SaaS Resource Management Model and Architecture Research

Zhang Xiaodong\textsuperscript{1,2,*}, Zhan Dechen\textsuperscript{1} and Chu Dianhui\textsuperscript{2}

\textsuperscript{1}School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, P.R. China; \textsuperscript{2}School of Computer Science and Technology, Harbin Institute of Technology at Weihai, Weihai 264209, P.R. China

Abstract: Nowadays many enterprises have to face a real problem in their operation processes, which is mismatch between active demands and resource. To solve this problem, traditional literatures employed Resource aggregation of Cloud Computing. However the most of these researches focus on computing resources but only few on application resources. And there does not exist a complete solution. In this paper, a resource service model RSM which is dynamic, customizable and fixable is presented. This model can be able to help enterprise add surplus resources into cloud resource pool. Then, a SaaS resource management architecture CARMA is built. The CARMA is used to solve the problems in sharing and management resource of enterprises. Meanwhile, resource optimization selection is taken into CARMA to reduce resource waste and task delay. Applied in the transport service field, practical work proved that the proposed models are feasible and achieve the expected effect.

Keywords: Private resource management, resource service model, SaaS resource management, service, service equivalent.

1. INTRODUCTION

Cloud computing is a kind of service resource pool in which a lot of resources are aggregated. Tenants dynamically select a number of resources to serve themselves according to their own demands [1, 2]. Cloud computing mainly provides the resources for computing, such as CPU, disk, memory, network, software, data and so on. The attribute of quality of service (QoS) is relatively simple. It generally uses time or cost as the parameters of QoS to establish the model which is based on performance or economic [3]. The purpose is to seek the shortest running time (the optimal performance). Besides computing resources, other shared resources are also required in the cloud computing based industry application system. For example, manufacturing cloud includes the hard manufacturing resources, the soft manufacturing resources and the manufacturing capacity [4, 5]. And logistics cloud includes the hard distributing resources (such as transportation vehicles, manpower, etc.), the soft distributing resources (such as data, software, knowledge, etc. in the process of transportation) and distribution capabilities (such as distribution, scheduling and integration, etc.) and so on. Enterprise’s adding some resources which can be shared to the cloud, and on the one hand, can improve the utilization rate of resources, on the other hand can also make cloud application more widely. However, the type of resources in application system is various and the number of them is large, their access and management are more difficult. In addition, there are some different constraints and evaluations between the user’s demands for the QoS and QoS of cloud resources in different fields. The requirement of users is not only for time and price, but usually including TOPS (Time, Quality, Price, Service etc.). Therefore, scheduling based on performance or economics cannot meet the actual needs of the enterprise.

For a case of transport of enterprise, there is usually such a situation: resources provided by Resource Service Provider (RSP) are idle, but enterprises who need these resources could not find them. The cause is as follows: 1) There exist poor information interaction and the information isolated island; 2) the best allocating solution, which meets the interests of both supply and demand sides at the same time, can’t be made while there are many resources and tasks. And at all the primary cause is: 1) resource management mode is incomplete; 2) resource management mode has some faultiness. So in this paper characteristics of the cloud resources are discussed and researched, then resource model, management architecture and service mode are constructed; and these are applied into the related field.

2. RELATED WORK

2.1. Resources Service Mode and Optimization Schedule

Resource is usually integrated into cloud environment via service mode. After combining with cloud computing and network manufacturing, Li Bohu et al. [4, 5] present a cloud manufacturing architecture and add virtual resource layer for the integrating resources. This architecture can access all kinds of resources by the cloud technology, achieve the svrtivation and virtualization of resources, and advances the ‘many to one’ service mode of ‘scattered resources concentrated use’ to ‘many to many’ service mode. Liu Lilan et al. [6] study the current networked manufacture and various patterns of manufacturing cloud, and propose the four levels manufacturing cloud architecture which uses the manufacturing resources as the bottom of cloud computing, emphasizes on that encapsulating distributed resource in the form of services and manage them in a centralized manner. Casati F
Table 1. Analysis of resource’s characteristics in cloud environment.

<table>
<thead>
<tr>
<th>Resource Feature</th>
<th>Features Cause</th>
<th>The Cloud Resource Management Technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autonomy of Distribution</td>
<td>In different organizations and geographic areas, the owner masters full knowledge of resources, can control and manage resources</td>
<td>Provide resources integration model, through the interaction with resource management system of different domain or resource calendar provided by RSP to schedule resources</td>
</tr>
<tr>
<td>Heterogeneity</td>
<td>A wide range of different characteristics</td>
<td>Define standard resource management model and information interaction protocol [12]</td>
</tr>
<tr>
<td>The diversity of technology</td>
<td>Different organizations have different resources management, scheduling and maintenance policies to resources</td>
<td>Make standard mechanism of resources and users’ demand expression [13] and build extensible resources framework</td>
</tr>
<tr>
<td>Dynamic</td>
<td>Resource’s configuration, ability, and running state dynamically changes constantly in the process of operation</td>
<td>Have a certain adaptive ability and can handle the fault tolerance of failure</td>
</tr>
<tr>
<td>Non-real-time</td>
<td>Production cycle is long, the process is relatively independent</td>
<td>Establish resource calendar, monitoring resources, scheduling resource</td>
</tr>
<tr>
<td>Perceiving</td>
<td>By the Internet of things and RFID technologies and so on, the current information of the resources can be gotten real-time</td>
<td>Access to awareness technologies of the Internet of things and RFID</td>
</tr>
<tr>
<td>Collaborative</td>
<td>More complex task requires resources from different providers work together to complete</td>
<td>Understand security mechanism, the resources characteristics of different areas, do task tracking, form &quot;many-to-many&quot; service mode</td>
</tr>
</tbody>
</table>

et al. [7] establish a matching and search framework based on the task’s functional requirements, and propose optimization algorithm based on intuitionistic fuzzy set resources. When resource is evaluated, trading experience, decay with time and other non-functional services quality attributes are added. This kind of methods cannot effectively guarantee displaying the service features comprehensively; pay more attention to the theoretical study of service selection rather than practicality and generality. In order to further improve the condition of resource using, scholars had studied a large number of algorithms, such as service network of the simulated annealing algorithm [8], ant colony system optimization algorithm based on trust perception [9] and genetic algorithm based on multi-objects GODSS [10], etc. The resource aggregation service chain constructed by these methods is the optimal single solution satisfying the constraint conditions rather than acceptable multi-solutions, which cannot fully show the personality of services and motivate service providers to optimize service quality. To make resources networked the most commonly method is to package the resource as a service. After that, many optimization selections and scheduling for service can be applied to the resource’s optimization scheduling. Tao F et al. [11] propose a group decision making fuzzy hierarchical analysis model based on service quality. The whole process of resource scheduling can be divided into resource search, resource scheduling based on the service quality and dynamic interaction and consultation three stages, and schedule resources with fuzzy analytic hierarchy process and group decision.

The above studies generally believe that the resource can be encapsulated as a service, which can effectively shield heterogeneity, distribution and diversity characteristics of kinds of different resources thus reduce the difficulty of the selection and scheduling algorithm. In practice, however, if we do not consider or less consider the related features of the resources, it will usually block the optimal selection of resources, thus cannot achieve the ideal effect. If those features are considered, the complexity of management will be increased. Faced with this situation, this paper proposes an extensible architecture, in which RSP and RSD can dynamically select the resource characteristics, build resource model, obtain optimal selection and scheduling scheme of resources through relevant algorithms at the same time of reducing management difficulty.

2.2. Analysis of Resource Features in the Cloud

The resources in cloud computing management that come from different RSPs are wide-area distributed, heterogeneous and dynamic. The cloud resource management doesn’t have a complete control of resources, and can’t predict the state of the resources. And the heterogeneous resources greatly complicate the resource management. All of above make managing resources and scheduling tasks be different. Therefore, this paper summarizes and analyzes the features of resource management in the cloud and gets the technologies that management in the cloud environment. All results are shown in Table 1.

3. RESOURCE SERVICE MODEL AND RESOURCE MANAGEMENT ARCHITECTURE

3.1. The Scene Description and Analysis

Firstly, the cloud resource service management should solve the problem of the service mode. So we need to analyze and design the resource management framework. The scene analysis is very useful to define the style of the system framework [14]. This paper confirms the framework style of resource management by the scene of Fig. (1). In the service management, Cloud Platform supports RSPs to register kinds of resources to the cloud to realize ‘the unified management to scattered resources’. At the time of sharing, RSP can also manage resources independently, which includes monitoring
the running state of the resources, allocating resources by utilization rate and benefit maximization, and cooperating with other resources. Cloud resources, packaged into service, can be assembled into different processes and serve for different users, which is different from network resource management mode. So the mode turns from ‘many to one’ into ‘many to many’. The system can receive feedback information coming from different users and formulate and improve service strategy specifically. RSD can submit tasks to the cloud and manage tasks independently. RSD can divide task, and make up the workflow. Then according to the needs, the tasks can be executed by the specific resources which are allocated independently by RSD, or can be assigned by the system automatically. RSD can monitor the running state of the tasks, investigate the task optimization solution and feedback the problems related with quality of service.

In most case, RSD need not to consider the discontinuity problems which emerge during running the tasks. When a resource produces a problem, the system will assign another new resources to execute the task dynamically and feedback the information to RSD timely. In other words, the cloud platform can provide the indirect guarantee for the trust relationship between supply and demand, which is decided by resource aggregation of cloud platform.

3.2. Resource Service Model

As shown in Fig. (1), the resource management should be divided into two parts: a part of it can show the functional management of resource purpose, including capacity, consumption, operations, etc., displays how to match the task; The other shows non-functional of monitoring and scheduling resource, including scheduling, coordination, control, optimization, etc., displays how to implement of task on resource. It can make better use advantages of resource only if both of two aspects are combined organically. The cloud resource management, therefore, can be able to aggregate different types of resources into the system and fully embody the features of them. On the one hand, it allows RSP to manage their resources independently; on the other hand it also allows the system and RSD to schedule resources according to the requirements of task. The paper builds a multi-tenant, customizable, scalable SaaS model resource model based on

Fig. (1). Cloud resources service operation scene.

Definition 1: Resource Meta Model RMM={G,P,T} where G is general resource type, each kind of resources has attributes. For example G = {resource identifier, and resource type, resource location ...}; P is private resource type, has private attributes and methods; T is resource types, different resource type has different private resources type.

Definition 2: Resource Private Type

Fig. (1) scene. Resource management’s goal is to make better use of resources. It’s closely related with activities of the task on the resources. Here elements (object) and the relationship between the elements in resource model are described by IDEF3, as shown in Fig. (2). In order to be able to describe more accurately of objects and relationship between them in the resource model, and ease computing, this paper makes the following definition:

Definition 1: Resource Meta Model RMM={G,P,T} where G is general resource type, each kind of resources has attributes. For example G = {resource identifier, and resource type, resource location ...}; P is private resource type, has private attributes and methods; T is resource types, different resource type has different private resources type.

Definition 2: Resource Private Type

According to definition 1 and 2, RSP can customize resource information according to their own resource characteristics, register their resources into the system, and manage them. Here Job is management task, its properties (JP) express the management task characteristics, and management tasks can be performed under a certain condition (JEC). Management tasks is a management work for resource, so it combines with resource to product an activity (UOB-J), and record the target (UJT) and process (UJP) of the activity. RSD can select the resource meeting demands (TCQ) from the resource pool to complete the task. Here demand refers to resource capacity. The condition of performing task is described by TEC, and task description is TP. The task per-
formance shall be done by the resource. Therefore, tasks and resources can produce a series of business activities (UOB-T).

The Fig. (2) shows that the interaction between the components is through interface, which simplifies constructing the following system architecture on the whole and improves the visibility of interaction. Reality and service provided by the components are decoupling, it promoted independent resolvability. In order to better show SaaS software engineering principles of design patterns, this paper introduces two symbols into the IDEF3, one is a component customization file (*.XML), another is the influence operator. Customization file makes different tenants (RSD and RSP) expression according to their own demands. And the influence operator expresses the influence of resource service quality evaluation (Ev) to model of resources and resource selection. Ev associated with the task (Task and Job). Data comes dynamically from two activities (UOB-T and UOB-J), and it impacts on activities (such as the choice of resource). In addition, it also reflects the influence on evolution of the resource model (such as increasing or decreasing the relevant attributes or methods).

Definition 3: Let R be resource set which meet demands of resource model RM in the cloud resource pool, T types of resources and E be service capacities. Kinds of resources to complete task is \( R_q = \{r_1, r_2, \ldots, r_i, \ldots, r_n\} \), where resource type of \( r_i \) is \( t_i \), capacities required is \( E_{qi} = \{e_{i1}, e_{i2}, \ldots, e_{ik}\} \). Relational operation \( \delta_{r_i} (R) \) is selection to resource type and the process of filtering the set by service capacity is \( \bigwedge_{i=1}^{n} \delta_{t_i} (\delta_{r_i} (R)) \) if and only if \( R_p \supseteq R_q \) (1).

Definition 3 is derived from definition 1 and 2, is one of conditions to perform activity UOB-T.UEC, where \( \delta_{t_i} (\delta_{r_i} (R)) \subseteq f_j, \bigcup_{i=1}^{n} \delta_{t_i} (\delta_{r_i} (R)) \subseteq f_j \). It can be seen by definition 3 that selection of resource may need to match many capacities of a resource. Thus, type and capacity conditions can’t be used to filter resources concurrently in general. Because different types of resources have different service attributes. It is difficult for computing concurrently. Obviously, this is decided by the resource model, since different types lead to different models. Thus, it can be proved that the resource model described by definitions 1 and 2 has a certain elastic extensible ability.

By the formula (1), the supply of resource is not exactly as same as demand of task. The amount of supply resource must be greater than the amount of demand resource; otherwise the task cannot be completed. Even \( R_p \supseteq R_q \) met, formula (1) has still the following problems:

(1) The demand and supply of the same resources are not considered in different period, i.e. resource calendar constraint is absent. Therefore, the result of selection may not complete the task. The result provided by the formula a necessary condition to meet the task requirements but not sufficient conditions.

(2) The quality, efficiency and cost etc. of resources to perform a task are not considered, so the result of selection may not be optimal.

(3) It is suitable for a single resource selection, but the efficiency is not high while selecting multiple resources.

If let schedule of task i be \( c_i \), it is easy to get the following corollary:
Corollary 1:

\[ R_p = \bigcup_{j=1}^{n} | R_j \cap \delta_{T_j,C_j} (R) | \] and there must be \( R_p > R_q \)

(2)

The evaluation of the resource service quality is mainly from the dynamic data of activities UOB-T and UOB-J. The following definition can be defined:

**Definition 4:** Let QoS of resource \( R-QoS = \{Q_1, Q_2, \ldots, Q_n\} \) is multi-dimensional, all of evaluation values are from the following formula:

\[ R-QoS = f_i, \left( \sum_{i} UOB-J, UBP \right) + f_i, \left( \sum_{i} UOB-T, UBP \right) \]  

(3)

Where \( i \) is a task sequence.

Definition 4 points out that the resource QoS is from analysis of historical data which resources have undertook tasks. It is an objective evaluation method of the resource QoS. Thus similar time efficiency, service, cost, reliability, availability etc. aspects of the evaluation of each resource can be obtained by it. While the task is assigned, the resource selection is optimized with formula (2) after filtering. At the same time, it also has the effect to monitor QoS. It can help system find imperfections and correct them in real time, and is an important part of the extensible mechanism.

As the above definition shown, although the general resource model is almost same, different types of resources can have a different description of resource private type. Capacity attributes of private resource type reflects service features of resources. Different resources’ service capacities are different, which support the operation of different resources. Capacity and private attributes are the resource static descriptions, is the basis of resource selection. With the resource calendar, various tasks are completed according to the different business processes. And all kinds of the UOB-T data are produced; it is a dynamic description of resources. The Ev gets these feedback data through the UOB-T interface to analyze aspects of the business, which is the perspective of the optimization scheduling resource from RSD. And UOB-J reflects more about autonomy of the resource distribution. RSP maintenances and manages resources from their own interests, determine whether to accept the scheduling, and transmit information to Ev management to analyze through the UOB-T interface, which is the perspective of the optimization scheduling resource from RSP.

### 3.3. Resource Management Architecture

The resource model can extend to be Cloud Application Resource Management Architecture (CARMA), as shown Fig. (3). This architecture can be approximately divided into several modules, such as resource management, task management, system maintenance, model parsing engine and load balancers, resource base, mate-model base, model base, domain knowledge base and so on. We do not introduce all
modules in detail and only describe some critical parts due to limited space. The model parsing engine f mentioned in RMM is the most critical part in CARMA. The domain experts make all kinds of meta-resource models and save the models on the provisions to the domain knowledge base. The model parsing engine displays these models on RSP(visibility). And RSP registers and manages relevant resources, which is parsed and executed by model parsing engine. Furthermore, this engine schedules the resources when we choose and optimize them in task assignment and load balancing. The model parsing engine shields the heterogeneity among different resources and the difference to the management technology, which realizes customized management to multi-tenant resources and improves the resource aggregation. The model parsing engine also can realize the virtualization resources under the control of resource calendar. The four bases (resource base, mate-model base, model base, domain knowledge base) can enhance the scalability of the system and improve more resource aggregation which guarantees that every task has many implementations (reliability). And based on this, we can realize optimization selection. The CARMA aggregation can make the resource capacity continuous in a wide scope(for example, the load of a truck changes continuously from 0.5 tons to 100 tons), which can simplify matching algorithm and improve the executable sequence probability of the task matching with resources (availability).

In order to support CARMA fully, this paper describes the resource managing layers architecture (RMLA) in detail, which is shown as Fig. (4). RMLA is divided into five layers: resources provision, resources aggregation, resources service, service workflow and service demand (task) from the bottom to up.

(1) Resources provision: it is the layer of executing tasks, which consists of physical resources provided by RSP.

(2) Resources aggregation: RSP registers resources into the system based on the management strategy of RSP and the integration rules of CARMA. The resources, redistributed by type, become a free and scalable layer of resources aggregation, which can reduce the interference between service ability and services selection and the difficulty of designing and realizing relevant algorithm. From the definition of meta-resource models, we know that registering resources can support large amounts of resources without the limitations of region, RSP and RSD’s demands, which guarantees the enough resource supply during the task assignment process [15].

(3) Resources service: it is called the layer of resource combination. And it can become a more powerful unity by the integration of one or more resources with their service characteristics. For example, the collaborative couplings to the forklifts load 1 ton and 90cm up, the trucks height 70cm and load 10 tons and the truck drivers become a group of transport service. It also supports RSP to package resources as a service, access the system directly, which is accessed through two different system resource layers.

(4) Service workflow: the packaged resource service based on business flow becomes a more perfect service which includes non-resource service. It is equivalent to a knowledge base composed of services.

(5) Service demand(task): users can depart a task into several subtasks on this layer. On one hand, it can choose relevant workflow model from service workflow layer by model matching. On the other hand, it can also choose resource service to compose directly based on the needs of the tasks.

The selection of resource service is a bottom-up process which passes five layers: task decomposition→workflow/service selection→service binding→resource binding→resource performing. The optimization of service workflow layer is the optimization of service path. And the optimization of service layer is the optimization of choosing resources ultimately.

4. RESOURCE SELECTION ON QOS AND SERVICE CAPACITY MATCHING
RMLA not only shows the idea of Resource management, but also shows the relation and connection between tasks and resource, including matching models from the service requirement (task) layer to the service process layer, selections from the service requirement layer to the resource service layer, optimal combinations of resource and tasks, and the task assignment, resource scheduling and related load balance. This section, taking the service management in logistics field as an example, discusses the foundation of implementing the processes above: matching of tasks and resource service capability and resource optimization based on service quality.

4.1. Resource Optimization Based on QoS

In RMLA, after filtered based on capacity matching of resource service and service by formula (2), many resources satisfying service quality can be elected, but not all of the resources are able to become finally service providers, only some of which are needed for the task. Therefore, the optimal group are demanded to be selected from the resources. Thus, the factors influencing the service quality need be determined and multidimensional service quality model need be built, as is state in definition 4. Every parameter in the service quality model is derived from evaluation results of service usage records by resource service evaluation model, as is obtained by formula (3).

For example, a five-dimension service quality model including time Qti(r, ti), cost Qco (r, co), reliability Qrel(r, rel), availability Qav(r, av) and reputation Qrep(r, rep) is represented as QoS = {Qti (r, ti), Qco (r, co), Qrel(r, rel), Qav(r, av), Qrep(s, rep)}. So, the formula (3) need to obtain information from UOB-T and UOB-J in order to provide the basis for resource selection.

In order to unify computing standard and simply the computing process, the ideal of service equivalent is involved, the definition of which is following:

Definition 5: Service Equivalent is quantization standard for service capability, which is the ratio of the capability of certain type of resource in industry standard units to the minimal resource capacity in this field, is \( \lambda \).

For example, the loading capacity for a lorry is 10 ton, if which for the minimal lorry is 5 ton, the service equivalent for the lorry should be \( \lambda_w = 10/5 = 2 \); If the minimal ship of a certain type of cargo can load 20 20GP containers, the service weight of the ship should be \( \lambda_{container} = (40 \times 20)/(20 \times 20) \). In general, introducing the ideal of service equivalents can simplify the complexity of calculating service capabilities.

Similar with service equivalent, ordinal utility function is often used for service election in micro-economics, which can provide weak ordinal relations in numerical value and promote rational consumption for service. Constructive Ordinal Utility Function is used herein as a scale of service resource to obtain the optimal solution of service resource selection.

Definition 6: Resource Optimization Model Based On QoS (ROMBOQ), let \( R = \{r_1, r_2, \ldots, r_n\} \) is the set of candidate service resources obtained by formula (2). \( Q = [q_{ij}]_{m \times n} \) is QoS decision matrix, where \( q_{ij} = Q(r_i, j) \) is resource \( r_i (r_i \in R) \) is valued from QoS attributes \( q_i (q_i \in QoS) \), obtained by formula (3) under data normalization. In constructive ordinal utility function \( f(r_j) = \sum_{j=1}^{n} w_j q_{ij} \), \( w_j \) is weight of attribute of service resource, \( j \), \( i=1,2,\ldots,m \).

According ROMBOQ, following decision model can be obtained:

\[
S_{opt} = \min (f(r_1), f(r_2), \ldots, f(r_n)) = \sum_{j=1}^{n} \sum_{i=1}^{m} (q_{ij} - q_j^*)^2 w^j
\]

\[
\begin{align*}
\sum_{j=1}^{n} w_j &= 1 \\
&\text{s.t.} \quad w_j \geq 0, \quad j = 1,2,\ldots,n \\
E_q(r, \lambda_j) &= \sum_{i=1}^{m} E_p(r_i, r_j)
\end{align*}
\]

where \( q_j^* = \max(q_{1j}, q_{2j}, \ldots, q_{mj}) \) is the ideal value of attribute \( q_j \) in decision matrix, the objective function \( S_{opt} \) restrict the ideal value in the minimal variance with attribute \( q_j \) for other candidate services. So, the ordinal service resources \( R \) satisfied that: \( \forall r_j, j \in R \), if and only if \( f(r_j) \geq f(r_i) \), the resource \( r_i \) is better than \( r_j \); where \( w_j \) is weight of attribute of service resource, \( j \).

4.2. Evaluation of Resource Service Quality

In the whole Cloud, registered resources in different location form a huge and complex network by their related business. It is important to monitor, estimate and modify the network as well as pretty complex. It can help observer obtain the situation of resource service, find the advantage and disadvantage of the algorithm and cope with the problems and modify the algorithm on time. The evaluation herein is divided into 3 aspects: the consumption of resource individual service capability; the consumption of the whole resource service capability; the load balancing of resource.

(1) Resource utilization rate: it reflects the task saturation degree for resource in the network, obtained by the following formula:

\[
CoS_r = \left( \sum_{j=1}^{n} \left( \delta_j C_{qj}^{\sum}(r_i, \lambda_j) \right) \times \frac{t_j}{T} \right)
\]

where \( C_{qj}^{\sum}(r_i, \lambda_j) \) is the service capability in use and \( C_{qj}^{\max}(r_i, \lambda_j) \) is the maximum service capability provided by
resource j. $\delta_j$ is the importance of the service capability in the service process. For example, the regulation rules that the weight baggage for airline should be no more that 45kg and the sum of height, width and length should be no more than 203cm, in reality, the maximum of which will be the limitation and $\delta_j$ should be 0 or 1. Formula (8) shows the ratio of service resource supply to service resource consumption in a certain period. The more the ratio is, the more loading the certain individual is and the higher resources utilization rate is. CoSp is named as the saturation of resources load. T is the work time designated by RSP, $t_i$ is reality work time in T. For instance, if a RSP designate its resource i the range of work time from 8 am to 6 pm and the actual work time is 4 hour, there will be $t_i/T=4/10$. Obviously, every RSP sets different work time period for different resources that is the usage of resource calendars concerned in section 3.2, which is a data structure with SaaS characteristic.

(2) load balancing of resource it reflects the equilibrium of resource load in network, which can be obtained from the saturation of individual resource service capability, the formula is following:

$$\text{avgl} = \frac{\sum_{i=1}^{n} \text{CoSp}_i}{n} \quad (9)$$

$$B = \sqrt{\frac{\sum_{i=1}^{n} (\text{CoSp}_i - \text{avgl})^2}{n}} \quad (10)$$

As is shown from formula (9), (10), the smaller B is, the more load balancing of the resource will be. It is improper here to use absolute resource load as the standard, because that the resource load cannot be balanced when resource which has large service equivalent loads as many as resource which has small service equivalent do. For example, if a 5-tone lorry carries 2-ton cargo as the same time a 2-ton lorry also carries 2-ton cargo, it is an unbalanced tasking with serious resource waste. So, individual service saturation is used to calculate relatively load herein.

5. EXPERIMENTAL METHOD AND RESULTS ANALYSIS

In order to test the correctness of the model, BirisCloud of Harbin Institute of Technology was used in this experiment as the base platform, where CARMA was built, RMLA was realized and various resource models were built. Then, follow three representative types of enterprises were compared: (1) petty individual-owned transport agent with no more than 5 vehicles in no more than 2 types; (2) small transport company with 10 to 20 vehicles in no more than 5 types; (3) large transport enterprise with more than 80 vehicles in any type. The emphases of resource management for the 3 types of enterprises are different. Thus, it was permitted that the enterprises can customize and management the attributes of the general resource type, private resource type in convenient for the cooperation and performance comparison of enterprises. The capability attribute, UOB-T and UOB-j were set by industry standard. In this perspective, building RMLA both helps for the individual management of enterprises for resources and improves the management mode. It enhances the competitive power of enterprises in market eventually. Data filtering is done when all of RSP, RSD and related industries are involved into the platform and pass to stable phase. According to the estimation model mentioned in section 4.2, experiment and results analysis are carried on two aspects.

5.1. Resource Utilization

By formula (8), a longitudinal comparison among the 3 type of enterprises was done, as is shown in Fig. (5). Before joining the Cloud, there were lowest load saturation and stability for type (1). Otherwise, the load of type (3) was the most stable. As is seen from the left line chart in Fig. (5), there were large quantities of tasks in type (1), but cannot cooperate with type(3) which was in eager of tasks. This verified the problem mentioned in section 1. The right of Fig. (6) shows that business of every type of enterprise has been improved. The state of individual-owned agents has been closed to that of small companies. And rate of the load saturation increasing of type (2) is obviously larger than that of type (1), which is related with the reputation and market recognition.

5.2. Load Balancing of Resource

Due to the load saturation used as parameter, value B is always lower than one. Therefore, small change derived from the variance may reflect a great load difference. For example, if an average saturation was 0.3 and the task af-
forded by a vehicle was 0 in a certain day, the perturbation would only be 0.09. Absolutely, this result also showed that the whole utilization of resource was low. By service tracking and data analysis, result in Fig. (6) was obtained. Due to the standard scheduling policy for resources, employers and tasks in type (2) and (3) of enterprises, the load is far more balanced than that of type 1. After joining the Cloud, the load balancing of type (1) improved a lot (Pay attention different numerical range between the left and the right in Fig. (6)). It also indicated the management mode of medium-sized and small enterprises has improved.

CONCLUSION

This paper focuses on the characteristics of resource management in Cloud and builds resource model. Based on these, a resource management architecture is constructed. Meanwhile, this paper proposed evaluation method of this resource management. Then, the solution in this paper is implemented on BirisCloud platform and following advantages are reflected during the application process: 1) Good openness and extendibility: It mainly behaved in the ability of customizing and adding new resources, resource service capability and quality with no change in model architecture and keeping the disturb from resource heterogeneity; 2) The solution combines distribution self-management and unified scheduling and is able to monitor and schedule the resource and balance load dynamically, adjust the relations between high efficient operation and safety maintenance, reduce the business risky of enterprise; 3) Improving the resource utilization significantly: the solution both can adjust the supply and demand relations between service consumers and service providers, enhance the providers to improve the quality of service and promote the cooperation among enterprises, raising the competitive power of enterprises. The practical work proved that the proposed models and architecture are feasible and achieve the expected effect.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

The national science and technology support funding project (2012BAF12B16), the national natural science foundation of China (61273038), the scientific and technological torch-plan project of Shandong Province (2010GZX20126).

REFERENCES


Received: September 16, 2014 Revised: December 23, 2014 Accepted: December 31, 2014

© Xiaodong et al.; Licensee Bentham Open.

This is an open access article licensed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/3.0/) which permits unrestricted, non-commercial use, distribution and reproduction in any medium, provided the work is properly cited.