Formal Behavior Modeling: Business Processes Based on Cloud Platform

Bo Huang¹, Yong Xu², Mengting Yuan¹, and Guoqing Wu¹

¹School of Computer, Wuhan University, Wuhan, China
²School of Mathematics and Information Sciences, Zhaoqing University, Guangdong, China

Email: huangbowhu@gmail.com, {ymt,wgq}@whu.edu.cn, xyowhu@gmail.com

Abstract—From a macro level, cloud computing is gathering all the distributed resources originally, and then provided them as a service to the users. Current research focuses the cloud architecture and local small-scale grid computing mainly. The researchers are ignoring to provide a channel to help users to understand the business processes of the entire platform. This article focuses the communication and business links and abstracts them into business processes behaviors. At the same time, this article provides a modeling method to help these behaviors being visual depiction at the initial design of the cloud computing. We combine the SOA architecture and cloud computing together. At last the transferring and the dynamic processing of the service/data of the business processes can be shown to the stakeholders to verify whether the cloud platform was meeting their requirements which are regarded as simulation of cloud services.

Index Terms—SOA, business processes, BDL4CM, cloud model, cloud behaviors

I. INTRODUCTION

Cloud computing is the development result of utility computing, distributed computing, grid computing, virtualization and SOA technology. It manages a large number of highly virtualized computing resources, forms a large pool of resources and provides users with computing services through the network. SOA and cloud computing have a certain similarity. Both of them stressed the concept of service. The basic element of SOA is software - oriented service, and cloud computing is an extension of the IT infrastructure for SOA concept. Cloud computing which takes all computing resources (including hardware and software) as a service is a more efficient and more economical choice of architecture of SOA [1-3]. The SOA is more strategic and abstraction. In contrast, cloud computing is more tactical and specificity. The behaviors of cloud further enrich and expand the resource-sharing content and service patterns of cloud computing. They can be considered to be the classes of the object-oriented design. When SOA is described as an abstract class, the cloud computing can be seen as a subclass of SOA or a structure instance [4]. On this basis, the service itself can be seen as the behavior on the cloud platform level [5]. The implementation of behaviors, in fact, is the execution of the service.

Cloud Computing Service mainly consists of Infrastructure as a Service (IaaS), Software as a Service (SaaS) and Platform as a Service (PaaS) [7]. The resources include hardware, platform and software. Resources for cloud platform are not only the computing resources (such as storage unit, ALU and so on), but also the broad hardware resources. The resources have a variety of types and are distributed. Different resources have separate functions and properties. Resources are exclusive while working, that means, which cannot process two or more tasks at the same time. A task flow is ordinarily made up of several working procedures with certain order. Each working procedure needs unique resources to process. Problems of communication between entities exist in reality, so the influence of entity logistics interactions on process time and cost need to be considered.

Currently, Amazon, Google, IBM, Microsoft, Sun and other companies have put forward their own cloud computing infrastructure or cloud computing platform. In most cases, their cloud computing infrastructure and cloud computing platforms bias towards commercialization, but after studying their cloud platform can also help us for the cloud service model research.

(1) Google ‘s cloud computing infrastructure [8]
Google’s cloud computing infrastructure’s gradual expansion is based on search application as the service in the initial. The Google experts believe that the role of cloud computing is data. This platform is made up by a distributed file system-Google File System (GFS), large-scale distributed database-BigTable, programming model-MapReduce and distributed lock mechanism-Chubby. This platform is mainly used to deal with large-scale data.

(2) IBM’s "Blue Cloud" computing platform [9]
"Blue Cloud" computing platform is consisted by a data center, IBM Tivoli Monitoring, IBM DB2 database, IBM Tivoli Provisioning manager, IBM WebSphere application server, open-source virtualization software and open-source information processing software. IBM’s "Blue Cloud" is a leading provider of cloud computing architecture to build large-scale data processing services to users.

(3) Sun's cloud infrastructure [10]
Sun's cloud infrastructure is a six-level pattern, which is services, applications, middleware, operating systems, virtual servers and physical servers.

(4) Amazon's elastic computing Cloud [11]
Amazon is one of the earliest companies to provide cloud computing service. The elastic compute cloud (EC2)
platform is established by the large-scale computer server cluster in-house. It provides users the "cloud" running virtual machine instances with the operation of the network interface. Users only need to pay for the instances. The billing comes to an end after the end of the run of the "cloud". They believe that the calculation is the core of cloud.

These cloud platforms for users are too difficult to understand. The user is only concerned about what kind of service is satisfied his needs. In Saas situation, cloud platform includes a number of software entities. The services which are provided by the software are call cloud services. When cloud services come into a small granularity, the service itself has become the basic unit on the cloud platform.

When users need to access to a range of services in the cloud platform, they are willing to get the information that what kind of work the services can be able to complete in a similar way of workflow method. So it is necessary to modeling the cloud service. At the same time, this cloud model should be verified and simulated to ensure that the cloud model is satisfied the requirements of users.

On the other hand, SOA and cloud computing are complementary. Cloud computing provides the remote cloud service which is available to SOA, and SOA provides the method of the cloud service composition for complex business application requirements. At the same time, their focuses are different. SOA focuses on the system design which adopts the service architecture [1]. It emphasizes how to process services and its reusability, agility, loose coupling, etc. But cloud computing places emphasis on the provision and usage of service [12-13]. It focuses on how to provide service and its virtualization, on-demand dynamic extension, resources, services and so on. Facing the requirements of "cloud computing" and "service-oriented" of cloud modeling, it is necessary to merge them [1, 14-15] together.

Based on the main idea above, the paper merges the cloud platform and SOA to build a new architecture of the cloud services based on the system architecture. The new architecture is developed from the traditional SOA design pattern. Meanwhile, it is necessary to provide a visualization mechanism to help users understand the model.

The structure of this paper is as follows. Section I is an introduction and the correlational research; Section II introduces the Cloud-BDL model based on the cloud platform; Section III takes the behavioral model as foundation, introducing the visualization mechanism for Cloud-BDL modeling; Section IV introduces a modeling sample of a Cloud-Bookstore. And we discuss the feedback of the modeling method; Section V gives a conclusion and future work.

II. CLOUD-BDL MODEL BASED ON CLOUD PLATFORM

In order to modeling a cloud model based on the behavior of the business processes, Firstly, we have to create a cloud services architecture. Secondly, a description language for the business processes of the cloud platform has to be established. Modeling method mentioned in this article is using a Behaviors Description Language for Cloud Model (BDL4CM). BDL4CM has simple syntax and strict semantic to create business processes model for a cloud platform.

A. Cloud Architecture based on Business Processes

The core of SOA is service [2]. There are no technical limitations for services to achieve. The services can be achieved with a variety of languages. Traditionally, SOA service is using web services technology which uses a series of standards and protocols to achieve related functions. The basic protocols are including Simple Object Access Protocol (SOAP), Web Services Description Language (WSDL) and Universal Description Definition Integration(UDDI). SOA is inspired from the web services standards, and defines the inchoate SOA model as a web-based architecture model which is composed by service provider, service requestor and service registry center, which is shown in Fig. 1.
can discover and access the services. Service requesters (consumers) have to find out the required services through a UDDI query, and bind/invoke them by using SOAP according to the interface contract. Service registry contains a repository of available services. It is used to catch the interface of the service providers for the service requesters. In the cloud platform of this article, the three parts is called cloud provide end, cloud request end and cloud services operators, which are shown in Fig. 2.

Cloud computing platform focuses on IaaS, SaaS and PaaS, which can be seen in Fig. 3. By the references of SOA and cloud computing architecture, this article presents a new cloud service architecture. In Fig. 4, the architecture reflects the integration of SOA and cloud computing, and the extends/expands traditional computing resources to the cloud resources. The left of figure 4 represents the relationship between the three participants (service providers, service requestor and service registered agent/service operators) and three basic operations (publish, discover and bind). (Cloud) Service provider forces resources and capability to become cloud services (referred to as service) and publish them through virtualization and servitization. (Cloud) Service requester defines a business process based on business requirements and finds the service by (cloud) service agent. The service bus, infrastructure and cloud service operators and other support provide services for the service requester as required.

![Figure 4. The cloud service architecture in this article.](image)

<table>
<thead>
<tr>
<th>Name of Cloud Behavior</th>
<th>Expression</th>
<th>Demonstration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atomic Cloud Behavior</td>
<td>$ABehID: F(\text{Sub}, \text{Obj} &amp; \text{Obj explanation})$</td>
<td>F: operation or action of the cloud system; Sub Obj: cloud services subject/cloud services object; When: precondition, as the pre-requirements of a cloud service; INFrom/OUTTo: input/output of cloud service, the parameter mark to obtain &amp; invoke &amp; send the object; ID: label of atomic cloud behavior;</td>
</tr>
<tr>
<td>Null Cloud Behavior</td>
<td>$ABehID: \text{Idle}$</td>
<td>Return to a atomic cloud behavior;</td>
</tr>
<tr>
<td>Ending Cloud Behavior Of Composite Cloud Service</td>
<td>$ABehID: \text{Return}(ABehID)$</td>
<td>Return ( )</td>
</tr>
<tr>
<td>Simple Cloud Behavior</td>
<td>$\implies ABehID$</td>
<td>A atomic cloud behavior forms a simple cloud behavior;</td>
</tr>
<tr>
<td>Composite Cloud Behavior</td>
<td>Sequence Cloud Behavior</td>
<td>$\implies \neg ABehID &amp; \neg ABehID$</td>
</tr>
<tr>
<td></td>
<td>Parallel Cloud Behavior</td>
<td>$\implies \neg ABehID &amp; \neg ABehID &amp; \ldots \neg ABehID$</td>
</tr>
<tr>
<td></td>
<td>Non-determinate Cloud Behavior</td>
<td>$\implies \neg ABehID &amp; \neg ABehID &amp; \ldots \neg ABehID$</td>
</tr>
<tr>
<td></td>
<td>Determinate Cloud Behavior</td>
<td>$\implies \neg ABehID &amp; \neg ABehID &amp; \ldots \neg ABehID$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\text{IF } b \text{ THEN } \text{BehID} \text{ ELSE } \text{BehID} \text{ FI}$</td>
</tr>
</tbody>
</table>
In this paper, the new cloud service architecture (referred to as CSA) has the following characteristics:

- CSA is developed from the integration and the inheritance relationship between SOA and cloud computing. So CSA also can be seen as the results of which cloud services extend and expand. CSA on one hand merges the (cloud platform) services by the SOA specification. On the other hand, according to the concept of cloud computing, the CSA should centralize the services for intelligent management and operation. So that, users can get the full life-cycle services.
- CSA reflects a loosely coupled hierarchical architecture, some layers resource objects, virtualization, service-oriented business processes and interface from a functional aspect for the view of the service provider and service consumer. At the same time, service integration, business intelligence, infrastructure, cloud service management and other shared facilities are extracted to form different vertical level for all services.
- The CSA reflects an integration of SOA, cloud computing, networking and intelligent science technology. Virtualization service environment is rapidly constructed by SOA for enabling technology; Cloud computing technology provides the enabling technology for resource intensive operations; management and services as required. IoT technology provides the enabling technology to the interconnection between all types of objects.
- The core of the CSA is cloud service. SOA architecture is extended to enterprises remote cloud service providers by CSA for deploying the appropriate resources, services and applications based on business requirements, and be continuously adjusted, dynamic scalability and high fault tolerance and reliability.

B. BDL4CM

In practical work, researchers need to mode the cloud services business process with CSA. BDL4CM in this paper is a semi-formal modeling language. It is similar to BPEL for the description and the specification of business processes. The work which is done by all these business processes is regarded as business behaviors on the cloud platform. Thus, we define 5 benchmark cloud behaviors: atomic cloud behavior, null cloud behavior, ending cloud behavior of composite cloud service, simple cloud behavior and composite cloud behavior. Composite cloud behavior which is in different logical interaction point to four basic relationship.

The grammatical structure of BDL4CM for detail can be seen in Table 1.

III. VISUALIZATION MECHANISM FOR CLOUD-BDL MODELS

Generally, visualization of a model is often represented as a description of a static state chart (such as a point-line-edge structure). In this section, a visualization mechanism is contained to drive the BDL4CM model, and generates a simulation program to help users to improve the understanding of the process of cloud services. The visualization mechanism is seen in Fig. 5 below:

For BDL4CM model, users have to refine and process it to obtain an execution model with high accuracy. Then, map it to a finite automaton, and execute the states to drive the model to get the simulation of business processes in a cloud model.

A. Modeling

In the Fig. 5, we need to get the BDL4CM cloud model through modeling. In section II, we have already introduced the grammatical structure of BDL4CM which have 5 benchmark cloud behaviors: atomic cloud behavior, null cloud behavior, ending cloud behavior of composite cloud service, simple cloud behavior and composite cloud behavior. Composite cloud behavior which is in different logical cloud environment has to express different logical interaction of the cloud services. In this paper, the different logical interaction point to four basic relationship as which the definition is: sequence service relationship, parallel service relationship, non-deterministic choice service relationship and deterministic choice service relationship.

In every cloud platform, the modeling method for the cloud business processes is also a very important part of it. Taking into account the characteristics of BDL4CM, the model established by using our method is very similar to a process model, such as LSC [16]. The specific steps of modeling can be shown in Fig. 6, which is divided into two parts: 1. requirements specification elicitation and analysis of cloud services; 2. modeling, verification, simulation of the cloud model.

Requirements specification of cloud services, in fact, is a description of the business process of a cloud platform. This paper takes cloud services themselves as a starting, firstly analyzes subjects and objects of cloud services, and then the parameters of cloud services which are transferred between subject and object.

After established the cloud model, it is necessary to verify it. Verification can be based on formal model checking or dynamic simulation. Formal verification is
based on the BDL4CM which is a semi-formal language with formal semantics and validates some of the characteristics of the cloud model. The dynamic simulation, in fact, after the formal verification, is the simulation of the cloud model. The main purpose of the dynamic simulation of cloud services is user-oriented, which is designed to help users more effectively understand and analyze the target cloud services.

Therefore, as a cloud service modeling language, BDL4CM is fully equipped to describe the semantics foundation of cloud business processes which are regarded as business behaviors on the cloud platform. At the same time, due to the dynamic semantics of BDL4CM for describing the change of cloud behavioral expressions, and close to the dynamic semantics expressed by calculus of communication systems (CCS) [17], the operation semantics of cloud services and cloud model can be available. All of these can provide a better basis for validating various characteristics of the target cloud.

B. Verification

In Fig. 6 of this paper, developers still be unable to determine the correctness of cloud services model after established a preliminary model. There are two ways to validate the cloud model: a formalization method for validation and demonstrating simulation animation to users to check the correctness of the cloud model.

Dynamic semantics of the cloud model, similar to the dynamic semantics of CCS model, can be converted into CCS model in Fig. 7. At the same time, the user-defined characteristics of the cloud services (such as consistency, non-termination, credibility, etc.) can be described by formal behavioral temporal logic and get the corresponding characteristic expression. This paper argues that, when a model $M$ meeting the expected running trail (the behavioral temporal logic formulas), the established model is valid.

C. Simulation of the Cloud Model

Simulation (such as animation in [19]) of the cloud model is designed for model-driven. Simulation provides project-related personnel with an avenue to communicate and a platform to verify the correctness of the information of cloud services. It can obtain the high-quality requirement information of the cloud model and express/understand users' requirements accurately. It also can ensure that the cloud model and requirements specification have accurately expressed the true wishes of users. Combination of simulation and formal verification methods as well as application of graphs and images, could increase readability of requirement model of cloud services, help non-professional users and project-related personnel get a better understanding and communication of the cloud model. Much experience has shown that, if developers could find errors in the modeling phase, it is very beneficial for the conservation of the future development costs and improvement of target cloud quality.

As the cloud behavior of the cloud services acquires such properties as dynamic, timing and interaction and so on, it is necessary to study cloud behavior of cloud services especially structural relation of cloud behavior for cloud services, the interaction of composite cloud behavior, as well as BDL4CM in order to establish the cloud model of cloud services. Of course, when creating new cloud model based on BDL4CM, developers need to study the characteristics of simulation so that the new method could be used independently for modeling and also could be used to describe the key information that simulation implementation needed. It also helps enhance isomorphism of BDL4CM model and the simulation model, and simplify the model conversion and improves the degree of automation of the conversion. Simulation mechanism could be used to describe the cloud behavior of the simulation model. These actions could describe such static information as the structure of cloud platform

In author's article (which can be seen in [18]), there is a detailed description of the first method. The formalization method to verify the correctness of the cloud model is always the mainstream method in software engineering. Which in this article, we use the idea of model checking to verify the description of cloud services of the cloud model.

Figure 6. The specific steps of modeling.

Figure 7. The model checking of the cloud model.
D. Execution Engine

Simulation execution engine of cloud services in Fig. 8 is used to explain the simulation description model of the cloud model, and control the actual implementation of simulation in a stage as an interface for users in different situations. Therefore, researchers firstly need to study how to interpret the simulation description model and generate the actual operation of the simulation model. Then, researchers should turn to study how the simulation engine runs simulation model according to the actual operation of the simulation model, organizes arranged simulation component and performs the simulation program. In addition, the execution engine also needs to adjust the execution path based on users’ feedback. When execution arrives in the wrong state, the execution engine needs to submit bug reports, indicating that the existing problems, guide developers to correct service process.

This topic chooses state machine model as simulation description, which is using the transition of the state machine control the state changes of the various elements during the execution of simulation. The transition of the state machine is regard as the cloud behaviors. The execution of the transition can be seen as the function of cloud services.

Firstly we need to create a formal model that describes state machine, using the following (1) of state structure to describe the structure of the state machine,

\[ M_1 = s | M_1 < M_2 > | M_2 | M_1 || M_2 | M_1 + M_2 \]  

A structural formula \( M \) represents a state machine. A state structural formula can mean a normal state \( s \), or a compound state \( s < M_i > \) (the internal structure of compound state is \( M_i \)). The symbols of \( , || , + \) means the sequence, concurrent, or selecting relationship between the two sub-state structure. In addition, in order to emphasize the transition of a state structural formula, we could use \( < T > \) \( M \) unambiguous case to represent that transition \( t \) is the move into transition of state structural formula \( M \). The state structural formula can accurately describe the structural characteristics of the state machine, and can describe the transition rules of the state machine. State structural formula makes itself easier to be converting from cloud behavioral models to the state model and creates model transformation rules which are based on two formal systems for automatic conversion.

Atomic action of cloud behavioral expression corresponds to a transition of the state machine, after the execution of an atomic cloud behavior and prior to the execution of next atomic cloud behavior, this state corresponds to a state of the state machine. The cloud behavior expression’s structures (such as sequential, concurrent and selection structure) respectively correspond to the state structural formulas.

In order to ensure the correctness of the conversion from the cloud behavioral model to a state machine model, we could establish the bisimulation equivalence relationship between the two models, which can be seen in Fig. 9. Supposing there is a cloud behaviors model \( B \) and a state structural formula \( M \). In the execution process, after an execution operation of an atom action, the cloud behavioral model \( B \) is converted into model \( B' \). Then \( M \) can perform a corresponding transition \( t \). The transition after execution makes the state structural formula transformed into \( M' \) which can be seen be Fig. 9. \( B' \) and \( M' \) also have such a equivalence relationship, or vice versa. So the relationship between cloud behavioral model \( B \) and state structural formula \( M \) is bisimulation. Bisimulation equivalence relation can guarantee the consistent of the demonstration of the behaviors between the two models.

In addition, in Fig. 5 and Fig. 8, after extracting the state blocks of the object from the state machine, developers should determine whether the transition will be added to the state of the object’s block according to the rule that weather transition of the state machine causes the changes of the state of the object or not. Every extracted state block must be consistent with the original state machine behavior. The extraction algorithm can be detected by using a one-way bisimulation equivalence relation.

Execution engine interpreted simulation cloud model posed by the BDL4CM, generated the running real-time model (RM), and controlled the operation of the state program. RM, which is composed by a plurality of state
blocks, reflects the snapshot of simulation of the cloud model at runtime. In order to express the operating state and transition process of the cloud services. Each simulation component corresponds to a state machine, each state machine in the runtime of the program saves the current state of the simulation component. When a cloud behavior occurs, the execution engine would keep subject service and drive the state machine. At the same time, it commands the various components in the stage to demonstrate which is in accordance with the appropriate action.

According to the description of the simulation of the cloud model, simulation execution engine associated cloud behavior with the action of components and associated subject and object of simulation with simulation component. Through the state machine for each simulation component structure, it generated the simulation model and the initial stage. Then, according to the executable statement of simulation of the cloud model, it gradually controls simulation process.

Human-computer interaction is an important function of the cloud model simulation. In the implementation process, when confronting the subfield, it is sometimes necessary to select and give the rights for choosing back to users, so developers need to add human-computer interaction mechanism in the execution engine. Furthermore, developers also need to design a control clock which is used to control the execution of the simulation in fixed time segment. In order to facilitate the developers to make simulation debugging, the control clock uses time segment in steady interval to control the simulation progresses.

IV. A MODELING SAMPLE

In this paper, book sales management in a cloud bookstore is introduced in this section for business process modeling and simulation which are based on the cloud platform. Cloud bookstore business process model is shown in Fig. 10.

In this sample, the cloud bookstore needs to invoke different cloud services. The cloud model is established by a sales scenario. In this scenario, which is hypothetical, we assume that there are three atomic cloud services in this cloud model: Book-Buy Service, Out-Storage Service and Logistics Service. The three atomic cloud services are exhibited as the workflow process similarly.

From the aspect of the business processes, all cloud services can be regarded as the nodes of the business processes layer in Fig. 2. Corresponding to this article can be shown in Fig. 12.

As described in the previous section, this paper drives the model and generates the corresponding simulation program. Fig. 11 shows the simulation which is based on model-driven. The cloud bookstore simulation can be performed dynamic in our tool system, as a play. More details about the tool system can be seen in [20]. The data is transferred between subjects and objects, such the booklist as a parameter, and received by cloud services. Cloud services are the operations between subjects and objects. In our tool, the transferring of the parameters and the cloud service operations are dynamically manifested.

We also have completed a variety of examples based on cloud services which are as follows:

- The modeling and simulation of the campus cloud
- The cloud model of the logistics enterprise's warehouse management platform

From the feedback of users, it is generally believed that our modeling method can effectively improve the understanding mastery of the users on the target cloud system.

V. CONCLUSION

This paper presents a new train of thought to the design and implementation of cloud-based platform for business process. A semi-formal modeling approach is used to deal with business process. The approach is mainly based on the initial design for professionals and the user-oriented
system simulation. The simulation for cloud model is the blue print to complete the establishment of a target cloud. Compared to some other tools, this work is sound and effective. Future work will deepen and refine the modeling of the cloud platform.

ACKNOWLEDGMENT

This research is supported by National Natural Science Foundation of China (91118003, 61003071), Fundamental Research Funds for the Central Universities (3101046, 201121102020006), and Special Funds for Shenzhen Strategic New Industry Development (JCYJ20120616135936123).

REFERENCES


