A FRAMEWORK FOR SELECTION OF BEST CLOUD SERVICE PROVIDER USING RANKED VOTING METHOD

Gaurav Baranwal
School of Computer and Systems Sciences
Jawaharlal Nehru University
New Delhi, India
gaurav11_scs@jnu.ac.in

Deo Prakash Vidyarthi
School of Computer and Systems Sciences
Jawaharlal Nehru University
New Delhi, India
dpv@mail.jnu.ac.in

Abstract— **Cloud computing provides computing resources on demand. It is a promising solution for utility computing. Increasing number of cloud service providers having similar functionality poses a problem to cloud users of its selection. To assist the users, for selection of a best service provider as per user's requirement, it is necessary to create a solution. User may provide its QoS expectation and service providers may also express the offers. Experience of existing users may also be beneficial in selection of best cloud service provider. This paper identifies QoS metrics and defines it in such a way that user and provider both can express their expectation and offers respectively into quantified form. A dynamic and flexible framework using Ranked Voting Method is proposed which takes requirement of user as an input and provides a best provider as output.**

Keywords—Cloud Computing; Quality of Service; Ranked Voting; Service Measure.

I. INTRODUCTION

Cloud computing, emerged as a new paradigm for utility computing, is growing very rapidly and gaining attention by not only large organizations but academic organizations, government organizations, small and medium organizations also. Like utility, It offers resources on demand. It offers broadly three types of services: Software as a service (SaaS), Platform as a service (PaaS) and Infrastructure as a service (IaaS) and three deployment models: Public Cloud, Private Cloud and Hybrid Cloud.

Big organizations and users can reap the benefits of Cloud computing. Cloud customers have to pay only operational cost unlike traditional data centre which reduces computing cost significantly. If company is growing, customer can rent more computing resources from Cloud without bothering to pay for unneeded resources. Since Cloud customers need not to bother about the infrastructure maintenance and consume resources on pay per use basis, organization becomes more agile. Also, Cloud computing can utilize resources more efficiently i.e. same infrastructure may be used by many customers resulting in less number of required servers. It makes Cloud greener and Cloud customers more eco-friendly. Other Cloud benefits include multi-tenancy, flexibility, disaster recovery etc. Though, some key issues like security, interoperability, standardization, SLA etc. require a deeper addressing in order for Cloud to be fully functional.

Many public Cloud service providers offer same service on low cost with better performance than others. Their customer support also may vary. Some providers charge higher for CPU but lower for RAM. From security point of view, they may have different certifications. Increasing number of Cloud service providers is making Cloud market more competitive day by day. Each service provider claims their best. This, in turn, makes the Cloud customers difficult to select a provider which fulfils their QoS requirement. May be an application was implemented with different needs but the needs might change over time like language, operating system etc. For such application, service provider with multilingual support will be a better option. It may be the case that an application is simple in its starting phase but after some time as company grows and scales, its application may become more complex. So Cloud customers have to find a Cloud service which not only satisfies its current need but will also adapt with future requirement.

For selection of a best Cloud provider, a customer must identify its QoS measures that is used to compare various service providers. But QoS measurement may be a difficult task because of lack of standard to understand it. For creating a standard to measure QoS in Cloud, Cloud Services Measurement Initiative Consortium (CSMIC) was formed in 2010 in Carnegie Mellon University. CSMIC is a group of globally established organizations. Professionals, from these organizations, have developed a standard measurement framework called SMI (Service Measurement Index). SMI includes seven major characteristics, each characteristic with 3 or more attributes. SMI clearly defines each attribute which helps decision makers to measure QoS requirement of customers, compare this to offerings of different Cloud service providers and to choose a best Cloud service provider.

In this work, a framework has been designed which considers not only metrics defined by SMI but also others metric found in analysis of QoS parameters of Cloud computing in research. Several challenges arise in designing this framework. First being how to quantify all identified

978-1-4799-2572-8/14/$31.00 ©2014 IEEE

metrics for a Cloud service. This work categorizes metrics in two categories: application dependent metrics and user dependent metrics as some metrics may be application dependent. It gives a clear idea to quantify each metrics separately. To rank the Cloud providers, this work uses ranked voting method of voting system to rank providers. Each identified metric will act as a voter and compare its required value with providers providing value of that metrics to rank providers accordingly. Values of metrics can be of different types like numeric, Boolean etc. This work also gives rules for metric comparison. After obtaining ranked voting data containing voter list (i.e. identified metrics and rank of Cloud services for metrics), ranked voting method proposed by Tsuneshi Obata and Hiroaki Ishii [1] is applied to find a best Cloud service provider.

This paper is organized as follows. Section II mentions that why SMI is used. Section III provides a global view of cloud computing. Section IV details quality of service metrics. Section V defines comparison rules for QoS metrics. Section VI details the method to rank service providers. Section VII concludes the work.

II. WHY SMI (SERVICE MEASURE INDEX)

In ranked voting method, voters select candidates and rank them in order of their preferences. Voters will rank service providers as candidates. SMI framework, developed by the Cloud Services Measurement Initiative Consortium (CSMIC) [2], is considered to create voter list. It provides SMI (service measure index), which is a set of business relevant key performance indicators (KPI's). It is globally accepted measure which covers all aspects of Cloud computing and helps organizations to measure and compare Cloud based IT services in a standard way based on their own specific business and technology requirements. In hierarchy, at the top level it categorizes measurement space into 7 categories: Accountability, Agility, Assurance, Financial, Performance, Security and Privacy and Usability. Each category is further categorized in attributes. Each attribute in SMI has been clearly defined which helps to understand the metric and collecting data to measure that metric. Hierarchy of SMI framework is given in figure 1. SMI is a long list of metrics. It consists of nearly 50 metrics. Some metrics are application dependent like cost, type of security and response time. If a provider wants to deploy more than one application in Cloud, value of application dependent attribute may have different values for different application. From user's point of view, user creates a list of metrics for its requirement; this list will contain all application dependent metrics multiplied by number of application deployed by user and also contain all remaining metrics i.e. user dependent metrics. This work includes some more metrics in this list not considered in SMI, like location of data centres, reputation of provider, client interface etc. This long list is very helpful to create a list of voters in this work. Each metric in this list will be considered as a voter and it will rank providers after comparing its required value and value provided by providers. Although all attributes in list are important, this work provides flexibility to user to select some attributes as voter according to its need.

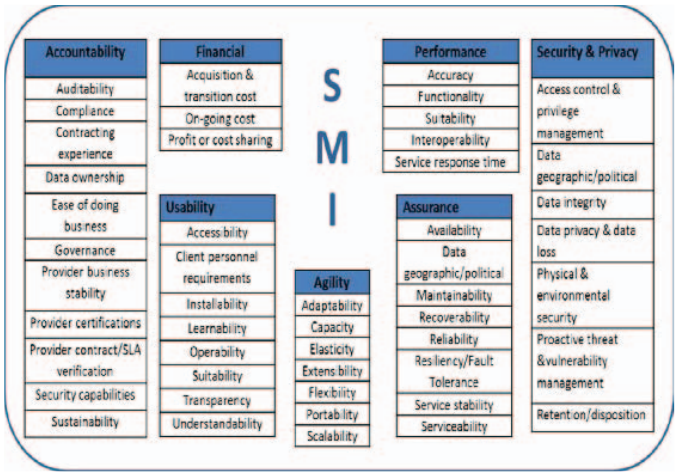

Figure 1: Categories and Attributes of Service Measure Index (SMI)

III. GLOBAL VIEW OF CLOUD COMPUTING

At present, there are huge numbers of service providers in Cloud market. They advertise about their services in their own manner. User experience also plays an important role in choosing a best provider. Architecture is presented here, that brings together all Cloud providers which makes implementation of proposed framework easy. A global view of Cloud computing as a Federated Cloud or Inter Cloud [3] has already been proposed. This work considers same global view with some modifications. Figure 2 depicts such Cloud architecture.

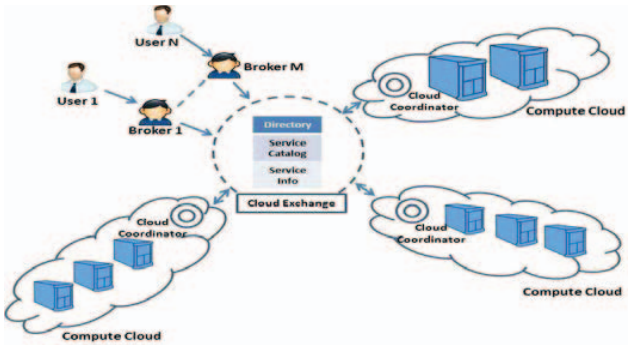

Figure 2: Global View of Cloud Computing

The components of Cloud architecture are as follows.

A. *Cloud Exchange*

Cloud exchange acts as central coordinator which brings together Cloud providers, Cloud coordinator, Cloud user and Cloud broker. It contains directory to provide information to broker about Cloud services.

B. *Directory*

Directory contains all information about service providers which are required in selection of a best provider. It contains two type of information: Service catalogue and Service info.

Service Catalogue

Using service catalogue, all service providers give details about their service on the basis of defined quality metric.

Service Info

Service info is a repository which contains Service log and Review log. Service log is a database which stores log records of all registered service providers. Log records keep information about history of service providers. Here, history means response time, number of successful or unsuccessful transactions, availability etc. Review log contains information about experience of user for a particular service like stability, transparency, reputation, feedback, recommendation etc.

C. Cloud Broker

Cloud broker takes details of requirement of user and details of service provider from directory and analyse them using proposed framework in order to choose a best provider. It is assumed that user has already informed the Cloud broker about its essential QoS. If any essential QoS is not offered by a service provider, broker does not consider that provider for comparison. For example, a user wants to implement an application on JAVA platform. Here JAVA is an essential requirement; broker will not consider those providers who do not support JAVA platform. This framework supports only those providers that satisfy all essential QoS. Since Cloud broker directly interact with users, it collects information from users about their experience for used service providers, provide service info and update service info when required.

D. Cloud Coordinator

Cloud coordinator acts as a representative for Cloud service provider. It is responsible to provide service catalogue and service log and periodically update both.

IV. QOS METRICS

As mentioned, SMI has been developed to compare all type of service providers: IaaS, PaaS and SaaS providers. For example, cost is a metric relevant to all type of service providers, while elasticity is more relevant to IaaS in comparison to PaaS and SaaS. Though, definition and measurement of cost metric will be different for Iaas, Paas and Saas i.e. same metric can have different dimension based on service. This work focuses only on Iaas. So, definition and measurement of all metric should be according to Iaas service providers. Also to notice that some metrics are application dependent like Virtualization technique, Platforms supported etc., while some metrics (stability, suitability etc.) can be measured by experience of existing users only and some metrics are sub-attribute dependent like cost, certifications etc. Thus, it is a complex task to create a list of metrics that should cover all the discussed aspects. To tackle all discussed issues this work divides metrics in to two categories; application dependent and user dependent. It defines application dependent metrics on the basis of application's requirement and user dependent metrics on user's requirement basis. For simplicity, a User Experience metric is considered to measure all existing users based metrics like stability, transparency etc. In the next section, all proposed metrics have been defined and explained to show how these metrics will help to measure and compare service providers.

A. Application Dependent Metrics

Some of the application dependant metrics are as follows.

Reliability

Using virtualization technology, Cloud computing utilizes resources more efficiently. A physical server can deploy many virtual machines and operating systems. However, with the increase in software and hardware components, more failures are likely to occur in the system. Thus, one should understand failure behaviour in the Cloud environment in order to better utilize the Cloud resources. In this work, MTBF (mean time between failures) is considered for reliability measure. It is assumed that MTBF will be provided by the service provider for each Cloud service. MTBF applies to a service that is going to be repaired and returned to operate, and is defined as

$$MTBF = \frac{Total\ time}{Number\ of\ failures}$$

A practical definition of reliability is "the probability that a service operating under specified conditions shall perform satisfactorily for a given period of time". It is assumed that failures occur randomly so reliability for a certain time period *t* can be described by the given exponential distribution.

$$R(t) = e^{\frac{-t}{MTBF}}$$

Availability

Availability is the degree to which a system or component is operational and accessible when required for use [IEEE 90]. Different applications require different availability rating. Online tools are available which run benchmark on different providers and provides an idea about availability of different providers such as Global Provider View. CLOUDSLEUTH's application Global Provider View brought by Compuware is a near real-time visualization tool, which provides availability and response time of different service providers [4]. It uses Gomez Performance Network (GPN) to measure the performance of an identical sample application running on several popular Cloud service providers. GPN is a benchmark which has been created unambiguously by Gomez. User can collect data of availability and response time at different time frame and at different locations or worldwide.

Security

Cloud providers take different types of security measures. Different application may have different requirement for security measures. Four types of security measures discussed in [5] are considered for service measures; (i) Crypto algorithms and key management, (ii) Physical security support, (iii) Network security support, (iv) Data security support. Very few providers are providing all security measures.

Data centres

"Performance is highly affected by the speed-of-light latency, TCP latency, (both of which are directly correlated to circuit distance between user and files), as well as packet loss. By placing the files closer to the user, both speed-of-light latency and TCP latency are minimized. Packet loss is also minimized because the probability of packet loss increases as distance increases" [6]. So, number of data centres their inter-

distance and the area must be criteria to rank the service providers.

Cost

Cost is an important factor to rank providers. Basically cost is a function of requirement of resources like CPU, memory, storage and data transfer in an application. Generally two types of pricing plan are used; Price bundling and Unbundling [7]. In price bundling plan, price is charged for each characteristic separately e.g. $45 per CPU or $10 per 1GB RAM. In unbundling plan, providers offer predefined bundles e.g. 2.4 GHz Vcpu, 1GB RAM and 80GB HDD. These different pricing plans create problem to find a standard cost function. Growing market of Cloud has not only created competitive environment, but it has also encouraged designing online tools to compare cost metric of Cloud service providers such as CLOUDRODO [8]. A user can enter its requirement in these tools and get meaningful suggestions for different service providers.

Operating Systems Support

Providers support different OS like Mac OS X, Windows, and Open SUSE Linux etc. It is possible that one provider support some OS and other provider supports some other OS like Windows Azure supports only Windows operating system, while GoGrid supports Windows server 2003/2008, Red hat Linux 5.1/5.4 etc. Different applications require different OS support. Application can rank providers based on decision whether provider provides required platform or not.

Platforms Supported

Same as operating systems, different providers support different type of platforms. For example, CloudSigma supports Java, PHP, WinDev, Dot Net. While Firehost supports Ruby, Java, PHP, and Dot Net. Different application requires different platform support. Application may rank providers like it ranks providers based on Operating System.

Virtualization Technique

Today, any application can be deployed using any virtualization platform. This work considers that if one is going to deploy a new application, one can use any virtualization platform according to its requirements. Unlike platforms, generally a provider supports only one virtualization platform. But each virtualization technique has its own advantages. If any one does not require advance features it may seek Hyper-V. Xen is Linux based and its management and administration is difficult but it has its advantage for experienced Linux user. VMware provides robust features. There are many vendors with application to fulfil the need in Vsphere. Hyper-V is freeware but one has to buy windows server. So choice of hypervisor very much depends on application's requirement.

Customer support facility

Type of support, response time for support and the charge for customer support are important factors which define customer support facility. Generally new users prefer a provider with a good support system. Some providers offer free customer support service, but mostly the providers charge accordingly. GoGrid provides free 24/7 phone support and free 24/7 premium support. Amazon AWS provides premium Support (Urgent - 1 hour, High - 4 business hours, Normal - 1 business day, Low - 2 business days). So to measure customer support facility metric, the metric is assumed as unordered set which may contain elements like Free 24/7 phone support, Urgent support, Basic support, Low support, Diagnostic tools etc.

Response Time

Response time represents difference between time of request for service and time when service is available. Response time depends on infrastructure as well as application for which request has been generated. But when new users are going to use Cloud services, they can measure response time of particular Cloud service providers by some benchmark. Global Provider View tool, discussed in Availability metric, can also be used to measure response time of different service providers.

Throughput and Efficiency

Throughput means number of tasks completed per unit time by Cloud service provider. An application is constituted as collection of tasks (n). It may be an embarrassingly parallel application in which all tasks are independent and can be executed in any order. On the other side, all tasks may be dependent on each other. An application may belong to high-performance computing, high-throughput computing or many-task computing. So throughput of an application not only depends on service provider but also on application itself. Let T be the execution of tasks of an application in traditional data centre and T_o be the overhead of Cloud data centre. So, throughput (Thr_{app}) of service provider for an application is given by:

$$Thr_{app} = \frac{n}{T+T_o}$$

Efficiency (e_{app}) of a service provider is given by:

$$e_{app} = \frac{T}{T - T_o}$$

Capacity

Capacity means maximum amount of resources that a service provider can provide at peak times. Capacity of a service provider can be quantified by four metrics; CPU capacity metric (measured by multiplication of number of CPU and frequency of CPU), memory capacity metric (measured in GB), storage capacity metric (measured in GB) and network capacity metric (measured by bandwidth).

Let cpu_u, mem_u, sto_u, BW_u be required values of CPU, memory, storage and bandwidth respectively by user and cpu_{sp}, mem_{sp}, sto_{sp}, BW_{sp} be vectors of offered service providers value of CPU, memory, storage and bandwidth. Collectively, it can be written as follows.

$$cpu = \{cpu_u, cpu_{sp}\}$$
$$mem = \{mem_u, mem_{sp}\}$$
$$sto = \{sto_u, sto_{sp}\}$$
$$BW = \{BW_u, BW_{sp}\}$$

Normalized value of cpu, mem, sto, BW are

$$cpu_N = \frac{cpu}{\max\{cpu\}} = \{cpu_{N,u}, cpu_{N,sp}\}$$
$$mem_N = \frac{mem}{\max\{mem\}} = \{mem_{N,u}, mem_{N,sp}\}$$
$$sto_N = \frac{sto}{\max\{sto\}} = \{sto_{N,u}, sto_{N,sp}\}$$
$$BW_N = \frac{BW}{\max\{BW\}} = \{BW_{N,u}, BW_{N,sp}\}$$

To differentiate service providers clearly, if normalized offered value of a service provider is less than required normalized value by user, normalized offered value is made zero. It can be written as

$$cpu_{N,sp}(i) = \begin{cases} cpu_{N,sp}(i), & cpu_{N,sp}(i) \geq cpu_{N,u} \\ 0, & otherwise \end{cases}$$
$$mem_{N,sp}(i) = \begin{cases} mem_{N,sp}(i), & mem_{N,sp}(i) \geq mem_{N,u} \\ 0, & otherwise \end{cases}$$
$$sto_{N,sp}(i) = \begin{cases} sto_{N,sp}(i), & sto_{N,sp}(i) \geq sto_{N,u} \\ 0, & otherwise \end{cases}$$
$$BW_{N,sp}(i) = \begin{cases} BW_{N,sp}(i), & BW_{N,sp}(i) \geq BW_{N,u} \\ 0, & otherwise \end{cases}$$
$$F = [cpu_{N,sp}', mem_{N,sp}', sto_{N,sp}', BW_{N,sp}']$$

Capacity of service provider i is depicted as follows.

$$capacity_i = \sum_{j=1}^{4} F_{ij} * w_j$$

where w_1, w_2, w_3, w_4 are weights of capacity of CPU, memory, storage and network respectively for an application.

B. User Dependent Metrics

Some of the user dependant metrics are as follows.

Reputation

Reputation measures trustworthiness of a Cloud provider. It is based on the users' experience for service providers. Different users may have different opinions for the same provider. Reputation of a provider can be calculated taking average of rank assigned to a provider by different users. It is assumed users ranks the providers in a given range.

$$Reputation = \frac{\sum_{i=1}^{n} r_i}{n}$$

where r_i is rank of reputation assigned by user i to provider and n is the number of times a provider has been ranked.

Client Interface

Client interface is also an important criterion that a user may consider for the selection of a provider. We observed different providers and find that they support different types of client supports. Different client interfaces which a user can expect are Web Access, API, FTP Access, Website, Management Console etc. User can rank providers on the basis of client interface.

Monitoring

Monitoring is a subjective metric. Three levels to measure this metric is being considered; Poor, Average, and Extensive. Poor means provider is not providing any monitoring tool, monitoring is responsibility of user itself. Average means simple integrated monitoring tools are provided by provider. It may have few indicators or few alerts. In extensive, provider provides complete integrated monitoring tool with no cost and monitoring is not an issue for the user.

Free Trial

Some Providers provide free trial to test their services. It is very beneficial for users. User can test services before deployment. Definitely, provider with free trial service will get higher rank.

API

Whether provider is providing an API to interact with server or not is also an important parameter. It is also a subjective metric. Just like monitoring metric, again three levels to measure this metric is being considered; None, Average and Extensive.

Certification

Different providers have different certifications for industry regulatory compliance e.g. Amazon has certification of SAS 70 TYPE II, ISO 27001, and HAPP etc. Windows Azure has certifications of Safe Harbor policy, ISO 27001 etc. A user may prefer one certification over other. So, user can rank providers on the basis of the type of certification it expects from provider. User can give weights to certifications according to its need eventually to rank providers.

Sustainability

Sustainability has three dimensions; economic, social and environmental impact of Cloud service provider [9]. Environmental impact can be measured using metrics like carbon emission, water use and resource consumption. Economic impact can be measured using metrics such as power cost of servers, storage, and networking facility cost and support cost. Social impact can be measured using metrics like economic development and socio-political stability. Measurement of all three impacts is complex. Some common metrics are PUE (power usage effectiveness), CUE (carbon usage effectiveness), CUPS (compute units per second), CADE (corporate average data centre efficiency), DPPE (data centre performance per energy), and WUE (water use efficiency). Some providers use their own measurement and auditing tools.

Scalability

"Scalability is a measure of an application system's ability to provide increased throughput, reduced response time and/or support more users when hardware resources are added without modification in cost-effectively"[10]. There is two basic dimensions in scaling, vertical scaling (scale-up) and horizontal scaling (scale-out). In vertical scaling resources are added to a virtual machine to increase capacity. While in horizontal scaling, new virtual machines are initiated during peak loads. Generally, all Cloud providers support horizontal scaling unlike vertical scaling. So, if a provider is providing vertical scaling, it gets higher rank.

Elasticity

Elasticity can be understood by its two fundamental elements: Time and Cost [11]. Time means how much time a service provider takes to provision or de-provision resources and cost means whether service provider charges on per hour basis or per minute basis. For example, suppose a provider is charging on per hour basis and one of its consumer requests to de-provision an instance. It may be possible that de-provisioning request is accepted and consumer may continue to be charged for it for a while. So, to measure elasticity both elements are considered. Let w_1 and w_2 be weights of time and cost respectively i.e. importance of time and cost for a user with $w_1 + w_2 = 1$. This cost means charge per hour or charge per minute which is not same as cost metric. Elasticity can be written as

$$elasticity = w_1 * time + w_2 * cost$$

User Experience

A user, going to take services from Cloud, must consider the views of existing users for the services. Existing users of a service can describe accuracy, stability and transparency of the service in a better way. So, experience of a user with a service is an important factor. For the proposed framework, it is assumed that existing users rate Cloud services on level 0 to 10. Higher value of level indicates the better experience of users with that service.

V. Metric Comparison Rules

A metric can have different types of values like cost is a numerical value, scalability is a Boolean value, monitoring is a levelled value etc. This makes quantification of metrics, which helps voters to rank providers in order of preference, a challenge. To handle this, metric comparison rule is proposed (rank 1 is considered as highest rank and providers with same value get same rank).

Boolean

If a metric is Boolean type i.e. value is yes or no, provider with positive response will get rank one and with negative response will get rank two. For example, in case of free trial, provider with answer yes will get rank one and with answer no will get rank two.

Range Type

Suppose r_i is range provided by provider i and r_u is required range by a user. For each provider i, calculate $length(r_i \cap r_u)$, sort length in ascending order, arrange providers in order of sorted list, then rank them from high to low.

Numeric

Case 1: If metric's value of service providers must be as large as possible e.g. availability.

Suppose r_i is numeric value of metric of provider i and r_u is required value by user. Find all providers i such that $r_i \geq r_u$ and rank them 1. Sort remaining providers in descending order and rank them from high to low.

Case 2: If metric's value of service providers must be as low as possible e.g. cost.

Suppose r_i is numeric value of metric of provider i and r_u is required value by user. Find all providers i such that $r_i \leq r_u$ and rank them 1. Sort remaining providers in ascending order and rank from high to low. Subjective metrics are also considered as numeric. For example, three levels considered for measuring monitoring metric (Extensive, Average and Poor), can be ordered. So extensive is valued as 1, average is valued as 2 and poor is valued as 3. This method is used for all subjective metrics i.e. monitoring API.

Unordered Set

For unordered set type of metric, it is assumed that user gives not only its required set but also gives weight for each element in set i.e. importance of each element in set and sum of weight should be 1. Suppose s_i is set provided by provider i and s_u is required set by a user. For each provider i, calculate set difference i.e. $s_u - s_i$. All providers i for which $s_u - s_i = \emptyset$, rank them one, and for remaining providers for which $s_u - s_i \neq \emptyset$, calculate sum of weight of elements in $s_u - s_i$, sort these values in ascending order and rank them from high to low.

Data centre value

For data centre metric, number of data centre and distance between location of data centre and expected service location both are important. The procedure to rank providers for data centre metric is explained as follows. Suppose y_i is number of data centres provided by provider i and x is number of data centres required by an application. For each provider i for which $y_i \geq x$, find x data centres that are near to service locations required by an application. After finding x data centres as per the requirement of application find the distance between each service location and data centre and add them. A provider with minimum sum of distances will be ranked 1 and so on. Providers who does not satisfy constraints $y_i \geq x$, will be sorted by their number of data centres and ranked.

VI. Ranking of Cloud Service Providers

In this work, to find a best provider for a user, ranked voting method is used [1]. In ranked voting system, voter ranks alternatives in order of preference. As discussed, there is a long list of metrics to find an efficient Cloud provider. Each metrics will act as a voter, Cloud providers are candidates for them. Thus, a ranked voting data set is prepared. In research, some method has been proposed to analyse ranked voting data such as Data Envelopment Analysis (DEA) introduced by Cook and Kress [Management Science 36 (11) (1990) 1302]. But DEA often suggests more than one efficient candidate. Some methods are proposed to discriminate these efficient candidates. But order of preference may be changed because of existence of an inefficient candidate. Tsuneshi Obata and Hiroaki Ishii introduced [1] a novel method which does not use information of inefficient candidate to discriminate efficient candidates given by DEA. Proposed work considers the same method to find a best service provider for a user.

A. Ranked Voting Method

Ranked voting method, introduced by Tsuneshi Obata and Hiroaki Ishii, is as follows. Let m be the numbers of Cloud providers in market and $k(k \leq m)$ be the best Cloud providers

that a voter has to select and rank them from top to bottom i.e. from 1 to k. Let v_{ij} be the number of jth place votes of the candidate i where $i = 1 \ldots\ldots\ldots m$ and $j = 1 \ldots\ldots\ldots k$. Now preference score Z_i should be calculated for each Cloud provider i as a weighted sum of votes with certain weight w_j, i.e.

$$Z_i = \sum_{j=1}^{k} w_j \, v_{ij}.$$

After applying DEA, value of Z_i will be 1 for all efficient candidates. Let $\widehat{Z_o}$ be normalized preference score of efficient providers ($Z_i = 1$) that has to be calculated. Model for ranked voting method with discrimination of efficient candidates is as follows.

$$1/\widehat{Z_o} = minimize \; \|w\|, \qquad (1)$$

subject to

$$\sum_{j=1}^{k} w_j \, v_{oj} = 1, \qquad (2)$$

$$\sum_{j=1}^{k} w_j \, v_{ij} \leq 1, \qquad i \neq o, \qquad (3)$$

$$w_{j+1} - w_j \geq d(j.\epsilon), \quad j = 1 \ldots\ldots\ldots k-1, \qquad (4)$$

$$w_k \leq d(k.\epsilon).$$

where $d(\cdot.\epsilon)$ called discrimination intensity function is non-negative and non-decreasing in ϵ ($\epsilon \geq 0$) and satisfies $d(\cdot.0) = 0$. Constraint (2) is for efficient candidates, constraint (3) is for candidates who are not efficient and constraint (4) means vote of higher place may have greater importance than that of the lower place.

The normalized preference score $\widehat{Z_o}$ is obtained as a reciprocal of the optimal value. Provider with highest normalized preference score will be winner i.e. best Cloud provider for user.

VII. Conclusion and Future Work

Cloud computing provisions computing resources dynamically as a utility and resources are offered on pay-per-use basis. With a credit card any one can take services from cloud and deploy and configure servers in hours i.e. availability and functioning of computing resources in less time is now an easy task. So more customers are taking services from Cloud leading to the growth of cloud market which in turn increasing number of service providers also. Increasing number of service providers has created big confusion in selection of an appropriate service provider.

In this paper, QoS metrics are defined to measure the Cloud services and a framework is proposed for best provider selection using Ranked Voting Method. The contributions of this paper are discussed below.

Proposed work makes various QoS attributes like location of data centre, capacity, availability, certification etc. efficiently computable. It facilitates various providers specifying their QoS offers and various customers specifying their QoS requirements at different quality levels. It helps customers to understand each service provider clearly and also helps service providers to advertise their service in a better way. In turn, it also increases a healthy competition in Cloud Market.

Further, proposed work suggests a view of global cloud computing. Users have to inform their needs only and automated global market will discover and provide solution in the form of a best service provider. Users need not to look over the market to identify different offerings and compare them with their need. Global cloud computing also helps service providers to become a visible competitor in Cloud Market.

The proposed work adopted Ranked Voting Method to formulate best provider selection problem and developed a flexible framework (i.e. user can add or remove QoS metrics easily) for selection of a best provider which can be used for different applications with different QoS requirement. User can make their own voter list as per its requirement. A set of rules is defined to compare QoS values even though they may be of different value types. User may group more than one application to run on a single service provider or user may run applications on different service providers, the proposed framework suggests a best provider. Ranked voting method does not use inefficient provider's information to discriminate efficient providers therefore order of efficient provider never changes if inefficient providers are added or removed.

In order to study the performance of the proposed framework, future work will concentrate on the implementation part.

References

[1] T. Obata, H. Ishii, "A method for discriminating efficient candidates with ranked voting," European Journal of Operational Research, vol. 151, 2003, pp.233–237.

[2] J. Siegel, J. Perdue, "Cloud services measures for global use: the Service Measurement Index (SMI)," in Annual SRII Global Conference (SRII), IEEE, 2012, pp. 411–415.

[3] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, 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, 2009, pp. 599–616.

[4] https://cloudsleuth.net/global-provider-view

[5] S. M. Habib, S. Ries, M. Muhlhauser, "Cloud Computing Landscape and Research Challenges regarding Trust and Reputation," in Ubiquitous Intelligence & Computing and 7th International Conference on Autonomic & Trusted Computing, 2010, pp. 410–415.

[6] www.nirvanix.com/downloads/analyst-reports/nirvanix-vs-in-house-storage.pdf

[7] S. E. Kihal,C. Schlereth, B. Skiera, "Price Comparison for Infrastructure as a Service," in European Conference on Information Systems, Barcelona, Spain, 2012.

[8] http://www.cloudorado.com/

[9] M. Arlitt, S. Banerjee, C. Bash, Y. Chen, D. Gmach, C. Hoover,P. Mahadevan, D. Milojicic, E. Pelletier, V. RN, A. Shah, P. Sharma, "Cloud Sustainability Dashboard," International Symposium on Sustainable Systems and Technology, 2010, pp. 1.

[10] L. G. Williams, C. U. Smith, "Web Application Scalability: A Model Based Approach," in Proceedings of the Computer Measurement Group, Las Vegas,2004.

[11] S. Islam, K. Lee, A. Fekete, A. Liu, "How a consumer can measure elasticity for cloud platforms,", in Proceedings of the third joint WOSP/SIPEW international conference on Performance Engineering, Boston, Massachusetts, USA, 2012, pp.85-96.