A Flexible Accounting Model for Cloud Computing

Igor Ruiz-Agundez, Yoseba K. Penya and Pablo G. Bringas

DeustoTech, Deusto Institute of Technology,
University of Deusto
Bilbao, Basque Country
{igor.ira,yoseba.penya,pablo.garcia.bringas}@deusto.es

Abstract—**Cloud computing is the new paradigm that has changed traditional computer business schemes: static, close, centralised, and proprietary methods cannot cope with the new requirements that have emerged. Still, this new scenario poses a number of opportunities to use and novel problems to be faced. Specifically, we focus on the accounting of cloud computing services. These may include relations between different service providers, user connections to different simultaneous services, and the need for new services to be incorporated into the accounting systems to enable emerging business models, and so on. Classic solutions fail to provide a proper answer as they were not specifically design for cloud computing. Against this background, we put forward a flexible accounting model that allows the deployment of cloud computing services to accomplish all the service providers' requirements. Furthermore, we validate the model by presenting its implementation on a real cloud computing service.**

Keywords- cloud computing; accouting; intelligent service; business models

I. Introduction

Cloud computing has lately become a fashionable term, one of those buzz-words that must appear in every cutting-edge tech-talk, covering it with a gaudy, modern, and cool wrapping. Nevertheless, the reality hidden by this recent disguise has somehow always been there: cloud computing is about moving applications that traditionally were locally run to be remotely executed on the Internet. In this way, Buyya R. et al define cloud computing as "a type of parallel and distributed system consisting of a collection of interconnected and virtualised computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements" [6].

Nevertheless, business models (the rationale of how an organization creates, delivers, and captures value) in cloud computing are yet to be defined. In this way, accounting is an essential fact that enables the execution of business models. As far as the Internet is concerned, accounting is already a mature discipline, and offers support to network management and cost recovery of the services offered by a provider.

In the case of cloud computing, each service provider has developed its own proprietary accounting solution according to a limited set of requirements. A change on the requirements (e.g. the apparition of a new service) implies a change on all the accounting system, and hence, the business model as well. This approach is not flexible and limits the expansion and growth of cloud computing.

Against this background, the contribution of this paper is two-fold. First, we present a flexible accounting model for cloud computing services. Second, as a proof of concept, we describe a possible implementation of the previous model running a specific service of cloud computing.

The remainder of this paper is structured as follows. Section II introduces the related work on Internet accounting, focusing on cloud computing services. Section III describes our accounting model for cloud computing services. Section IV presents a case study for the model of the previous section, illustrating how it could be implemented. Finally, Section V concludes and draws the avenues of future work.

II. Internet Services Accounting

Accounting of the Internet joint many concepts from several disciplines such as economics or engineering. Section II-A introduces how the accounting is performed in the Internet, detailing all the functions that could be involved. Section II-B specifies the accounting requirements that are common to all the services in the Internet, including those of cloud computing. Section II-C defines the existing pricing schemes, the basic blocks that form the business models. Finally, Section II-D describes the specifications and the existing work in cloud computing accounting.

A. Accounting as a process

Internet services accounting is one of the challenges of service providers. Since it involves several motions and terminology that normally are not clearly defined, the semantics of some terms (e.g. pricing, charging or billing) is not always properly defined leading to confusion [30] [1].

Accounting has been applied to many areas, for instance, Wi-Fi connections [10], VoIP services [28], mobile networks [15], grid services [19], micro-payments [26], and packet-switched networks [14], among others.

In a previous work [29], we defined a taxonomy that outlined the accounting process on the Internet. This taxonomy introduced a common vocabulary regarding Internet accounting, which helps to develop accounting models. Figure 1 summaries this integrated vision of the accounting process. It begins with the registration of a *resource (or service) usage* by the metering function. Simultaneously, the mediation function generates *accounting records*, and, if required, it will also adapt their format.

These records are then transformed into *session records* by an accounting function. Afterwards, the pricing function defines the formula that specifies how to price *session records* on the charging function. This charging will generate *charge records* used by the billing function. Finally, the bill is sent to a financial clearing function to proceed to the payments.

Once settled a clear accounting process that unifies and controls the vocabulary involved on the Internet accounting, we focus on developing the accounting requirements of new systems, applications and systems such as cloud computing.
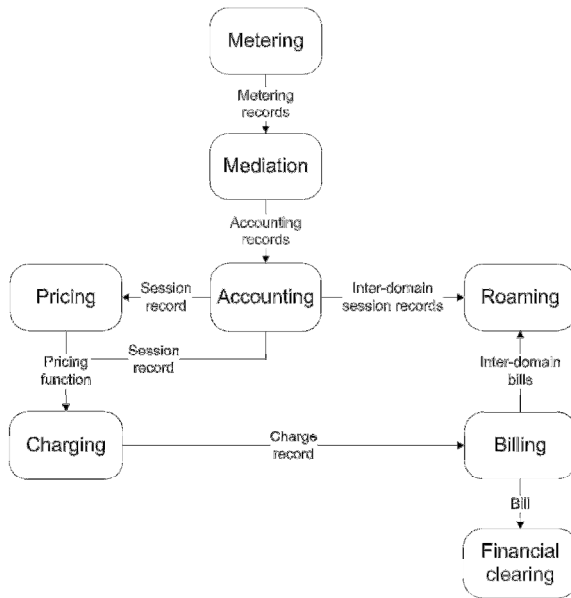

Figure 1. An integrated vision of the accounting process.

B. Accounting requirements

Multiple service providers may take part simultaneously during the same session in cloud computing (e.g. service request or data transfer) with users roaming through several networks and service providers. This fact implies that, consequently, a significant number of different charging schemes may also be involved simultaneously within a single session.

In this way, cloud computing demands new working paradigms and new billing requirements to suit all the requirements that providers may have. Finding a billing model for cloud computing is a real challenge due the operator structure and processes, being a key factor for their correct deployment. Despite the fast evolution of technologies, many service providers think that classic billing systems are enough for the emerging networks [3], [31]. Their use implies adapting them with the subsequent possibility of revenue reductions, system instability and slow new service introduction.

Within traditional models, the billing system operates in batch mode. It stores the so-called *Usage Detail Record*

(UDR)-s and in this period there is a time window in which the provider has no control over the user behaviour. There should not be this time window since it may leave records of services unprocessed. This is why real-time billing is preferred: it allows to process information every time it is generated, enabling immediate business support systems operations. The real-time efficiency will depend on the available processing time and the available processing cost.

Further, traditional models present a centralised architecture: all UDRs are processed by only one rating engine and one invoicing engine that limit the scalability of the system. On the contrary, cloud computing requires high scalability levels in order to support a large number of customers and inter-carrier settlement activities. With a centralised architecture, data can be accessed from any business application. The processing time, however, is long and management complicated. On the other hand, with a distributed architecture, the system loads are balanced but maintaining consistency between different instances increases the complexity of the system.

Moreover, classical billing systems are designed service-specific. That is, traditional systems lack of flexibility and are not capable of accommodating content-based pricing that will replace current flat-rate charging models. Due to this flaw, in case a new service is introduced, it would require a new billing system. Cloud computing will have heterogeneous data from a larger number of systems. The end user requires a convergent billing system that provides a unified view of the services that it consumes.

Further, it requires complex pricing capabilities fitting the emerging business models of cloud computing [11]. Specifically, the system should support any definition of the so-called *Service Level Agreement (SLA)* between the user and the provider. Additionally, such needs include security (all data and processes need to be reliable), adaptability (new services and business models may emerge), multi-service and multi-domain capability (in order to use a unified accounting model), and deployment possibility (being able to integrate it into an existing accounting system).

C. Pricing schemes

Pricing schemes are part of the accounting process. We are going to study them in greater detail due to their relevance in the business model. They provide the price of the users' usages and are represented as a formula that expresses the pricing function.

This formula consists of the pricing variables (consumption measure metrics of the session records) and several pricing coefficients [34].

Furthermore, these charging schemes may use different types of services [8]:

- Time-based: pricing based on how long a service is used.
- Volume-based: pricing based on the volume of a metric (e.g. downloaded bytes).
- QoS-based: pricing depends on the hired quality of service.
- Flat-rate: a fixed tariff for a specified amount of time.

- Paris-Metro pricing: used for shared resources. Resources are split by the amount of users per split.
- Priority pricing: services are labelled and priced according to their priority.
- Smart market pricing: services are priced in an auction.
- Edge pricing: calculation is done based on the distance between the service and the user.
- Responsive pricing: charging is activated only on service congestion.
- Effective bandwidth pricing: charging is based on an expected usage function.
- Proportional fairness pricing: it is according to the user's willingness to pay and service optimization costs.
- Cumulus pricing: based on flat pricing and dynamically priced by using a credit point system.
- Session-oriented: based on the use given to the session.
- One-off charge per service: one charge per service session.
- Usage-based: based on the general use of the service for a period of time, e.g. a moth.
- Content-based: based on the accessed content.
- Location-based: based on the access point of the user.
- Service type: based on the usage of the service.
- Differentiation on time-of-day: pricing based on the hour when the service is used.

In addition, we have the following related concepts:
- Free of charge: no charge is applied for the services.
- Periodical fess: payment of time to time quantities for the use of a service.
- Discounts: reduction in the usual price
- Pre-paid: the payment of the service is done in advance.
- Post-paid: the payment of the service is done after the use.
- Online: the accounting performed while the user makes use of a service.
- Offline: the accounting process is done after a service is used.
- Static pricing: the pricing function does not change.
- Dynamic pricing: the pricing function changes on the fly, being adapted to the usage of the users.

Furthermore, many of these schemes can be combined to create new ones that inherit the properties of the originals. The possibilities are endless as new pricing schemes can be created to model more complex business models.

In the case of cloud computing, due to its peculiarities, the most used pricing schemes so far are time-based and utility-based pricing (charge by allocation) [33] [6]. Time-based schemes in cloud computing pricing varies from service to service but the business formula is always the same: multiplying a fix price by a consumption time. In the case of utility-based pricing in cloud computing, the scheme may also vary from service to service. Please note that, it is not the same to model a *Software as a Service (SaaS)* provider or a *Infrastructure as a Service (IaaS)* provider. In the first one, SaaS, the utility is the service itself and consumption time is normally measured in number of uses. As for IaaS, time is measured in hours.

On the other hand, utility-based scheme charges the user on a per-use basis and its complexity relies in controlling the operating costs.

These models have been working properly in this early state of cloud computing. Business requirements, however, are changing and the introduction of other pricing schemes is essential. In order to enable this process in a flexible way, new engineering efforts are required.

D. Accounting in cloud computing

Accounting in cloud computing is a recent discipline. Nevertheless, there have been many attempts to find a model that fits all the accounting requisites.

For instance, the Diameter protocol [7] provides an *Authentication, Authorization and Accounting (AAA)* framework to work with. It allows connecting, authorizing and accounting of any component. There is an extension of this protocol that also supports charging options, the *A4C* protocol [18].

The *Common Reliable Accounting for Network Element (CRANE)* protocol enables efficient and reliable delivery of any kind of data, and mainly accounting data from Network Elements to any systems, such as mediation systems and *Business Support Systems (BSS)/ Operations Support Systems (OSS)* [32]. This protocol is designed to manage data from network elements, and not cloud computing services specifically.

The *Management Information Base (MIB)* protocol provides a framework for the exchange of messages which convey management information between the agents and the management stations [23]. This MIB protocol has been used in cloud computing to manage the usage information generated by the services [13].

Furthermore, there are other resource management and billing frameworks that have focused on presenting distributed resource usage metering as well as an accounting and account balancing mechanism, for instance, the Condor/G [17], Nimrod/G [4], GRASP [16], Tivoli [22], or TeraGrid [9].

It is worth mentioning existing solutions of cloud computing accounting do not normally allow situations where the providers are discovered and selected on the fly [27]. That is, there is not support of providers roaming. Nevertheless, the model we put forward here can implement roaming capabilities.

Still, none of these attempts have met all the requirements of cloud computing accounting for many different reasons. Some of them were not specifically designed for the cloud while others just consider some of the requisites and were not an integrated solution.

III. CLOUD COMPUTING ACCOUNTING MODEL

Our goal is to develop a flexible model that enables accounting of cloud computing usage. Within this model,

any use of cloud services, platforms or infrastructure can be accounted by the provider in accordance with the consumption of the users.

Ideally, any event that a user performs during the use of a service must be represented on the accounting system. Normally each service and each provider defines the usage record in a different and proprietary format. Therefore, it is necessary to use an open standard to handle any type of records.

In this section, we introduce the cloud computing services that can be accounted, a specification to format usage record of service usages, and an accounting platform to implement all the requirements of the model.

A. Accountable services in cloud computing

Cloud computing can be structured in different layers, considering its different functionalities [5]. The first layer involves the cloud clients that enable the access to the cloud. Examples include some computers, phones and other devices, operating systems, and browsers. The next layer is formed by the applications, also known as *Software as a Service (SaaS)*, and delivers software as a service over the Internet. There is no need to install or run the applications in the client equipment, and hence, the support and maintenance is simplified. A layer beyond, we find the platforms or *Platform as a Service (PaaS)*. They provide the possibility to deploy applications without the cost and work that implies managing the required hardware and software. In the upper layer, we find the infrastructure services or *Infrastructure as a Service (IaaS)*. Here, the provider offers outsourcing of servers, software, data-centre space or network equipment via a platform of environment

virtualization. Finally, in the last layer we find servers that are specifically designed for the delivery of cloud services.

Each of these layers generates different information that can be considered as *usage records*. These records can be classified as communication data, computational data or information data [18]. Unfortunately, each layer offers different type of information to the provider augmenting the complexity of the business support systems operations such as accounting. Furthermore, each layer can be implemented using different products and technologies from various providers and manufacturers.

We have to face a heterogeneous scenario in which each service presents different usage records both in format and in content. Next section introduces a standard that aims at solving these heterogeneity problems offering a standardised way of formatting usage records of any service, including those of cloud computing.

B. Internet Protocol Detail Record (IPDR)

The *Internet Protocol Detail Record (IPDR)* has the potential to become the standard protocol for exchanging service usage and for managing control information between IP networks, hosting elements and operations or business support systems. It provides a standardised framework that enables network and service accounting comprehensively [25].

This standard is defined by the Internet Protocol Detail Record Organization and the *TeleManagement Forum*. It is designed to enable cost-effective usage measurement and exchange for services across the entire value chain.

Figure 2 shows the IPDR reference model. The data flow is similar to the one introduced in Section II. A. The user
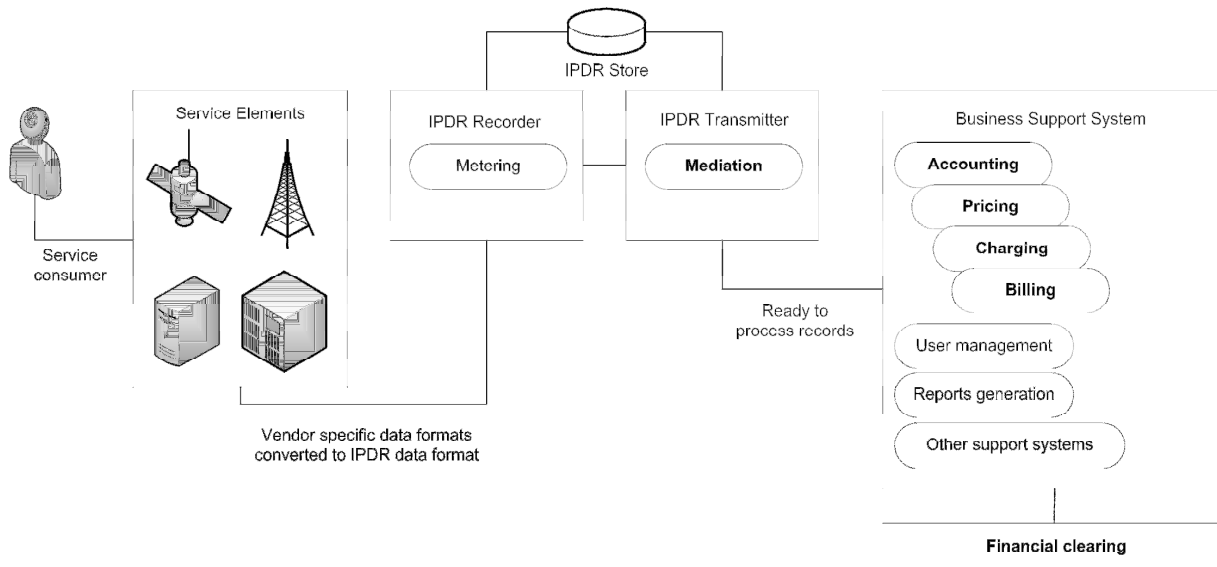

Figure 2.   IPDR reference model.

consumes a service and the metering function collects these records in IPDR format through the *IPDR Recorder*. These records can be stored o transmitted to the business support system by the mediation function using the *IPDR Transmitter*. The rest of the accounting process functions are allocated in the business support system, similarly to other functions such as user management, report generation, fraud management, error management or network elements configuration [2].

Moreover, the IPDR covers the billing requirements of cloud computing providing converged billing, avoiding provider dependence, and reducing the required interfaces. Real-time billing is allowed, and therefore charging is performed faster. Finally, as it offers great flexibility to define new services, it can be adapted to any emerging requirement of cloud computing accounting.

According to its service specifications [24], the IPDR is capable of collecting usage characteristics of any IP-based network or application service. All service specifications have five common attributes in their records. The first one describes the person in charge of the usage of a service, defining user identification. The second attribute tells when a certain service is used. The third attribute defines what service is being measured (e.g. quality of service, state information, event codes, connection state, etc.). The next attribute contents information to allow traceability by providing context, source, and destination, defining in this way the place the service is consumed. The final attribute informs about the reason that triggered the event.

So far and officially supported, the IPDR specifies services for Internet Protocol Television (IPTV), Public Wireless LAN (WLAN) access, Streaming Media (SM), Voice over IP (VoIP) or any other service specified by the service designer.

In this paper, we focus on service specification of a cloud computing service defining our own specification for this use case. This will give us a standard matter to format these cloud computing service usage records. Our methodology can be extended to incorporate any other cloud service.

C. An accounting platform

We propose the use of *jBilling* as an accounting platform [36]. It answers to all the requirements of Internet accounting as defined in Section II-B. Specifically, it is supports functions of mediation, those included in the business support systems, and the financial clearing as highlighted in Figure 2.

Table 1 shows a comparative of jBilling characteristics versus accounting platforms. This solution gives the flexibility to centre on the provider business model, there is no need to develop an *ad hoc* solution with the subsequent costs. Further, it offers freedom from proprietary solution dependencies. This fact gives us a full control of the accounting platform that can be adapted to the requirements of each service and provider.

Additionally, the system is ready to process big quantities of records that could be generated during the use of certain services, such as telephony. The system can manage to reach thousands of events per second. jBilling can handle these

records in batch mode, in real time or combining both. Mediation is possible through files, data bases, or applying the principles of the Software *Oriented architecture (SOA)* enabling the integration with other systems. Besides, each function can work independently in different systems, and hence, scalability is guaranteed.

TABLE I. ACCOUNTING PLATFORM COMPARISON

Comparison criteria	Accounting platforms		
	Own development	*Proprietary solution*	*jBilling*
Availability of source code	Yes	No	Yes
Complex business rules	Just the required	Yes	Yes
Licence cost	No	Yes	No
Development Cost	Very high	Included	No
Adaptability	Yes	No	Yes
Knowledge of area	No	Yes	Yes
Maintenance cost	High	Medium-High	Medium

Source: jBilling documentation

The mediation module can handle any type of service records: telephone calls, text messages, download, VoIP, IPTV, streaming video, etc. Any service that generates usage records can be implemented in this accounting platform, including cloud computing.

The pricing and charging functions use a *Business Rules Management System (BRMS)*. These rules may be able to use all the information defined in the usage records. The billing function is also flexible, enabling all types of options. Billing live cycle can be performed per month o at a fixed period, by user o group of users. More advanced options such as discounts, offers and promotions are also possible to use.

IV. A SYSTEM FOR CLOUD COMPUTING ACCOUNTING

At this point, we have introduced all the puzzle pieces. We have presented the accounting process, its requirements and the existing pricing schemes. More over, we have showed which cloud computing services can be accounted and which data format should be used to represent the usage records of these services. Finally, we have introduced an accounting platform that makes use of the information provided by the rest of the components enabling cloud computing accounting.

A. An implementation example

As aforementioned in Section III-A the different possible services available in cloud computing are vast. For the sake of simplicity, we are going to focus on an IaaS implementation. More accurately, we are going to use the Eucalyptus solution [20].

Eucalyptus is a software solution that implements scalable IaaS-style in private and hybrid clouds. The Eucalyptus architecture is highly modular with internal components consisting of Web services, which make them easy to replace and expand. Eucalyptus' flexibility enables it

to export a variety of APIs towards users, via client tools. Currently, Eucalyptus implements the Amazon Web Service (AWS) API, which allows interoperability with existing AWS-compatible services and tools. This also allows Eucalyptus users to group resources drawn both from an internal private cloud and external public clouds to form a hybrid cloud.

This solution provides some facilities that allow monitoring of the running components, virtual machines and storages. Specifically, it can integrate the *Ganglia* Resource Usage information gatherer [35]

Ganglia is a scalable distributed monitoring system for high-performance computing systems. It is based on a hierarchical design with federations of clusters in mind. It leverages widely used technologies such as XML for data representation, XDR for compact, portable data transport, and RRDtool for data storage and visualization [12]. This solution has many other characteristics; however, those are the ones we are using in our experiment. The data represented in this XML files gives a full report of the state of the resources informing the use the client makes of them.

The resource usage records include attributes of CPU, memory, network, cache, disk or load use. In particular, the attributes are the following: Boottime, Bytes-in, Bytes-out, Cpu-aidle, Cpu-idle, Cpu-nice, Cpu-num, Cpu-report, Cpu-speed, Cpu-system, Cpu-user, Disk-free, Disk-total, Gexec, Load-fifteen, Load-five, Load-one, Load-report, Machine-type, Mem-buffer, Mem-cached Mem-free, Mem-report, Mem-shared, Mem-total, Mtu, Network-report, Os-name, Os-release, Packet-report, Part-max-used, Pkts-in, Pkts-out, Proc-run, Proc-total, Swap-free, Swap-total, Sys-clock.

Further, these attributes can be extended with new metrics via *gmetric* extension [21]. This monitoring solution may reach aspects such as transaction rate of a database, number of databases, number of disks, used TCP ports or any other attribute that could have interest when defining a business model.

B. Flexible adaptation to our system

In this section we are going to show the deployment of a cloud computing service in a flexible way using our proposed model.

Thanks to Eucalyptus we may offer a cloud computing service. We also have Ganglia, which provides us with usage records in a proprietary format.

Now we have to convert these records to the IPDR data format to be able to manage them according to our reference model. This can be automatically performed with a small converter. Figure 3 shows an example of a record in IPDR format representing Eucalyptus usage records information gathered by Ganglia.

In a regular use scenario, many records are generated. Therefore, we consider that real-time accounting is more adequate. In this way, the period between resource usage records generation and its accounting is the shortest possible. The system manages the records in real time the system towards a *Simple Object Access Protocol (SOAP)*, since it allows invoking the mediation function at any time and place.

Once the platform has completed the mediation function the accounting process starts. This process will represent the business model of the system provider through the pricing and charging functions. As mentioned before, these functions use a BRMS system that has records as the one introduced in Figure 3 as input. Then, the rule-based system implements, executes, monitors and maintains the complexity of the business rules.

For instance, suppose that the business model is based on a time-based and usage-based pricing scheme. In concrete, the usage measures the number of CPUs used by the user. With these rules, the BRMS will check the values of the *CPUs* and *Uptime* attributes of the IPDR record.

Then, the pricing function will give a monetary value to charge the resource usage and the flow of the accounting process will continue.

Finally, the accounting platform generates a bill and sends it to the financial clearing functions, thus completing all functions of the accounting process.

In order to use other pricing schemes, new rules in the BRMS could be designed. These rules will use the required metrics and generate new pricing functions.

```
<IPDRCreationTime>2010-10-17T13:11:06.031+02:00</IPDRCreationTime>
    <seqNum>1</seqNum>
    <subscriberID>i1.Millennium.Berkeley.EDU</subscriberID>
    <Location>Unknown</Location>
    <heartbeat>19 seconds</heartbeat>
    <Uptime>27 days, 14:36:33</Uptime>
    <Load>0.02;0.01;0.00</Load>
    <CPU-Utilization-user>1.1</CPU-Utilization-user>
    <CPU-Utilization-sys>0.1</CPU-Utilization-sys>
    <CPU-Utilization-idle>98.7</CPU-Utilization-idle>
    <CPUs>2</CPUs>
    <CPU-power>2.99 Ghz</CPU-power>
    <Memory-RAM>1.98 GB</Memory-RAM>
    <Local-Disk>96.621 of 137.380 GB</Local-Disk>
    <Most-Full-Disk-Partition>70.4%</Most-Full-Disk-Partition>
    <OS>Linux 2.6.26-2-686-bigmem (x86)</OS>
    <Booted>September 20, 2010, 10:57 am</Booted>
    <Uptime>27 days, 14:36:33</Uptime>
    <Swap>Using 0.0 of 7812.8 MB swap.</Swap>
    <Bytes-in>720</Bytes-in>
    <Bytes-out>5.5</Bytes-out>
    <Cpu-aidle>99.9</Cpu-aidle>
    <Cpu-idle>99.9</Cpu-idle>
    <Cpu-nice>0</Cpu-nice>
    <Cpu-system>110</Cpu-system>
    <Cpu-user>1.25</Cpu-user>
    <Disk-free>41</Disk-free>
    <Disk-total>140</Disk-total>
    <Mem-buffer>170</Mem-buffer>
    <Mem-cached>420</Mem-cached>
    <Mem-free>1.4</Mem-free>
    <Mem-shared>0</Mem-shared>
```

Figure 3.   A record in IPDR format showing eucalyptus usage records information gathered by ganglia

V.   CONCLUSIONS

We have introduced a flexible model to account cloud computing services in a comprehensive and homogeneous way. This model allows the deployment of business models of service providers.

This model is based on the accounting process of the Internet and its requirements. It can fit any pricing scheme and any service of cloud computing through the use of IPDR

and the jBilling accounting platform. Finally, we have introduced a demonstration of the model in a system that implements an IaaS application.

This flexibility enables the operators to easily apply different pricing schemes (e.g. usage-based, volume-based, time-based, flat-rate, quality-of-service-based, etc.) and tariffs (a concrete pricing function for a determined pricing scheme), while the impact of change is smaller than with traditional accounting models. On turn, it also enables better accounting on the usage of the cloud computing.

Furthermore, this model could be expanded by including other business support systems such as client management or network operations.

Future work will focus on testing more methodologically the versatility of the model testing performances, response times and revenue calculations that have been proposed. In addition, new specifications of cloud computing services in IPDR are expected.

REFERENCES

[1] Airasian, P. W., Cruikshank, K. A., Mayer, R. E., Pintrich, P. R. and Wittrock, J. R. M. C. , Andersonand, L. W. and Krathwohl, D. R., (eds.) A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives, Allyn & Bacon, 2000.

[2] Baluja, W. and Llanes, A. "Estado actual y tendencias del enfrentamiento del fraude en las redes de telecomunicaciones," Ingeniería Electrónica, Automática y Comunicaciones (XXVI), 2005, pp. 46-52.

[3] Bihina Bella, M. A., Eloff, J. H. P. and Olivier, M. S. "Using the IPDR standard for NGN billing and fraud detection", in Hein S Venter, Jan H P Eloff, L. L. and Eloff, M. M., ed.,'Proceedings of the Fifth Annual Information Security South Africa Conference (ISSA2005)', Sandton, South Africa, Research in progress paper, http://mo.co.za/abstract/ipdrfraud.htm, 2005.

[4] Buyya, R., Abramson, D. and Giddy, J. "Nimrod/G: An architecture for a resource management and scheduling system in a global computational grid", Published by the IEEE Computer Society, 2000, pp. 283.

[5] Buyya, R., Pandey, S. and Vecchiola, C. "Cloudbus toolkit for market-oriented cloud computing," Cloud Computing, 2009, pp. 24-44.

[6] Buyya, R., Yeo, C., Venugopal, S., Broberg, J. and Brandic, I. "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," Future Generation Computer Systems (25:6), 2009, pp. 599-616.

[7] Calhoun, P., Loughney, J., Guttman, E., Zorn, G. and Arkko, J. "Diameter base protocol," .

[8] Choi, M.-J. and Hong, J. W.-K. "Towards Management of Next Generation Networks," IEICE Transaction Communications (E90-B:11), 2007, pp. 3004-3014.

[9] Dahan, M., Roberts, E. and Boisseau, J. "TeraGrid User Portal v1. 0: Architecture, Design, and Technologies"'International Workshop on Grid Computing Environments', 2007.

[10] Detal, G., Leroy, D. and Bonaventure, O. "An adaptive three-party accounting protocol", Proceedings of the 5th international student workshop on Emerging networking experiments and technologies, ACM, 2009, pp. 3-4.

[11] Elmroth, E., Marquez, F., Henriksson, D. and Ferrera, D. "Accounting and billing for federated cloud infrastructures"'2009 Eighth International Conference on Grid and Cooperative Computing', IEEE, 2009, pp. 268-275.

[12] Fenn, M., Murphy, M., Martin, J. and Goasguen, S. "An Evaluation of KVM for Use in Cloud Computing"'Proc. 2nd International Conference on the Virtual Computing Initiative, RTP, NC, USA', 2008.

[13] Han, H., Kim, S., Jung, H., Yeom, H. Y., Yoon, C., Park, J. and Lee, Y. "A RESTful Approach to the Management of Cloud Infrastructure," International Conference on Cloud Computing, 2009, pp. 139-142.

[14] Karsten, M., Schmitt, J., Stiller, B. and Wolf, L. "Charging for Packet-switched Network Communication - Motivation and Overview," Computer Communications (23), 2000, pp. 290-302.

[15] Koutsopoulou, M., Kaloxylos, A., Alonistioti, A., Merakos, L. and Kawamura, K. "Charging, accounting and billing management schemes in mobile telecommunication networks and the internet," IEEE Communications Surveys (6:1), 2004, pp. 50-58.

[16] Kwon, O., Hahm, J., Kim, S. and Lee, J. "GRASP: a grid resource allocation system based on OGSA". Proceedings of the 13th IEEE International Symposium on High performance Distributed Computing, IEEE, 2004, pp. 278--279.

[17] Litzkow, M., Livny, M. and Mutka, M. "Condor-a hunter of idle workstations" 8th International Conference on Distributed Computing Systems, IEEE, 2002, pp. 104--111.

[18] Meier, R. "Design and Implementation of a Charging Component and a Formal Tariff Specification for Grid Services".

[19] Morariu, C., Waldburger, M. and Stiller, B. "An Integrated Accounting and Charging Architecture for Mobile Grids" 3rd International Conference on Broadband Communications, Networks and Systems, 2006., 2006, pp. 1-10.

[20] Nurmi, D., Wolski, R., Grzegorczyk, C., Obertelli, G., Soman, S., Youseff, L. and Zagorodnov, D. "The eucalyptus open-source cloud-computing system"'Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid', IEEE Computer Society, 2009, pp. 124-131.

[21] Planning, C. "Operations," United States Government.

[22] Press, S. "IBM Tivoli Usage and Accounting Manager V7.,".

[23] Presuhn, R., Case, J., McCloghrie, K., Rose, M. and Waldbusser, S. "Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)", STD 62, RFC 3418, December 2002, 2002.

[24] Programme, I. "IPDR Business Solution Requirements", TeleManagement Forum, Technical report, TMForum, 2009.

[25] Programme, I. "IPDR Service Specification. Design Guide", TeleManagement Forum, Technical report, TMForum, 2008.

[26] Párhonyi, R. "Micro payment gateways", 2005.

[27] Rings, T., Caryer, G., Gallop, J., Grabowski, J., Kovacikova, T., Schulz, S. and Stokes-Rees, I. "Grid and Cloud Computing: Opportunities for Integration with the Next Generation Network," Journal of Grid Computing (7:3), 2009, pp. 375-393.

[28] Ruiz-Agundez, I., Penya, Y. K. and Bringas, P. G. "Tarificacion flexible de servicios en Internet" Proceedings of the XX Jornadas de Telecom I+D, Telefonica, Valladolid, Spain.

[29] Ruiz-Agundez, I., Penya, Y. K. and Bringas, P. G. "A Taxonomy of the Future Internet Accounting Process"'Proceedings of the Fourth International Conference on Advanced Engineering Computing and Applications in Sciences', Florence, Italy, 2010, pp. 111-117.

[30] Stiller, B., Fankhauser, G., Plattner, B. and Weiler, N. "Pre-study on "Customer Care, Accounting, Charging, Billing, and Pricing"," Computer Engineering and Networks Laboratory TIK, ETH Zurich, Switzerland, Pre-study performed for the Swiss National Science

Foundation within the "Competence Network for Applied Research in Electronic Commerce, 1998.

[31] Takuji, T. "Backend systems Architectures in the age of the next generation network," NEC Technical Journal (1:2), 2006, pp. 51-55.

[32] Zhang, K. and Elkin, E. "XACCT's Common Reliable Accounting for Network Element (CRANE) Protocol Specification Version 1.0,", 2002.

[33] Zhang, Q., Cheng, L. and Boutaba, R. "Cloud computing: state-of-the-art and research challenges" Journal of Internet Services and Applications (1:1), 2010, pp. 7-18.

[34] Zseby, T., Zander, S. and Carle, C. "RFC3334: Policy-Based Accounting," Internet RFCs, 2002.

[35] Massie, M.L. and Chun, B.N. and Culler, D.E., "The ganglia distributed monitoring system: design, implementation, and experience," Parallel Computing (30:7), Elsevier, 2004, pp. 817-840.

[36] Offermann, P. and Hoffmann, M. and Bub, U., "Benefits of SOA: Evaluation of an implemented scenario against alternative architectures," 13th Enterprise Distributed Object Computing Conference Workshops, 2009. EDOCW 2009., IEEE, 2009, pp. 352-359.