2011 Fourth IEEE International Conference on Utility and Cloud Computing

SMICloud: A Framework for Comparing and Ranking Cloud Services

Saurabh Kumar Garg*, Steve Versteeg† and Rajkumar Buyya*
*Cloud Computing and Distributed Systems (CLOUDS) Laboratory
Department of Computer Science and Software Engineering
The University of Melbourne, Australia
Email: sgarg @csse.unimelb.edu.au
†CA Technologies
Melbourne, Australia
Email:Steve.Versteeg@ca.com

***Abstract*—With the growth of Cloud Computing, more and more companies are offering different cloud services. From the customer's point of view, it is always difficult to decide whose services they should use, based on users' requirements. Currently there is no software framework which can automatically index cloud providers based on their needs. In this work, we propose a framework and a mechanism, which measure the quality and prioritize Cloud services. Such framework can make significant impact and will create healthy competition among Cloud providers to satisfy their Service Level Agreement (SLA) and improve their Quality of Services (QoS).**

***Keywords*-Cloud Computing, Service Measurement, Quality of Service**

I. Introduction

Cloud computing is a new paradigm for delivering on-demand resources (e.g., infrastructure, platform, software, etc.) to customers similar to other utilities (e.g., water, electricity and gas). The current Cloud computing architecture enables three layers of services [1]. Firstly, Software as a Service (SaaS) provides access to complete applications as a service, such as Customer Relationship Management (CRM). Secondly, Platform as a Service (PaaS) provides a platform for developing other applications on top of it, such as the Google App Engine (GAE). Finally, Infrastructure as a Service (IaaS) provides an environment for deploying, running and managing virtual machines and storage. Technically, IaaS offers incremental scalability (scale up and down) of computing resources and on-demand storage.

Due to several business benefits offered by Cloud computing, many organizations have started building applications on Cloud infrastructure and making their businesses agile by using flexible and elastic Cloud services. But moving applications and/or data into the Cloud is not straight forward. Numerous challenges exist to leverage the full potential that Cloud computing promises. These challenges are often related to the fact that existing applications have specific requirements and characteristics, that need to be met by Cloud providers.

Other than that, with the growth of public Cloud offerings, for Cloud customers it has become increasingly difficult to decide which provider can fulfil their Quality of Service (QoS) requirements. Each Cloud provider offers similar services at different prices and performance levels with different set of features. While one provider might be cheap for offering tera-bytes of storage, renting powerful VMs from them might be expensive.

Therefore, given the diversity of Cloud service offerings, an important challenge for customers is to discover who are the "right" Cloud providers that can satisfy their requirements. Often, there may be trade-offs between different functional and non-functional requirements fulfilled by different Cloud providers. This makes it difficult to evaluate service levels of different Cloud providers in an objective way such that required quality, reliability or security of an application can be ensured in Clouds. Therefore, it is not sufficient to just discover multiple Cloud services but it is also important to evaluate which is the most suitable Cloud service.

In this context, the Cloud Service Measurement Index Consortium (CSMIC)[2] has identified measurement indexes that are combined in the form of Service Measurement Index (SMI) and important for evaluation of a Cloud service. These measurement indexes can be used by customers to compare different Cloud services. In this paper, we are taking the work of this consortium one step further by proposing a framework (SMICloud) that can compare different Cloud providers based on user requirements. The SMICloud would let users compare different Cloud offerings, according to their priorities and along several dimensions, and select whatever is appropriate to their needs.

Several challenges are tackled in realizing the model for evaluating QoS and ranking Cloud providers. The first is how to measure various SMI attributes. Many of these attributes vary over time. For example, Virtual Machine (VM) performance has been found to vastly vary from the promised values in the Service Level Agreement (SLA) by Amazon [3]. However, without having precise measurement models for each attribute, it is not possible to compare different Cloud services or even discover them. Therefore, SMICloud uses historical measurements and combines them

978-0-7695-4592-9/11 $26.00 © 2011 IEEE
DOI 10.1109/UCC.2011.36

with promised values to find out the actual value of an attribute. We also give precise metrics for each measurable attribute.

The second challenge is how to rank the Cloud services based on these SMI attributes. There are two types of QoS requirements which a user can have: functional and non-functional. Some of them cannot be measured easily given the nature of the Clouds. Attributes like security and user experience are not even easy to quantify. Moreover, deciding which service matches best with all functional and nonfunctional requirements is a decision problem. It is necessary to think critically before selection as it involves multiple criteria and an interdependent relationship between them. This is a problem of multi-criteria decision-making (MCDM). Each individual parameter affects the service selection process, and its impact on overall ranking depends on its priority in the overall selection process. To address this problem, we propose an Analytical Hierarchical Process (AHP) based ranking mechanism to solve the problem of assigning weights to features considering interdependence between them, thus providing a much-needed quantitative basis for ranking of Cloud services.

The rest of paper is organized as follows. In the next section, we present an overview of SMI and its high level QoS attributes. Section III describes the SMICloud framework with its key components. Section IV shows how metrics for various quality attributes can be modelled. Section V presents the Cloud ranking mechanism which is explained by case study example in Section VI. Section VII concludes this article with some future works.

II. Service Measurement Index (SMI)

SMI attributes are designed based on International Organization for Standardization (ISO) standards by the Consortium [2]. It consists of a set of business-relevant Key Performance Indicators (KPI's) that provide a standardized method for measuring and comparing a business service. The SMI framework provides a holistic view of QoS needed by the customers for selecting a Cloud service provider based on : Accountability, Agility, Assurance of Service, Cost, Performance, Security and Privacy, and Usability. There are still no metrics or methods which define these KPIs and compare Cloud providers. This work is first effort in this direction. The following defines these high/top level attributes:

- Accountability - This group of QoS attributes is used to measure various Cloud provider specific characteristics. This is important to build trust of a customer on any Cloud provider. No organization will want to deploy its applications and store their critical data in a place where their is no accountability of security exposures and compliance. Functions critical to accountability, which SMI considers when measuring and scoring services, include auditability, compliance, data ownership, provider ethicality, sustainability etc.
- Agility - The most important advantage of Cloud computing is that it adds to the agility of an organization. The organization can expand and change quickly without much expenditure. Agility in SMI is measured as a rate of change metric, showing how quickly new capabilities are integrated into IT as needed by the business. When considering a Cloud service's agility, organizations want to understand whether the service is elastic, portable, adaptable and flexible.
- Cost - The first question that arises in the mind of organizations before switching to Cloud computing is that whether it is cost-effective or not. Therefore, cost is clearly one of the vital attributes for IT and the business. Cost tends to be the single most quantifiable metric today, but it is important to express cost in the characteristics which are relevant to a particular business organization.
- Performance - There are many different solutions offered by Cloud providers addressing the IT needs of different organizations. Each solution has different performance in terms of functionality, service response time and accuracy. These organizations need to understand through these properties how well their applications will perform on the different Clouds and whether these deployments meet their expectations.
- Assurance - This characteristic indicates the likelihood of a Cloud service that it will perform as expected or promised in the SLA. Every organization looks to expand their business and provide better services to their customers. Therefore, reliability, resiliency and service stability become an important factor for them before they decide switching to Cloud services.
- Security and Privacy - Data protection and privacy are the important concerns of nearly every organization. Hosting data in other organizations control is always a critical issue which require stringent security policies employed by Cloud providers. For instance, Financial organizations generally require high compliance regulations involving data integrity and privacy. Security and Privacy is also multi-dimensional in nature and include many attributes such as privacy, data loss and integrity.
- Usability - For fast usage of Cloud services, the usability plays an important role. The more easy to use and learn a Cloud service is, more faster an organization can switch to Cloud services. The usability of a Cloud service can depend on multiple factors such as Accessibility, Installability, Learnability, Operatibility.

III. SMICloud Architecture

We propose Service Measurement Index Cloud framework - SMICloud, which helps Cloud customers to find the most suitable Cloud provider and therefore can initiate SLAs.

SMICloud framework provides features such as service selection based on Quality of Service (QoS) requirements and ranking of services based on previous user experiences and performance of services. It is a decision making tool, designed to provide assessment of Cloud services in terms of KPIs and user requirements. Customers provide their application requirements (essential and non-essential) to the framework which gives a list of Cloud services where the customer can deploy his/her application. Figure 1 shows the key elements of the framework:

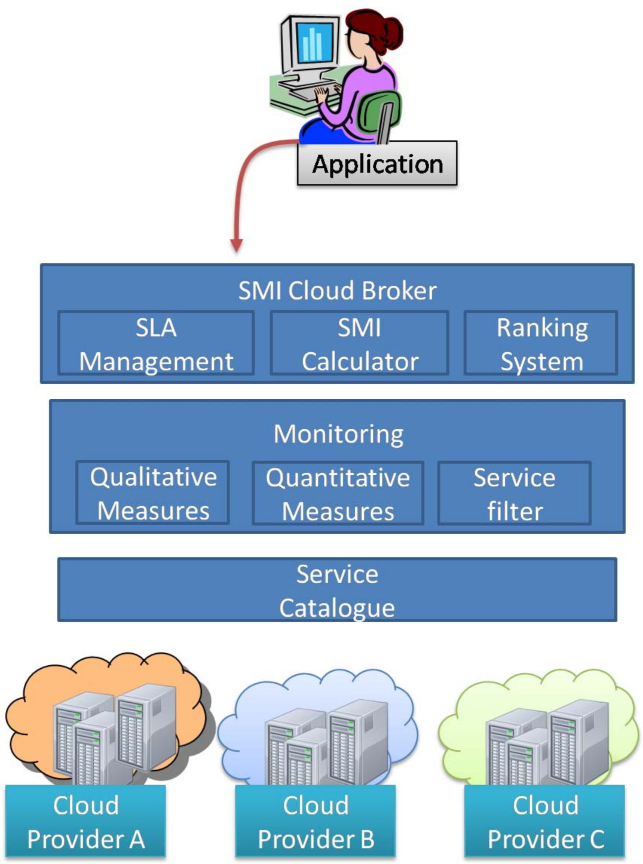

Figure 1. SMICloud Framework.

1) SMICloud Broker: It receives the customer's request for deployment of an application. It collects all their requirements and performs the discovery and ranking of suitable services using other components such as SMICalculator and Ranking systems. SLA Management is the component that keeps track of SLAs of customers with Cloud providers and their fulfilment history. The Ranking System ranks the services selected by the Cloud Broker which are appropriate for user needs. The SMI Calculator calculates the various KPIs which are used by ranking system for prioritizing the Cloud services.
2) Monitoring: This component first discovers the Cloud services which can satisfy user's essential Quality of Service requirements. Then, it monitors the performance of the Cloud services such as speed of VM, memory, scaling latency, storage performance, network latency and available bandwidth. It also keeps track of how SLA requirements of previous customers are being satisfied by the Cloud provider. For this layer, many tools are available some of which we discuss in the related work section.
3) Service Catalogue: stores the services and their features advertised by various Cloud providers.

The two important issues in building the framework as mentioned before are the measurement of various SMI KPI's and the ranking of Cloud services based on these measurements. In the next section, we present QoS model for IaaS providers based on SMI KPIs. This model can be easily extended for SaaS and PaaS.

IV. Quality Model For IaaS Provider

SMI KPI's are of two types: qualitative and quantitative. Qualitative are those KPIs which cannot be quantified and are mostly inferred based on user experiences. Quantitative are those which can be measured using software and hardware monitoring tools. For example, 'providers' ethicality' attribute is qualitative in nature. Since these KPIs since represent generic Cloud services, only some of them are important for particular applications and Cloud services. For example, the installability attribute in usability is more relevant to IaaS providers rather than SaaS providers since in SaaS there is almost no installation on the customer end. In addition, the same KPI can have different definitions based on the service. Some of these parameters depend on customer applications and some are independent. For example, suitability is more customer focused while flexibility is more provider focused. Therefore, it is complex to define precisely the SMI values for a provider particularly when there are many parameters involved and parameter definition also depends on many sub attributes. Here we define the most important quantifiable KPIs particularly in the context of IaaS Clouds. However, most of these proposed metrics are valid for other types of services. The modeling of qualitative attributes is beyond the scope of this paper.

A. *Service Response Time*

The efficiency of a service can be measured in terms of the response time, i.e. how fast the service can be made available for usage. The service response time depends on various sub-factors such as average response time, maximum response time promised by service provider, and percentage of time this response time level is missed.

- Average Response Time is given by $\sum_i T_i/n$ where T_i is time between when user i requested for an IaaS service and when it is actually available and n is the total number of IaaS service requests.

- Maximum Response Time is the maximum promised response time by the Cloud provider for the service.
- Response Time Failure is given by the percentage of occasions when the response time was higher than the promised maximum response time. Therefore, it is given by $n'/n * 100$, where n' is the number of occasions when service provider was not able to fulfil their promise.

B. Sustainability

Sustainability can be defined in terms of either the life cycle of the service itself or environmental impact of the Cloud service used. Therefore, we subdivide it into two attributes: service sustainability and environmental sustainability.

- Service sustainability is defined as how many components of a service can be reused without change with evolution of user requirements. In other words, we can say that the service that is more sustainable will have many more features than required. Therefore, service sustainability is given by: $\frac{\text{number of features provided by service}}{\text{number of features required by the customer}}$
- Environmental Sustainability can be measured as the average carbon footprint of the service. The metric of carbon footprint is complex and depends on many factors. Therefore, SMICloud can get the values using Carbon calculators such as PUE Calculator [4].

C. Suitability

Suitability is defined as the degree to which a customer's requirements are met by a Cloud provider. Now, there are two sub cases before we can define suitability. First, if after filtering the Cloud providers, there are more than one Cloud provider which satisfy all the essential and non-essential requirements of customer, then all are suitable. Otherwise, if filtering results in an empty Cloud provider list, then those providers which satisfy essential features are chosen. In this case, suitability will be the degree the service features come closer to user requirements. The resultant metric is:

$$\begin{aligned} Suitability &= \frac{\text{number of non-essential features provided by service}}{\text{number of non-essential features required by the customer}} \\ &\quad \textit{if only essential requirements are satisfied} \\ &= 1 \ \textit{if all features are satisfied} \\ &= 0 \ otherwise \end{aligned}$$

D. Accuracy

The accuracy of the service functionality measures the degree of closeness to user expected actual value or result generated by using the service. For computational resources such as Virtual Machines, accuracy's first indicator is the number of times the Cloud provider deviated from a promised SLA. It is defined as the frequency of failure in fulfilling promised SLA in terms of Compute unit, network, and storage. If f_i is the number of times the Cloud provider fails to satisfy promised values for user i over the service time T, then accuracy frequency is defined as $\sum_i \frac{f_i}{n}$ where n is the number of previous users. The another indicator of accuracy is the accuracy value which is defined by $\sum_i \frac{\frac{(\alpha_t - \alpha_i)}{\alpha_i}}{T_i}$, where α can be computational, network or storage unit of the service and T_i is service time T for user i.

E. Transparency

Transparency is an important feature of Cloud services due to the fast evolution of these services. It can be inferred as a time for which the performance of the user's application is affected during a change in the service. It can also be calculated in terms of frequency of such effect. Therefore, it can be measured by $\sum \frac{\sum \frac{time\_for\_service\_affect_i}{\text{number of such occurances}}}{n}$ where n is the number of customers using the service and i indicates the customer.

F. Interoperability

Interoperability is the ability of a service to interact with other services offered either by the same provider or other providers. It is more qualitative and can be defined by user experience. But since it is an important parameter for Cloud customers, we still defined as $\frac{\text{number of platforms offered by the provider}}{\text{number of platforms required by users for interoperability}}$.

G. Availability

The availability is percentage of the time a customer can access the service. It is given by: $\frac{\text{total service time-total time for which service was not available}}{\text{total service time}}$.

H. Reliability

Reliability reflects how a service operates without failure during a given time and condition. Therefore, it is defined based on the mean time to failure promised by the Cloud provider and previous failures experienced by the users. If $numfailure$ is the number of users who experienced failure in the amount of time less than promised by the Cloud provider and n is number of users. Let p_mttf be the promised mean time to failure. It is measured by:

$$\begin{aligned} Reliability &= \textit{probability of violation} \times p\_mttf \\ &= (1 - \frac{numfailure}{n}) * p\_mttf \end{aligned}$$

Reliability of storage can be defined in terms of durability that is chance of failure of a storage device

I. Stability

Stability is defined as the variability in the performance of a service. For storage, it is the variance in the average read and write time. For computational resources, it is the deviation from the performance specified in SLA i.e., $\sum \frac{\frac{\alpha_{avg,i} - \alpha_{sla,i}}{T}}{n}$ where α can be computational unit, network unit or storage unit of the resource; $\alpha_{avg,i}$ is the observed average performance of the user i who leased the Cloud service, $\alpha_{sla,i}$ is the promised values in the SLA; T is the service time; and n is the total number of users .

J. Cost

Cost depends on two attributes: acquisition and on-going. It is not easy to compare different prices of services as they offer different features and thus have many dimensions. Even the same provider offers different VMs which may satisfy user's requirements. For instance, Amazon Cloud offers small VMs in low cost than of Rackspace but the amount of data storage, bandwidth, compute unit are quite different between two providers [3][5]. To tackle this challenge, we defined a volume based metric i.e. cost of one unit of CPU unit, storage, RAM, and network. Therefore, if a VM is priced at p for cpu cpu unit, net network, $data$ data, RAM for RAM, then the cost of VM is $\frac{p}{cpu^a * net^b * data^c * RAM^d}$ where a, b, c, and d are weights for each resource attribute and $a + b + c + d = 1$. The weight of each attribute can vary from application to application. For example, for some applications RAM is more important than CPU unit, therefore for them $d > a$. So, we can use different weights of each attribute based on user application. Now, generally users need to transfer data which also incurs cost. Therefore, the total on-going cost can be calculated as the sum of data communication, storage and compute machine for that particular Cloud provider and service.

K. Adaptability

Adaptability is the ability of the service provider to adjust changes in the services based on customer's request. It is defined as the time taken to adapt to changes or upgrading the service to next level. For example, from small Amazon VM to medium size Amazon VM [3].

L. Elasticity

Elasticity is defined in terms of how much a Cloud service can be scaled during peak times. This is defined by two attributes: mean time taken to expand or contract the service capacity, and maximum capacity of service. The capacity is the maximum number of compute unit which can be provided at peak times.

M. Usability

The ease of using a Cloud service is defined by the attributes of Usability. The components such as operability, learnability, installability and understandability can be quantified as the average time experienced by the previous users of the Cloud service to operate, learn, install and understand, respectively.

V. Service Ranking using AHP

Ranking of Cloud services is one of the most important features of the SMICloud framework. The Ranking System computes the relative ranking values of various Cloud Services based on the quality of service requirements of the customer and features of Cloud services. As discussed before, Cloud services have many KPIs with many attributes and sub attributes which makes the ranking process a complex task. This problem in literature is defined as multiple criteria decision making (MCDM) [6]. The traditional weighted sum-based methods cannot be directly applied in such hierarchical structure of attributes. In addition, some of the attributes do not have any numerical value, for example, security.

Without a structured technique, the evaluation of the overall quality of different Cloud services would be very difficult given the number of attributes involved. In addition, the challenge is to compare each Cloud services based on each attribute, how to quantify them and how to aggregate them in a meaningful metric. To help in ranking such multi-attribute analysis techniques, we propose a ranking mechanism based on Analytic Hierarchy Process (AHP) [7] which is one the most widely used mechanism for solving problems related to MCMD. There are three phases in this process: problem decomposition, judgment of priorities, and aggregation of these priorities. AHP gives a very flexible way for solving such problem and can be adapted to any number of attributes with any number of sub-attributes. In the first phase, the ranking complex problem is modelled in a hierarchy structure that specifies the interrelation among three kinds of elements, including the overall goal, QoS attributes and their sub-attributes, and alternative services. In the second phase, firstly pairwise comparisons of QoS attributes are done to specify their relative priorities. Similarly, pairwise comparison of Cloud services is done based on each QoS attributes to compute their local ranks. In the final phase, for each alternative service, the relative local ranks of all criteria are aggregated to generate the global ranking values for all the services.

We describe the main steps to model the ranking problem in Cloud computing and then explain the overall calculation of ranks by a small case study example.

A. First Phase: Hierarchy structure for Cloud Services based on SMI KPIs

Figure 2 presents the Cloud service hierarchy based on SMI KPI's. The first layer is the goal of analysis which aims to find the relative service management index of all the Cloud services which satisfy the essential requirements of the user. The second layer contains hierarchy of QoS attributes both essential and non essential. The bottommost layer contains the values of all the Cloud services for all the lowest most QoS attributes in the hierarchy presented in the second layer.

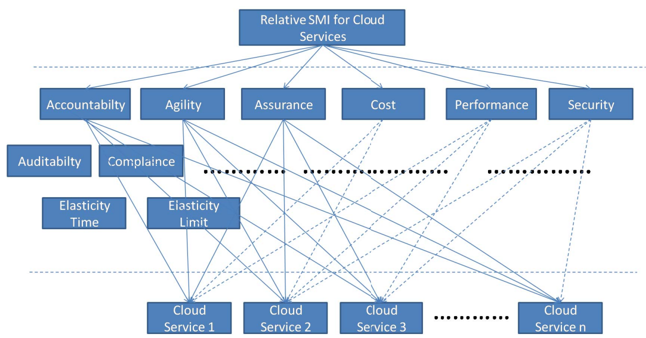

Figure 2. AHP Hierarchy for Cloud computing

B. Second phase: Computation of relative weights of each QoS and service

To compare two Cloud services, we need to assign weights to each attribute for taking into account their relative importance. To address this issue we consider two types of weights:

- User Assigned weight: The user of SMI Cloud can assign weights to each of the SMI attributes using values in some scale, for example [1..9] as suggested in the AHP method [7], to indicate the importance of one QoS attribute over the other. The table of relative importance is given in Table I. This methodology was proposed originally for calculating weights for each criteria in the AHP technique. This can be used to assign weights to all the QoS attributes. Customer expresses the preferences on each quality in each level.

Table I
RELATIVE IMPORTANCE VALUE

Equal importance/quality	1
Somewhat more important/better	3
Definitely more important/better	5
Much more important/better	7
Extremely more important/better	9

- Arbitrary user assigned weights: A user can assign weights in his own scale other than given by AHP technique. In this case, the sum of all weights may not be 1 which is a requirement of AHP. In this case, we normalize all the weight.

C. Phase3: Relative value-based weights for ranking Cloud services

These weights give relative performance of each Cloud services based on the values of the lowest level attributes. The process of assigning weights is not straightforward since the lowest level attributes can have various types of values. For example, the value of 'certifications' for a particular Cloud provider will be a list or a set. While the value of 'elasticity' will be a numeric value, values of some attributes may not be known. Therefore, the challenge is how to assign weights to each of the attributes when they are not quantifiable. To address this issue, we define the relative weights for ranking Cloud services based on a strategy proposed by Tran et al. [8]. In contrast to Tran et al. [8]'s work, the relative weight metrics designed in our work also consider two types of QoS requirements of Cloud users i.e., essential and non-essential.

Let w_q be the weight given by the user for SMI attribute q. Let v_i and v_j are the values of the attribute q for Cloud service i and j. Let s_i and s_j be the Cloud services and s_i/s_j indicates relative rank of s_i over s_j. Let v_r be the required value specified by the user. To compare both values for each Cloud service, firstly, we first need to make sure that the dimensional units of both the values are same. For example, if we want to compare the price of two VM instances, then the price should be \$ price of 1 CPU unit, 1 memory unit, and 1 hard disk unit. If it is data communication, the cost should have same dimension in terms of \$ price per 1 GB of data. Secondly, we have to compare the two values based on their types since the attributes values can vary from Boolean to a unordered set. For each type of attribute different comparison metric is proposed.

Thirdly, as discussed previously, users can specify essential and non-essential attributes. In the SMICloud framework, it is optional for users to specify their requirement values for non-essential attributes. Therefore, while comparing non-essential attribute for two services, there is a possibility that v_r is not specified by the user. Another possibility is that some of the attributes may not be possible to be monitored by SMICloud due to the non-availability of such APIs by the Cloud provider. Our proposed relative ranking model is flexible to tackle this issue and thus assigned ranking weights to Cloud services either based on weight w_q if any of the attributes values cannot be monitored or based on v_i and v_j if v_r is not specified by the user.

The proposed relative ranking model for each type of attribute is given by:

Boolean:

$$\begin{aligned} s_i/s_j &= 1 \ \textit{if} \ v_i \equiv v_j \\ &= w_q \ \textit{if} \ v_j = 1 \ \textit{and} \ v_i = 0 \\ &= 1/w_q \ \textit{if} \ v_j = 0 \ \textit{and} \ v_i = 1 \end{aligned}$$

Numeric: It can be of two types, higher is better or lower is better. If higher is better than it is $\frac{v_i}{v_j}$ is the value of s_i/s_j. If lower value is better, $\frac{v_j}{v_i}$ is the value of s_i/s_j. If q is a non essential requirement, then it may be possible that one service may not have a value. In that case, s_i/s_j is equal to w_q if v_j is given otherwise it will be $1/w_q$.

Unordered Set: This can occur in attributes such as portability which may be defined by the number of the platforms supported. To assign weights to the Cloud services for such time of QoS attribute values, the size of unordered set is considered. Let $size(v_i)$ and $size(v_j)$ be the number of elements in the set value for service i and j. Let $size(v_r)$ be the size of set requested by the user for QoS attribute q. If q is essential QoS attribute, then the Cloud service with the largest number of elements will be considered better and therefore, higher weight is assigned to it. The weights for such QoS attribute type values are calculated in the following way:

- if q is essential:

$$s_i/s_j = \frac{size(v_i)}{size(v_j)} \quad (1)$$

- if q is non-essential and if v_r is specified:

$$s_i/s_j = \frac{size(v_i \cap v_r)}{size(v_j \cap v_r)} \text{ if } v_j \cap v_r \neq \phi \wedge v_i \cap v_r \neq \phi \quad (2)$$

$$= 1 \text{ if } v_j \cap v_r \neq \phi \wedge v_i \cap v_r \equiv \phi \quad (3)$$

$$= w_q \text{ if } v_j \cap v_r \neq \phi \vee v_i \cap v_r \equiv \phi \quad (4)$$

$$= 1/w_q \text{ if } v_j \cap v_r \equiv \phi \vee v_i \cap v_r \neq \phi \quad (5)$$

Range type: Many QoS attributes of Cloud services are given as a range of values. For example, the initiation time of a Virtual Machine can be presented as a range. In that case, if v_r is the value range required by the user, then weights assigned to the services are:

- if q is essential:

$$s_i/s_j = \frac{len(v_i \cap v_r)}{len(v_j \cap v_r)} \quad (6)$$

- if q is non-essential and if v_r is specified:

$$s_i/s_j = \frac{len(v_i \cap v_r)}{len(v_j \cap v_r)} \text{ if } v_j \cap v_r \neq \phi \wedge v_i \cap v_r \neq \phi \quad (7)$$

$$= 1 \text{ if } v_j \cap v_r \equiv \phi \wedge v_i \cap v_r \equiv \phi \quad (8)$$

$$= w_q \text{ if } v_j \cap v_r \neq \phi \wedge v_i \cap v_r \equiv \phi \quad (9)$$

$$= \frac{1}{w_q} \text{ if } v_j \cap v_r \equiv \phi \wedge v_i \cap v_r \neq \phi \quad (10)$$

Using the above comparison metrics for each Cloud service, we obtain a one to one comparison of each Cloud service for a particular attribute. This will result in a one to one relative ranking matrix of size $N \times N$ if there are total N services. The relative ranking of all the Cloud services for the particular attribute is given by Eigen vector of the matrix. This Eigen vector matrix is also called Relative Service Ranking Vector (RSRV).

D. Phase 4: Aggregation of relative ranking for each SMI attribute

In final phase, the relative ranking vectors of each attribute are aggregated with their relative weights assigned in Phase 2. This aggregation process is repeated up for all the attributes in the SMI hierarchy which results in the final ranking of all the Cloud Services.

The whole process is explained in next section through a case study example.

VI. Case Study: Ranking Compute Cloud Services based on User QoS requirements

In this section, we present a case study example of the ranking mechanism presented in the previous section. The data used for the computation of service index for three Cloud services from three Cloud Providers. The QoS data is collected from various evaluation studies for three IaaS Cloud providers: Amazon EC2, Windows Azure, and Rackspace [9][10][11]. The unavailable data such as security level is randomly assigned to each Cloud service. User weights are also randomly assigned to each QoS service attribute. The top level QoS groups are Accountability, Agility, Assurance, Cost, Performance and Security.

In the following, we show step by step the ranking computation process for Cloud services. The relative weighting method is used to calculate the relative ranking of Cloud services for each QoS attribute. For each attribute, a relative ranking matrix is constructed using the following method. Based on the data given in Figure 3, the Relative Service Ranking Matrix (RSRM) for security will be:

$$RSRM_{security} = \begin{array}{cccc} & S1 & S2 & S3 \\ S1 & 1 & 4/8 & 4/4 \\ S2 & 8/4 & 1 & 8/4 \\ S3 & 4/4 & 4/8 & 1 \end{array}$$

Computing the Relative Service Ranking Vector (RSRV) for security from the matrix $RSRM_{security}$, we have

$RSRV_{security}$=[0.25 0.5 0.25]

Similarly, we have the relative service ranking vector of the Accountability: $RSRV_{Accountability}$=[0.25 0.5 0.25]

For Agility, there are two QoS attributes which are further subdivided into sub-attributes. Elasticity of a Cloud service is inferred from the time it takes to scale up. Its RSRV is given by: $RSRV_{Elasticity}$=[0.3470 0.1991 0.4538]

For each sub-attributes i.e., CPU, Memory and Disk, RSRVs are given by:

$RSRV_{CPU}$=[0.3076 0.4102 0.2820]

$RSRV_{Memory}$=[0.3409 0.3181 0.3409]

$RSRV_{Disk}$=[0.3623 0.4373 0.2002]

Combining RSRV vectors of sub-attributes, i.e. CPU, Memory and Disk, we get RSRM for 'Capacity':

$$RSRM_{capacity} = \begin{pmatrix} 0.30769 & 0.34090 & 0.36234 \\ 0.41025 & 0.31818 & 0.43738 \\ 0.28205 & 0.34090 & 0.20026 \end{pmatrix}$$

Top level QoS Groups (Weights)	First level Attributes (Weights)	Second Level Attributes (Weights)		Service 1 (S1)	Service 2 (S2)	Service 3 (S3)	Value Type	User Required Value
Accountability (.05)	level:0-10 (1)			4	8	4	Numeric	4
Agility (0.1)	Capacity (0.6)	CPU (0.5)	0.5	9.6	12.8	8.8	Numeric	4x1.6 GHZ
		Memory (0.3)	0.3	15	14	15	Numeric	10 GB
		Disk (0.2)	0.2	1690	2,040	630	Numeric	500 GB
	Elasticity (.4)	Time (1)	0.4	80-120	520-780	20-200	Range	60-120 sec
Assurance (0.2)	Availability (0.7)	0.7		99.95%	99.99%	100%	Numeric	99.9%
	Service Stability (0.2)	Upload Time (0.3)	0.3	13.6	15	21	Numeric	
		CPU (0.4)	0.4	17.9	16	23	Numeric	
		Memory (0.3)	0.3	7	12	5	Numeric	
	Serviceabilty (0.1)	Free Support (0.7)	0.7	0	1	1	Boolean	
		Type of Support (0.3)	0.3	24/7,Diagnostic Tools, Phone, Urgent Response	24/7,Diagnostic Tools, Phone, Urgent Response	24/7, Phone, Urgent Response	Unordered set	24/7, phone
Cost (0.3)	On-Going Cost (1)	VM Cost (0.6)	0.6	0.68	$0.96	0.96	Numeric	< 1 dollar/hour
		Data (0.2)	inbound	10	10	8	Numeric	100 GB/month
			outbound	11	15	18		200 GB/month
		Storage (0.2)	0.2	12	15	15	Numeric	1000 GB
Performance (0.3)	Service Response Time (1)	Range (0.5)	0.5	80-120	520-780	20-200	Range	60-120 sec
		Average Value (0.5)	0.5	100	600	30	Numeric	
Security (0.05)	level: 0-10 (1)			4	8	4	Numeric	4

Figure 3. Case Study Example

Next, we compute the relative service ranking vector for the 'Capacity'.

$$RSRV_{capacity} = \begin{pmatrix} 0.30769 & 0.34090 & 0.36234 \\ 0.41025 & 0.31818 & 0.43738 \\ 0.28205 & 0.34090 & 0.20026 \end{pmatrix} \begin{pmatrix} 0.5 \\ 0.3 \\ 0.2 \end{pmatrix}$$

Therefore,

$$RSRV_{capacity} = \begin{pmatrix} 0.3286 & 0.3881 & 0.2834 \end{pmatrix}$$

Similarly, the relative service ranking vector for Agility is given by:

$$RSRV_{agility} = \begin{pmatrix} 0.3286 & 0.34701 \\ 0.3881 & 0.19914 \\ 0.2834 & 0.45384 \end{pmatrix} \begin{pmatrix} 0.6 \\ 0.4 \end{pmatrix}$$

$$RSRV_{agility} = \begin{pmatrix} 0.336 & 0.3125 & 0.3516 \end{pmatrix}$$

In similar way we can compute the relative service ranking vector of all other top level QoS attributes i.e, Assurance, Cost and Performance.

$$RSRV_{assurance} = \begin{pmatrix} 0.3812 & 0.2671 & 0.3517 \end{pmatrix}$$

$$RSRV_{Cost} = \begin{pmatrix} 0.4073 & 0.3338 & 0.2589 \end{pmatrix}$$

$$RSRV_{performance} = \begin{pmatrix} 0.2846 & 0.1181 & 0.5973 \end{pmatrix}$$

Finally, we aggregate all the RSRVs of all the attributes to get the relative service ranking matrix for three providers:

$$RSRM = \begin{pmatrix} 0.25 & 0.336 & 0.3812 & 0.1619 & 0.2846 & 0.25 \\ 0.5 & 0.3125 & 0.2671 & 0.1308 & 0.1181 & 0.5 \\ 0.25 & 0.3516 & 0.3517 & 0.7073 & 0.5973 & 0.25 \end{pmatrix}$$

To get the final relative service ranking vector, we multiply above RSRM with the weights of the top level QoS attributes.

$$RSRV = \begin{pmatrix} 0.25 & 0.336 & 0.3812 & 0.4073 & 0.2846 & 0.25 \\ 0.5 & 0.3125 & 0.2671 & 0.3338 & 0.1181 & 0.5 \\ 0.25 & 0.3516 & 0.3517 & 0.2589 & 0.5973 & 0.25 \end{pmatrix} \begin{pmatrix} 0.05 \\ 0.1 \\ 0.2 \\ 0.3 \\ 0.3 \\ 0.05 \end{pmatrix}$$

Therefore, the relative ranking of all the Cloud service can be decided based on resultant RSRV (0.3424, 0.2702, 0.3874). The Cloud services are ranked as $S3 \succ S1 \succ S2$.

VII. Related Works

In this section, we compare and contrast our work with previous research works for evaluating and comparing the performance of different Cloud services. With the increasing popularity of Cloud computing, many researchers studied the performance of Clouds for different type of applications such as scientific, e-commerce and web applications. For instance, Iosup et al. [12] analyzed the performance of many-task applications on Clouds. Similarly, many performance monitoring and analysis tools are also proposed in the literature [12]. Our work complements these previous works by utilizing these tools and data to rank and measure the QoS of various Cloud services according to users' applications. Other works such as CloudCmp [9] proposed frameworks to compare the performance of different Cloud services such as Amazon EC2, Windows Azure and Rackspace. These works again focussed on comparing low level performance of Cloud services such as CPU and network throughput. In our work, we use these performance data to measure various Quality of Service attributes and evaluate relative ranking of Cloud services.

Even though the evaluation and comparative ranking of various Cloud services is quite new in Cloud computing area, it is very old concept in other areas such as Web services. The most related work in this area is done by Tran et al. [8]. This work also proposed a similar AHP based ranking technique. However, the algorithm was designed for web services and thus did not consider various performance parameters such as VM's capacity which are specific to Cloud computing. In addition, we also define key performance and cost metrics based on SMI [2] framework for Cloud computing services.

In summary, according to authors' best knowledge, our work is the first to define all key performance metrics for QoS attributes in SMI framework and apply AHP-based ranking in Cloud computing.

VIII. Conclusions and Future Work

Cloud computing has become an important paradigm for outsourcing various IT needs of the organization. Currently, there are many Cloud providers who are offering different Cloud services with different QoS and SLAs. One of the challenging questions faced by Cloud customers is how to find out the best Cloud services which can satisfy their QoS requirements in terms of parameters such as performance and Security. Therefore, Cloud Service Measurement Index Consortium (CSMIC) proposed a framework based on common characteristics of Cloud services. The aim of this consortium is to define each of QoS attributes given in the framework and provide a methodology for computing a relative index for comparing different Cloud services.

In this context, this work presents the first framework, SMICloud, to systematically measure all the QoS attributes proposed by CSMIC and rank the Cloud services based on these attributes. We address some key challenges by designing metrics for each quantifiable QoS attributes for measuring precisely the service level of each Cloud provider. We proposed an AHP based ranking mechanism which can evaluate the Cloud services based on different applications depending on QoS requirements. Our proposed mechanism also address the challenge of different dimensional units of various QoS attributes by providing a uniform way to evaluate relative ranking of Cloud services for each type of QoS attribute.

We believe SMICloud framework represents a significant next step towards enabling accurate QoS measurement and Cloud service selection for Cloud customers. By using techniques given in this work, Cloud providers can identify how they perform compared to their competitors and therefore, they can improve their services.

It is the first version, further enhancements are planned. In future, we will extend our ranking algorithm to cope with variation in QoS attributes such as performance by adopting fuzzy sets. We also extend the quality model to non-quantifiable QoS attributes and validate the complete quality model based on assessment such as IEEE 1061.

References

[1] Buyya et al., "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] C. S. M. I. C. (CSMIC), "SMI Framework," *URL http://beta-www.cloudcommons.com/servicemeasurementindex*.

[3] J. Varia, *Cloud Computing: Principles and Paradigms*. Wiley Press, 2011, ch. 18: Best Practices in Architecting Cloud Applications in the AWS Cloud, pp. 459–490.

[4] H. Pan, *Green Data Centers Monthly Newsletter February 2010*. Information Gatekeepers Inc.

[5] Rackspace, "Cloud Servers," *URL http://www.rackspace.com*.

[6] M. Zeleny, *Multiple Criteria Decision Making*. McGraw-Hill New York, 1982, vol. 25.

[7] T. Saaty, *Theory and Applications of Analytic Network Process*. RWS Publications Pittsburgh, PA, 2005, vol. 4922.

[8] V. Tran, H. Tsuji, and R. Masuda, "A new qos ontology and its qos-based ranking algorithm for web services," *Simulation Modelling Practice and Theory*, vol. 17, no. 8, pp. 1378–1398, 2009.

[9] A. Li, X. Yang, S. Kandula, and M. Zhang, "Cloudcmp: comparing public cloud providers," in *Proceedings of the 10th Annual Conference on Internet Measurement, Melbourne, Australia*, 2010.

[10] J. Schad, J. Dittrich, and J. Quiane-Ruiz, "Runtime measurements in the cloud: observing, analyzing, and reducing variance," *Proceedings of the VLDB Endowment*, vol. 3, no. 1-2, pp. 460–471, 2010.

[11] A. Iosup, N. Yigitbasi, and D. Epema, "On the performance variability of production cloud services," *Proceedings of IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing, CA, USA*, 2011.

[12] A. Iosup, S. Ostermann, N. Yigitbasi, R. Prodan, T. Fahringer, and D. Epema, "Performance analysis of cloud computing services for many-tasks scientific computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 6, pp. 931–945, 2011.