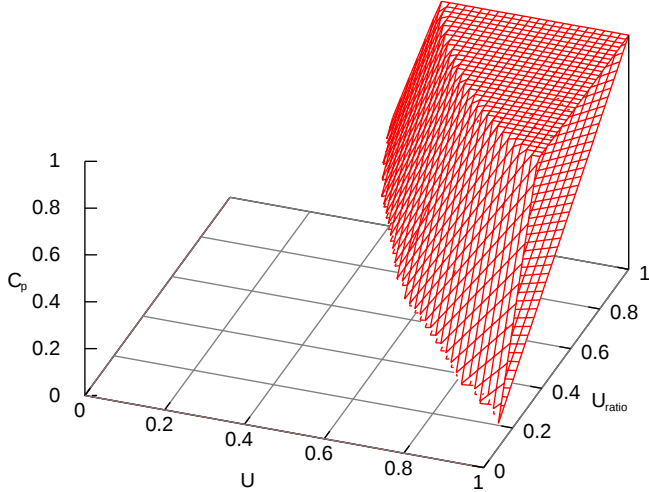


Figure 2. Relation between Utilization and Capacity using outsourcing (Equation 4)

On the other side, higher values of  $\beta$  and small utilizations allows also high capacities since the surplus resources can be sold to other providers.

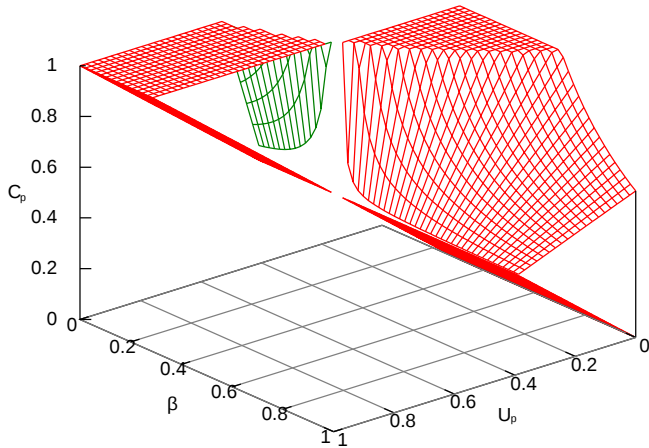


Figure 3. Relation between Utilization and Capacity when introducing insourcing (Equation 7)

Finally, using Equation 8, which models the provider taking into account both outsourcing and insourcing, we get Figure 4, which relates the capacity with the utilization and the ratio of local VMs assuming  $\alpha = 0.75$  and  $\beta = 0.5$ . It shows that having low provider's utilization (because the workload is low or it is being mostly outsourced) allows higher capacities since free VMs can be insourced.

### B. Potential benefit in a federated Cloud

In this section, we apply the formulas presented in Section II in order to calculate the potential benefits and costs of a service provider during a week with a real workload. This workload, which is shown in Figure 5, is extracted from a

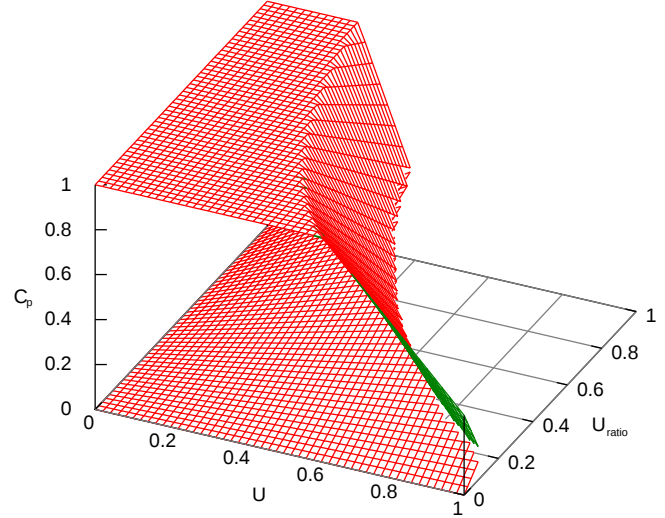


Figure 4. Relation between Utilization and Capacity when introducing insourcing and outsourcing (Equation 8)

provider's log during the week from Monday 27th of April until Monday 4th of May 2009.

The provider has a cluster with 100 nodes. However, as shown in the figure, sometimes the customers' demand is higher. Using a traditional resource management approach, the provider has to reject all the services that exceed its maximum capacity. Therefore, it loses many clients during rush hours. These lost clients can represent a great amount of money that is being wasted. In addition, the reputation of this provider is going down since customers stop trusting on it. For this reason, outsourcing resources to external providers can increase the provider's capacity when it is not enough to satisfy the demand. On the other side, the provider's capacity is underused during some periods. This reduces its total profit, since underused nodes are also consuming power. In order to avoid this, the provider can shut down those machines it guesses will not be required during a long period, for instance, during night. This decreases the power consumption during that period. Alternatively, the provider can also offer these unused resources to other providers, so they can execute their services on them. This option will be profitable for the provider when the obtained revenue is enough to compensate the cost of maintaining all these nodes up.

Of course, shutting down nodes, outsourcing, and insourcing can be jointly applied to maximize the provider's profit. In this section, we evaluate the impact on the profit when using these techniques. The results, which assume  $\alpha = 0.75$ ,  $\beta = 0.5$ , are displayed in Table I and demonstrate the benefit of outsourcing resources, which is inversely proportional with the  $\alpha$  factor. In addition, outsourcing allows the provider maintaining its reputation by being always available to give service to its customers. The second part of the table

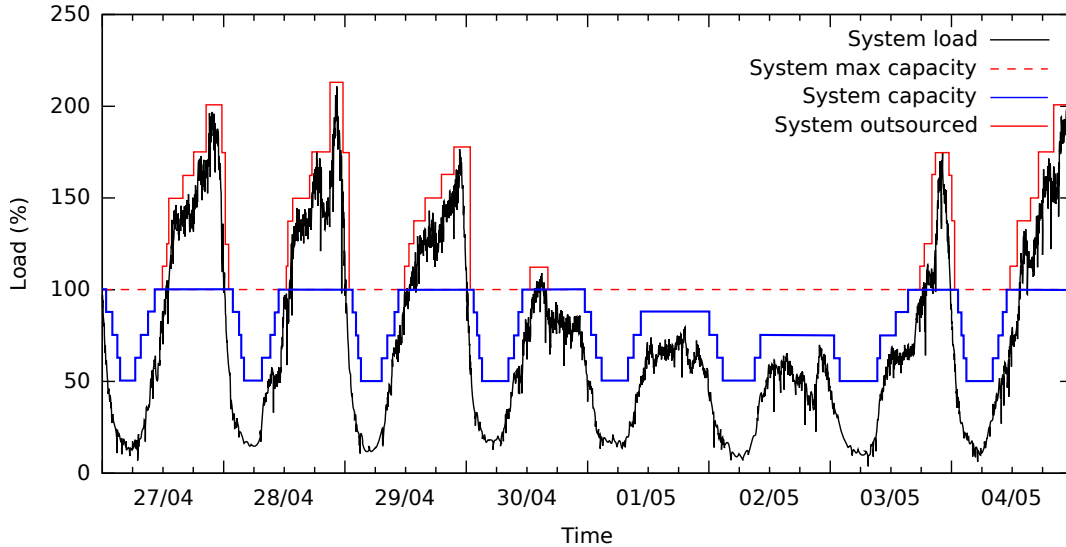


Figure 5. Service provider's workload during a week

presents the profit in case the provider decided to increase its maximum capacity (up to 200 nodes) in order to support the whole workload without using outsourcing. It shows that the revenue has increased regarding the previous table, but also the fixed costs, such as hardware and maintenance. For this reason, global profit is lower in this case. In fact, only the *Insource-Nodes always up* configuration is profitable in this case.

The values in these tables are graphically represented in Figures 6(a) and 6(b). The benefit from shutting down nodes can be clearly appreciated in the 'Typical' and the 'Outsourcing' configurations in Figure 6(a). The same argument applies in the 'Typical' configuration in Figure 6(b). In the configurations with 'In sourcing' in Figure 6(a), it is more profitable not to shut down nodes, because the provider has more resources to offer to other providers. This also applies to the 'In sourcing' configuration in Figure 6(b), though in this case the profit is noticeable smaller due to the increased fixed costs.

#### IV. RELATED WORK

Cloud computing has rapidly spread in recent times. [2] presents some key concepts of this paradigm such as the illusion of infinite computing resources available on demand and the ability to pay for use of computing resources on a short-term basis as needed. This allows companies to have a small set of resources that can be increased according to their needs, saving costs. [12] establishes how Clouds can be viewed as a logical continuation from Grids by providing a higher-level of abstraction. [1] defines Cloud computing and provides the architecture for creating Clouds with market-oriented resource allocation by leveraging technologies such as Virtual Machines (VMs). It also proposes ideas for inter-connecting Clouds for dynamically creating global Cloud

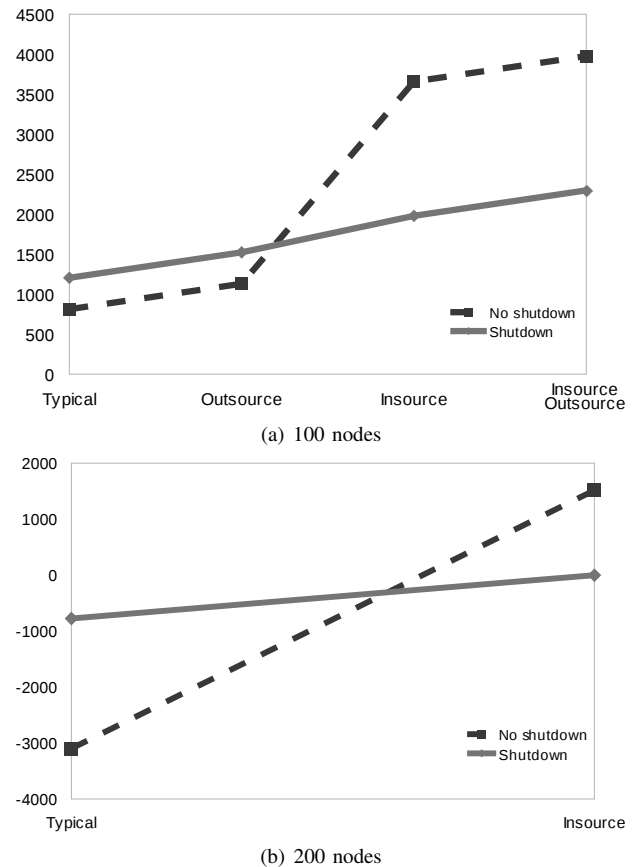


Figure 6. Comparison of provider's profit with different capacities

exchanges and markets and presents some representative Cloud platforms.

The idea of federating systems was already present in

	Nodes always up			Shutting down nodes		
	Revenue	Cost	Profit	Revenue	Cost	Profit
100 nodes						
Typical	5997.60	5185.84	811.76	5997.60	4793.41	1204.19
Outsource	7270.56	6140.56	1130.00	7270.56	5748.13	1522.43
Insource	8843.40	5185.84	3657.56	6771.29	4793.41	1977.88
Outsource & Insource	10116.36	6140.56	3975.80	8044.25	5748.13	2296.12
200 nodes						
Typical	7270.56	10371.68	-3101.12	7270.56	8048.17	-777.61
Insource	11888.1	10371.68	1516.42	8044.25	8048.17	-3.92

Table I  
SERVICE PROVIDER'S PROFIT IN EUROS

the Grid. For instance, works such as [13] and [14] use federation in order to get more resources in a distributed Grid environment. The application of federation in the Cloud was initially proposed within the Reservoir project. In particular, [3] describes the difficulty to merge different providers with different APIs and features. Nevertheless, they do not present any model to decide when to move tasks to a federated provider based on economic criteria. A first approach introducing this idea is presented in [15] where they state some factors such as provider occupation and maintaining costs in order to dimension a Cloud provider and when to outsource to a federated provider.

One of the strong points of Cloud federation is the possibility of incorporating public Clouds within the federation. There are a number of Cloud offerings that provide VMs on demand, being Amazon EC2 [16] probably the most popular. Nevertheless, it is a private implementation and it does not allow working with low-level aspects. Other public Cloud solutions with similar capabilities are GoGrid [17], 3Tera [18], and ElasticHosts [19].

In order to set up private Clouds with the same capabilities than public Clouds, different Cloud solutions that implement the EC2 API, such as Eucalyptus [20] or Nimbus [21], have appeared. Similarly, Aneka [22] is a .NET-based service-oriented resource management platform, which is based on the creation of containers that host the services and it is in charge of initializing services and act as a single point for interaction with the rest of the Aneka Cloud. Moreover, it provides SLA support such that the user can specify QoS requirements such as deadline and budget. Other open-source alternatives, such as AbiCloud [23], EMOTIVE Cloud [24], and OpenNebula [25], also add outsourcing capabilities by adding external resources.

## V. CONCLUSIONS

In this paper, we have presented a characterization of Cloud federation aimed at enhancing providers' profit. Our characterization includes equations that assist decisions in a federated Cloud, namely when to outsource resources to other providers, when to insource free resources to other

providers, and when to shutdown unused nodes to save power.

Our experimentation has evaluated these equations with real data in order to determine the impact of some parameters in the providers' profit. Evaluated parameters include the provider's incoming workload, the cost of outsourcing additional resources, the ratio of outsourced resources, the ratio of unused resources to be sold, and the cost of maintaining the provider's resources operative.

Our results demonstrate that the provider requires a minimum utilization and a minimum price per VM in order to be profitable when all the nodes are operative. In addition, local resources are preferred over outsourced resources, though the latter can enhance the provider's profit when the workload cannot be supported locally. Furthermore, when the utilization is low, the best option for the provider is insourcing the unused resources (though this is not always possible). We can summarize that all the described actuations can have a positive impact on the provider's profit depending on the environment conditions.

Based on these results, we plan to develop a scheduler for a real Cloud system that uses the presented characterization in order to take resource management decisions.

## ACKNOWLEDGMENTS

This work is supported by the Ministry of Science and Technology of Spain and the European Union (FEDER funds) under contract TIN2007-60625 and grant AP2008-0264, by the Generalitat de Catalunya (2009-SGR-980).

## REFERENCES

- [1] R. Buyya, C. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility," *Future Generation Computer Systems*, vol. 25, no. 6, pp. 599–616, 2009.
- [2] M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "Above the Clouds: A Berkeley View of Cloud Computing," *University of California at Berkeley, Tech. Rep. EECS-2009-28*, 2009.

- [3] B. Rochwerger, D. Breitgand, E. Levy, A. Galis, K. Nagin, I. Llorente, R. Montero, Y. Wolfsthal, E. Elmroth, J. Caceres *et al.*, “The RESERVOIR Model and Architecture for Open Federated Cloud Computing,” *IBM Journal of Research & Development*, vol. 53, no. 4, 2009.
- [4] E. Elnozahy, M. Kistler, and R. Rajamony, “Energy-Efficient Server Clusters,” in *2nd Workshop on Power-Aware Computing Systems, PACS 2002, Cambridge, MA, USA, February 2*. Springer, 2002, pp. 179–196.
- [5] T. Horvath, T. Abdelzaher, K. Skadron, and X. Liu, “Dynamic Voltage Scaling in Multi-tier Web Servers with End-to-end Delay Control,” *IEEE Transactions on Computers*, vol. 56, no. 4, pp. 444–458, 2007.
- [6] J. Heo, D. Henriksson, X. Liu, and T. Abdelzaher, “Integrating Adaptive Components: An Emerging Challenge in Performance-Adaptive Systems and a Server Farm Case-Study,” in *28th IEEE International Real-Time Systems Symposium (RTSS 2007), Tucson, AZ, USA, 3–6 December, 2007*, pp. 227–238.
- [7] A. Opitz, H. König, and S. Szamlewska, “What Does Grid Computing Cost?” *Journal of Grid Computing*, vol. 6, no. 4, pp. 385–397, 2008.
- [8] J. Koomey, “Estimating Total Power Consumption by Servers in the US and the World,” *Oakland, CA: Analytics Press, February 15, 2007, 2007*.
- [9] J. Koomey, “Worldwide electricity used in data centers,” *Environmental Research Letters*, vol. 3, no. 034008, 2008.
- [10] C. Belady, A. Rawson, D. Pfeuger, and S. Cader, “The Green Grid Data Center Power Efficiency Metrics: PUE and DCiE,” *White Paper. The Green Grid Consortium.*, October 23 2007, <http://www.thegreengrid.org/en/Global/Content/white-papers/The-Green-Grid-Data-Center-Power-Efficiency-Metrics-PUE-and-DCiE>.
- [11] “Europe’s energy portal,” <http://www.energy.eu>.
- [12] S. Jha, A. Merzky, and G. Fox, “Using clouds to Provide Grids with Higher Levels of Abstraction and Explicit Support for Usage Modes,” *Concurrency and Computation: Practice and Experience*, vol. 21, no. 8, pp. 1087–1108, 2009.
- [13] B. Boghosian, P. Coveney, S. Dong, L. Finn, S. Jha, G. Karniadakis, and N. Karonis, “NEKTAR, SPICE and Vortronics: using federated grids for large scale scientific applications,” *Cluster Computing*, vol. 10, no. 3, pp. 351–364, 2007.
- [14] M. Sobolewski and R. Kolonay, “Federated Grid Computing with Interactive Service-oriented Programming,” *Concurrent Engineering*, vol. 14, no. 1, p. 55, 2006.
- [15] R. Campbell, I. Gupta, M. Heath, S. Ko, M. Kozuch, M. Kunze, T. Kwan, K. Lai, H. Lee, M. Lyons *et al.*, “Open Cirrus™ Cloud Computing Testbed: Federated Data Centers for Open Source Systems and Services Research,” in *USENIX Workshop on Hot Topics in Cloud Computing (HotCloud), San Diego, CA, USA, June 14-19, 2009*.
- [16] “Amazon Elastic Compute Cloud,” <http://aws.amazon.com/ec2>.
- [17] “GoGrid,” <http://www.gogrid.com/>.
- [18] “3Tera,” <http://www.3tera.com/>.
- [19] “ElasticHosts,” <http://www.elastichosts.com/>.
- [20] D. Nurmi, R. Wolski, C. Grzegorzczuk, G. Obertelli, S. Soman, L. Youseff, and D. Zagorodnov, “The Eucalyptus Open-source Cloud-computing System,” in *9th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID), 18–21 May, Shanghai, China, 2009*, pp. 124–131.
- [21] “Nimbus Science Cloud,” <http://workspace.globus.org/clouds/nimbus.html>.
- [22] X. Chu, K. Nadiminti, C. Jin, S. Venugopal, and R. Buyya, “Aneka: Next-Generation Enterprise Grid Platform for e-Science and e-Business Applications,” in *3rd IEEE International Conference on e-Science and Grid Computing, 10–13 December, Bangalore, India, 2007*, pp. 151–159.
- [23] “Abiquo,” <http://www.abiquo.com>.
- [24] “EMOTIVE Cloud,” <http://www.emotivecloud.net>.
- [25] “OpenNebula,” <http://www.opennebula.org>.