RESEARCH ON KEY TECHNOLOGIES OF RESOURCE MANAGEMENT IN CLOUD SIMULATION PLATFORM

Ting Yu Lin [(a)], Xu Dong Chai [(b)], Bo Hu Li [(c)]

[(a)] School of Automatic Science and Electrical Engineering, BeiHang University, Beijing 100191 China
[(b)] Beijing Simulation Center, Second Academy of Aerospace Science & Industry Co., Beijing 100854 China
[(c)] School of Automatic Science and Electrical Engineering, BeiHang University, Beijing 100191 China

[(a)] lintingyu2003@foxmail.com, [(b)] Xdchai@263.net, [(c)] bohuli@moon.bjnet.edu.cn

ABSTRACT
For the diversity and the life-cycle dynamics of the resources on the cloud simulation platform, this paper discuss how to cross and integrate all types of resources management systems to realize the centralized management of distributed resources and then supply services of unifying managed resources to distributed users efficiently. This paper proposed the guideline and architecture of the cloud simulation platform at first. Then, from the perspective of resource operation and running, present the unified simulation modeling to support the dynamic management for the life cycle of the resources. After that, based on the concept of the resource group, this paper proposes the resource selection technology to support the efficient allocation for types of the resources. Finally, this paper introduces an application in the collaborative design and simulation for the multi-disciplinary virtual prototype.

Keywords: cloud simulation platform, resource management, unified modeling, resource selection

1. INTRODUCTION

Cloud Simulation is a service-oriented, intelligent, agile, green, new networktized simulation paradigm. Combining with the emerged information technologies such as cloud computing, service-oriented, virtualization, high performance computing, and developing the existing networktized modeling and simulation (M&S) technology, cloud simulation encapsulates the simulation resource and capability as virtualization and service-oriented forms, and then constructs the cloud service pool of simulation resource and capability so as to implement unified, centralized management and operation, which will support users to access the services of simulation resource and capability on demand at any time through the network for their various activities during the life cycle of simulation. The paper (Bo Hu Li. *et al*. 2009) has fully introduced the technical content, application mode, the architecture and key technologies of the cloud simulation platform.

There are kinds of resources on cloud simulation platform, including physical machine resource (CPU, memory, storage), virtual machine resource, software resource, licensing resource, model resource, equipment resource and capability resource. The distribute resources are accessed through virtualization middleware, service-oriented middleware and sensing middleware, and appeared as virtual resources, service resources and physical resources to the upper layer, shown in the Figure 1.

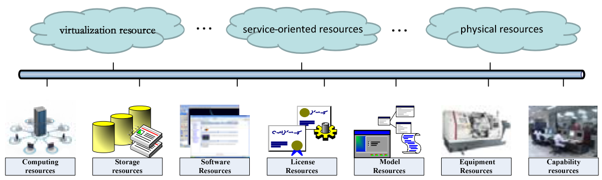

Figure 1: Resources in Cloud Simulation Platform

In terms of virtual resource, both Xen (Barham *et al*. 2003) and VMware (Anderson *et al*. 2005) are popular technology. The virtual machine cluster is managed through VMMs on each nodes and management center on the master node. In terms of service-oriented resources, Apache axis2[4] is used as service container on each node to deploy and monitor the services, and Platform ego[5] is used to manage the services of entire cluster. In terms of physical resources, the performance of physical cluster are monitored by Ganglia[6], and the management of them are implemented by distribute agents.

However, the diversity of resources on the cloud simulation platform makes it difficult to aggregate resources on demand. We need to integrate all types of resource management systems, and take the resources' collaboration into account, to achieve the optimal selection and combination of various resources. Cloud simulation platform uses resource management middleware to achieve the unified management of various resources, and uses dynamic construction modular to achieve the dynamic combination of various resources, so as to solve the problem effectively.

Based on the analysis of related work in Section II, we propose the guideline and the architecture of the resource management in cloud simulation platform in Section III. The resource management of cloud simulation platform involves several key technologies, including unified modeling techniques, selection techniques, combination technology, and operation/running technology, fault-tolerant technology, evaluation technology and so on. This paper focuses on the modeling technology and selection technology. In section IV, we present the unified formal description model of the simulation resource and resource instance from the perspectives of operation and running. In section V, we present the selection technology based on the resource group. Then, we introduce an application of the resource management in the field of multi-disciplinary virtual prototype. Finally, in section VII, we give a summary and future outlook.

2. RELATIVE WORK

Since 1983, the development of distributed simulation technology has experienced four periods which are Simulator Networking (SIMNET), Distributed Interactive Simulation (DIS), Aggregate Level Simulation Protocol (ALSP) and High Level Architecture (HLA) (Dahmann *et al.* 1998). HLA, whose latest progress is the HLA-Evolved[8-10], is a popular technology of distributed simulation. It can provide a flexible general-purpose simulation framework for the M&S of complex systems, and can improve the Interoperability and reusability of the simulation model and simulation system. Therefore, it has been widely applied in M&S.

After making good solution of the Interoperability and reusability on the level of simulation model such as components, federates and services, in order to manage and use all kinds of simulation resources better, Grid technology (Foster 2002) is introduced into simulation, which is so called Simulation Grid (Bo Hu Li *et al.* 2006). Simulation Grid, such as Cosim Grid (Bo Hu Li *et al.* 2006), NessGrid (Pearlman *et a*l. 2004) and FederationX Grid (Einstain 2005), is a new generation of M&S support system, combined with the new network technologies such as Internet technology, Web Service technology and Grid technology and the traditional M&S support technologies such as HLA technology. In the simulation grid, the simulation models are packaged as loose coupling simulation services with interoperability deployed and shared on the grid node, and are dynamically discovered and invoked in the runtime. As a result, the traditional pattern of chimney-like development and resource utilization are broken, and establish new resource management and utilization patterns of autonomous, dynamic sharing and on-demand collaborative.

However, from the view of application, current simulation grid services are almost fixed on some grid nodes, rather than created resource copy flexibly. In addition, current simulation grid services cannot penetrate the underlying hardware facilities so as to share fine-grained resources, including CPU, memory, software, etc, and it is difficult to satisfy multi-users to access all kinds of M&S services through the Internet anytime, anywhere. Again, the heterogeneous and loose environment of simulation grid cannot provide the security and high availability of operating environment for the upper applications.

With the rapid development of virtualization technologies recently, we can shield the differences of the hardware architecture, operating environment and other simulation resource, and uncouple the coupling relationship between them, making the construction and operation of simulation systems more flexible; we can provide a unified encapsulating standard for kinds of the heterogeneous simulation resources, making share and transparent use of simulation resources as much as possible; we can migrate and expand simulation operating environment dynamically and efficiently, so as to enhance the reliability and stability of the simulation runtime environment. After merging the latest technology of information field, such as virtualization technology, into the simulation grid, we have developed Cloud Simulation, which makes a better solution to the shortcomings of the simulation grid.

It is known that the cloud simulation is the further development of the simulation grid, therefore, some technology of resource management in the simulation grid can be learned and continue to be adopted. Chang Feng Song, *et al.* (2009) proposes a resource selection model and algorithm in the environment of the simulation grid. For the "All-to-All" task model that any federate in the distribute interactive simulation possibly interacts with any other federate, the paper aims to select the grid node with CPU frequency as large as possible, communication delay as small as possible, communication speed as quick as possible from the grid nodes contained the required grid services. The work presents a Resource Selection Model (RSM) in matrix, and then uses various intelligent optimization algorithms to select a set of optimal resource instances from eligible simulation resources.

However, since there are subtle changes in the resource form, the resource management of cloud simulation platform needs to fine-tune. For example, in the past, the task was to select M optimal nodes from N candidate nodes contained the required resource services (N>M), however, one node can use virtualization technology to build multiple copies of resource services at present. In the past, the required resource services were just fixed on N (finite) candidate nodes, however, the number of candidate nodes now can be much larger than N, since the introduction of virtualization technology which can build virtual nodes dynamically. These differences are related to the adjustment of the unified modeling technology and the selection technology for the simulation resources, which is the focus of this paper.

3. GUIDELINE AND ARCHITECTURE OF RESOURCE MANAGEMENT

Before discussion the simulation resource management technology about unified modeling and selection of resources in detail, we need first to introduce the guideline and the architecture of cloud simulation platform.

The guideline of cloud simulation platform is shown as follow:

- Centralized management of distributed resources

Fully integration of networked and shared computing resources, storage resources, software resources, License resources, knowledge / model resources, equipment resources and capability resources conducts unified management;

- Supplying services of unifying managed resources to distributed users

The aim of system is to organize simulation resources and capabilities quickly and flexibly, so as to provide services transparently on demand at anytime and anywhere.

- Effective collaboration of resources on the life cycle of M&S

The resource management covers the whole life cycle of M&S to support four application patterns, which include multi-users complete the design and analysis tasks independently, multi-users complete the simulation tasks collaboratively, multi-users complete the muti-analysis process collaboratively and multi-users access simulation capability, which is the combination of the intellectual resource and the traditional simulation resource, on demand. This paper does not take the fourth pattern into consideration.

The architecture of cloud simulation platform is shown in the Figure2.

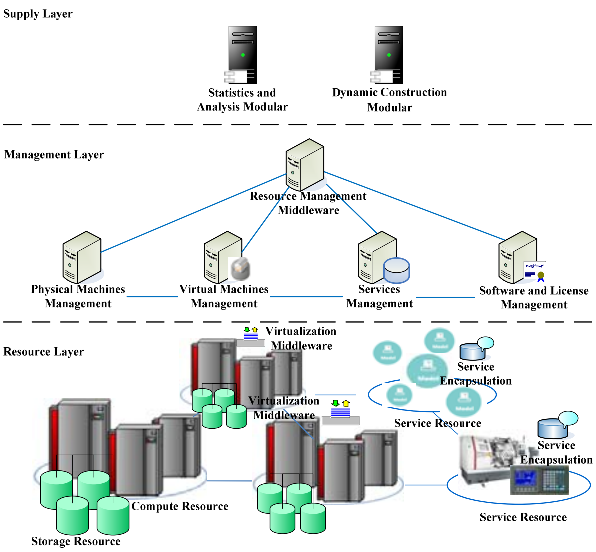

Figure 2: Architecture of Cloud Simulation Platform

There are three layers in the architecture, which are Supply Layer, Management Layer and Resource Layer.

Resource layer includes distributed computing resources, storage resources, service resources, which including service-oriented simulation software, simulation models, simulation equipments, and their license resources. Parts of the cluster is deployed virtualization middleware and become virtual computing resources so as to build virtual computing nodes on demand, other parts of the cluster is deployed monitor agents and so called physical computing resources.

Management layer includes physical machine management, virtual machine management, service management, software and their License Management. They will be integrated by resource management middleware, which is deployed on a dedicated server, so as to manage resource uniformly. With the expansion of the resources' scale and terrain, the deployment of the resource management middleware will be distribute, however the pattern of management and allocation of the resources keeps still.

Supply layer includes statistics and analysis modular and dynamic construction modular, which both deploy on the dedicated servers respectively. Dynamic construction modular prepares hardware, software and service environment ready for the upper application, and statistics and analysis modular provides the reporting service about resources for the users.

From the architecture of cloud simulation platform mentioned above, we can summarize that the resource management of cloud simulation platform covers physical machine resource management, virtual machine resource management, service resource management, software resource management and license management, shown in the Figure 3. In essence, license resource management subordinates to software resource management.

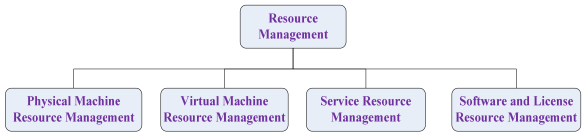

Figure 3: Types of Resource Management

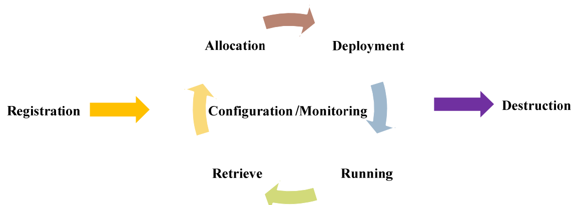

Figure 4: Life Cycle of Resource Management

In order to achieve the centralized management of distributed resources, as well as supply services of unifying managed resources for distributed users, resource management of cloud simulation platform covers the entire life cycle of resource, includes the unified registration, flexible configuration, real-time monitoring, on-demand allocation, efficient deployment, transparent running, timely retrieve and safe destruction for all kinds of simulation resources, shown in the Figure 4.

4. UNIFIED MODELING OF SIMULATION RESOURCES FOR OPERATION AND RUNNING

To support the unified management for the life cycle of various simulation resources, we first need to model all kinds of simulation resources on the cloud simulation platform uniformly. Generally, there are two perspectives of unified modeling, the one is resource operation, and the other is resource running. The former concerns about the resource's grouping, ownership, availability and state of allocation; the later describes the static configuration, dynamic performance and running state of the whole life cycle for kinds of simulation resources.

Figure 5 shows the formal description model of simulation resource and resource instance, as well as the relationships between them, from the perspective of resource operation. The resource attributes include resource ID, resource name, resource type and set of resource instances. Resource type can be divided into physical machine resource (identified as HPC), virtual machine resource (identified as VM), service resource (identified as SVR), and software resource (identified as SFW), from the analysis mentioned above. The set of resource instances achieve the association between resource and resource instances. A simulation resource includes a number of resource instances, and can be reverse indexed by any resource instance.

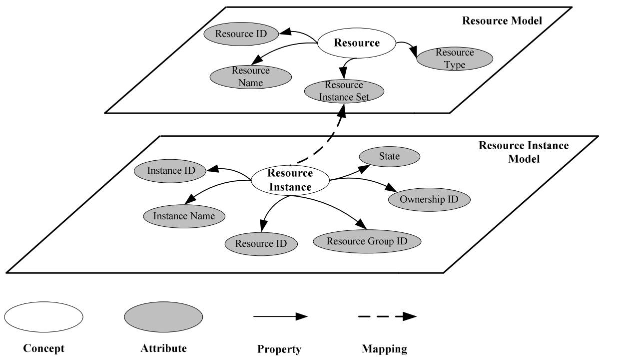

Figure 5: Formal Description Model from the perspective of Resource Operation

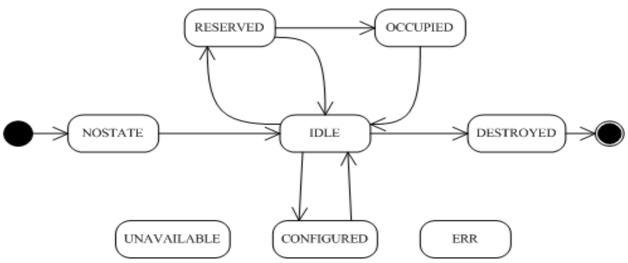

Figure 6: State Transition between Resource Instances

In addition, the resource instance attributes include instance ID, instance name, resource ID, resource group ID, ownership ID and state of allocation. Resource group ID is an important basis for the allocation of resources. Resource instances contacted with each other closely and executed collaboratively will be assigned to the same resource group. Ownership ID is an important symbol described the occupancy status and availability of the resource instance. From the perspective of resource operation, the state of allocation is an important property of the dynamic management for the life cycle of resource instances, including NOSTATE, IDLE, CONFIGURED, RESERVED, OCCUPIED, UNAVAILABLE, ERROR and DESTROYED. The conversion between the various states is shown in Figure6.

Figure 7 shows the formal description model of simulation resource and resource instance, as well as the relationships between them, from the perspective of resource running. The resource attributes include basic information (such as resource ID), the character information (such as operating system type), resource type and set of resource instances. The resource instance attributes include static configuration (such as the number of CPU), dynamic performance (such as CPU utilization) and running state of the whole life cycle. Similarly, the set of resource instances achieves the association between resource and its instances. A simulation resource includes a number of resource instances, and can be reverse indexed by any resource instance.

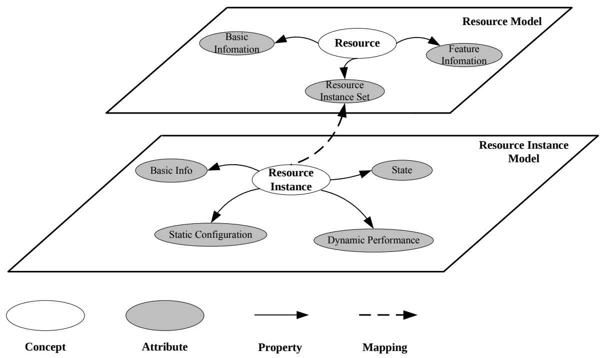

Figure 7: Formal Description Model from the perspective of Resource Running

In fact, there are subtle differences for different types of simulation resources and resource instances in their formal description models, from the perspective of resource running. There are detailed definitions of resource and resource instance for the physical machine, virtual machine, service and software respectively, shown as follow.

Definition I: Physical Machine Resource and Resource Instance

Physical Machine Resource
< Basic Information <Resource ID, Resource Name>,
Character Information <Is Cluster, OS Type> >
Physical Machine Node
< Basic Information <Node ID, Node Name, Resource ID>,
Static Configuration <Node IP, CPU Num, CPU Speed, Memory Size>,
Dynamic Performance <CPU Utilization, Memory Utilization>,
State {HPC_NOSTATE, HPC_INI, HPC_RUNNING, HPC_SHUTOFF, HPC_ERR}>

Definition II: Virtual Machine Resource and Resource Instance

Virtual Machine Resource

< *Basic Information* <*Resource ID, Template Name*>,
Character Information <*OS Type, Software List*> >
Virtual Machine Node
< *Basic Information* <*Node ID, Node Name, Resource ID*>,
Static Configuration <*Node IP, CPU Num, Memory Size, Host ID*>,
Dynamic Performance <*CPU Utilization, Memory Utilization*>,
State {VM_NOSTATE, VM_INI, VM_RUNNING, VM_PAUSE, VM_BLOCKED, VM_SHUTOFF, VM_ERR }>

Definition III: Service Resource and Resource Instance

Service Resource
< *Basic Information* < *Resource ID, Resource Name* >,
Character Information < *Is Parallel, OS Type* > >
Serviece Instance
< *Basic Information* <*Instance ID, Instance Name, Resource ID* >,
Static Configuration < *UDDI, CPU Num, Memory Size, Host ID* >,
Dynamic Performance < >,
State {SVR_NOSTATE, SVR_INI, SVR_IDLE, SVR_RUNNING, SVR_PAUSE, SVR_SHUTOFF, SVR_ERR}>

Definition IV: Software Resource and Resource Instance

Software Resource
< *Basic Information* < *Resource ID, Resource Name* >,
Character Information < *Is Parallel, OS Type* > >
Software Instance
< *Basic Information* <*Instance ID, Instance Name, Resource ID* >,
Static Configuration < *UDDI, CPU Num, Memory Size, Host ID (List)*>,
Dynamic Performance < >,
State {SFW_NOSTATE, SFW_INI, SFW_RUNNING, SFW_PAUSE, SFW_SHUTOFF. SFW_ERR }>

5. RESOURCE SELECTION TECHNOLOGY BASED ON RESOURCE GROUP

In the unified modeling of simulation resources mentioned above, we introduce the concept of resource group. Administrators can classify the simulation resource instances closely as a resource group. For example, a virtual machine template was deployed in a high performance cluster with shared storage, and then a virtual machine can be start up at any node of the cluster according to the mission requirement. Then the high performance cluster with the virtual machine template can be classified as a resource group. As another example, considering the security, the providers of some simulation models or services usually do not allow them migrating and being deployed free on the simulation platform, so as to isolate and publish them as the form of SOA. Then any of the simulation models or services can be classified as a separate resource group to reflect its isolated feature.

In general, there is high communication efficiency in a resource group, such as the high performance cluster inside which the point to point communication is based on infiniband, so that the communication performance can almost always meet with the requirement of the collaborative simulation. Therefore, for the resource selection in a resource group, which usually refers to select suitable physical machines to deploy simulation models or startup virtual machines, we focus on considering the performance of each node from the perspective of CPU num and memory size. In the resource group, the algorithm of selecting and allocating the resources for collaborative simulation was shown as follow with the form of pseudo-code.

```
Variable Declaration
  rCPUNum: the requirement of the cpu num, which is an integer
  rMEMSize: the requirement of the memory size, which is an integer
  rIndex: the requirement of the computing performance, which is a real number
  pCPUNum，pCPUSpeed, pCPUUtil: the cpu number, its speed and its utilization in the physical compute node
  pMEMSize，pMEMUtil: the memory size and its utilization in the physical compute node
  pCPUIndex: the available performance of the cpu, which is a real number
  pMEMIndex: the available performance of the memory, which is a real number
  pIndex: the available performance of the computing, which is a real number
Selection Algorithm
  For i = 1 to M //M is the requirement number of the computer node
    rIndex[i] = α1 * rCPUNum[i] +α2 * rMEMSize[i] //α1 and α2 are the weighting factors
  EndFor
  Rearrange the array rIndex in descending
  For j = 1 to N //N is the candidate number of the computer node
    pCPUIndex[j] = pCPUNum[j] * pCPUSpeed[j] * (1 – pCPUUtil[j])
    pMEMIndex[j] = pMEMSize[j] * (1 – pMEMUtil[j])
  End For
  For i = 1 to M
    For j = 1 to N
      If (rCPUNum[i] * pCPUSpeed[j] < pCPUIndex[j])
        Then pIndex[j] = β1 * pCPUIndex[j] //β1 is the weighting factor
```

```
ELSE pIndex[j] = β1 * pCPUIndex[j] * δ1 //δ1 is the punishment factor
    End If
    If (rMEMNum[i] < pMEMIndex[j])
      Then pIndex[j] = pIndex[j] + β2 * pMEMIndex[j] //β2 is the weighting factor
      ELSE pIndex[j] = pIndex[j] + β2 * pMEMIndex[j] * δ2 //δ2 is the punishment factor
    End If
  End For
  Select the node j, the pIndex of which is max, as the target node for the requirement i
  pCPUIndex[j] = pCPUIndex[j] - rCPUNum[i] * pCPUSpeed[j]
  pMEMIndex[j] = pMEMIndex[j] - rMEMNum[i]
End For
```

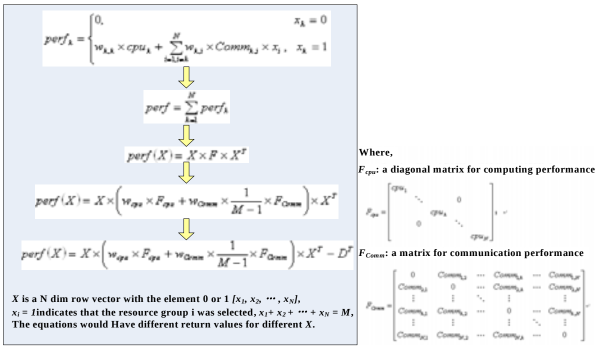

Figure 8: Resource Selection Model for Selecting Resource Groups

In the cloud simulation platform, every user corresponds to a prior resource group for utilization according to his identity, which is configured by the administrators. When the amount or performance of specific resource in the resource group can not satisfy the requirement of collaborative simulation task for a user, cloud simulation platform will find the specific resource in other resource groups. However, the cloud simulation platform has the feature of large-scale virtualization, the communication between resource groups may be on the WAN or on the Internet. At this situation, communication efficiency should be considered as an important indicator. If there are suitable resource instances in more than one resource groups to meet with the remaining simulation requirement, which cannot be satisfy in the prior resource group, both computing performance and communication performance need to been considered. The RSM (Chang Feng Song, et al. 2009) in simulation grid mentioned above may be learned and continue to be adopted to select the resource group, shown in the Figure 8. What we need to do is just setting the weights of communication performance between the resource groups instead of the grid nodes. After selecting a resource group, we can use the above method to select specific resource instances in the resource group for the simulation task.

6. APPLICATION

The resource management technologies of cloud simulation platform described in this paper have played an important role in the field of multi-disciplinary virtual prototype engineering in their preliminary application. Based on the cloud simulation platform COSIM-CSP (Bo Hu Li *et al.* 2009), the application about collaborative design and simulation of the landing gear system of an aviation aircraft is shown as follow.

The multi-disciplinary virtual prototype of the landing gear system consists of several sub-system models, such as electronic control model, multi-body dynamics model and hydraulics model and so on. These models refer to various commercial software, such as control system design/simulation tools MATLAB / SIMULINK, dynamics system design/simulation tools MSC ADAMS, hydraulics system design/simulation tools MSC EASY5 and structure design tool CATIA. The top-level system model of the virtual prototype constructed by the COSIM-CSP was shown in Figure9.

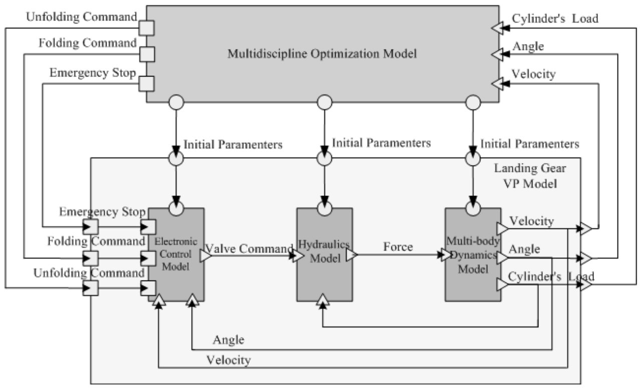

Figure 9: Top-level System Model of the Virtual Prototype Constructed by the COSIM-CSP

According to the formal description of unified modeling mentioned above, we can describe the requirements of various sub-system models of the virtual prototype as follow.

```xml
<?xml version="1.0" encoding="UTF-8" ?>
- <requirement>
  - <vmNode>
      <modelName> Electronic Control
          Model</modelName>
      <osType>Windows XP</osType>
    - <softwareList>
        <softwareName>MATLAB/SI
            MULINK</softwareName
            >
      </softwareList>
      <cpuNum>1</cpuNum>
      <memSize>1</memSize>
    </vmNode>
  - <vmNode>
      <modelName>Hydraulics
```

```
            Model</modelName>
        <osType>CentOS5.4</osType>
      - <softwareList>
            <softwareName>MSC
              EASY5</softwareName>
        </softwareList>
        <cpuNum>1</cpuNum>
        <memSize>1</memSize>
    </vmNode>
  - <vmNode>
        <modelName>Multi-body Dynamics
            Model</modelName>
        <osType>Windows XP</osType>
      - <softwareList>
            <softwareName>MSC
              ADAMS</softwareName>
        </softwareList>
        <cpuNum>1</cpuNum>
        <memSize>1</memSize>
    </vmNode>
  - <vmNode>
        <modelName>Structure
            Model</modelName>
        <osType>Windows XP</osType>
      - <softwareList>
            <softwareName>CATIA</softw
              areName>
        </softwareList>
        <cpuNum>2</cpuNum>
        <memSize>2</memSize>
    </vmNode>
</requirement>
```


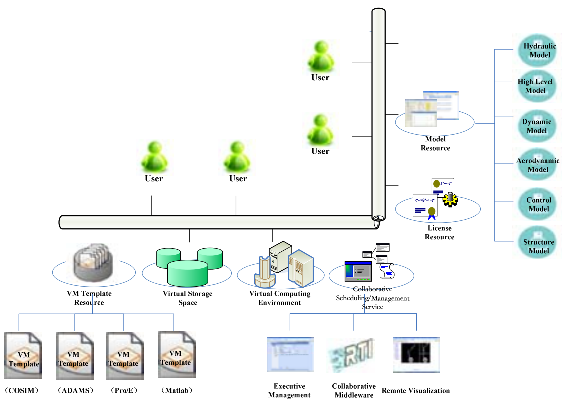

Figure 10: Virtual Resource Environment

In the COSIM-CSP, the overall system designer and the sub-system designers can see the virtual resource environment shown in the Figure 10, including virtual computing environment, virtual storage space, collaborative scheduling / management service, virtual machine template resource, licensing resource and model resource. Specifically, virtual machine template resource includes templates installed COSIM top-level modeling software, MATLAB / SIMULINK software, MSC ADAMS software, MSC EASY5 software, CATIA software respectively; cooperative scheduling / management service includes execution management tool, collaboration middleware, remote visualization service; model resource includes top-level model, the electronic control model, multi-body dynamics model, the hydraulics model, structure model.

In the process of implementation, the overall system designer and the sub-system designers complete the top-level system modeling and sub-system design respectively in their own virtual desktops, which are the research environment customed, and then upload the corresponding model files through the application portal of COSIM-CSP. Based on the resource management technologies mentioned above, COSIM-CSP constructs the running environment of the collaborative simulation dynamically, and builds the system-level virtual prototype automatically, so that achieving the aggregation and collaboration of the simulation resources. The process is shown in the Figure 11.

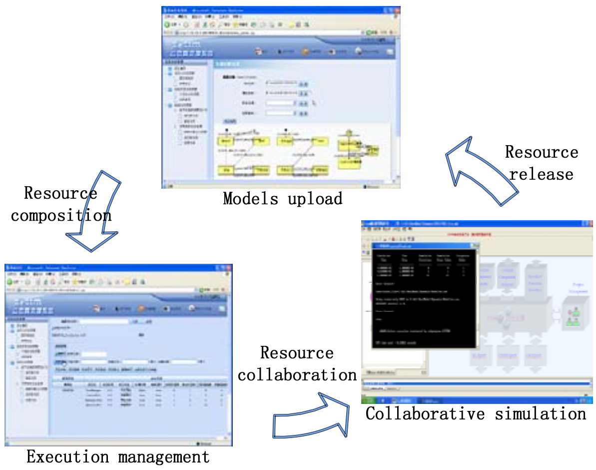

Figure 11: The Process of the Implementation

7. CONCLUSION AND FUTURE WORK

The research of this paper proposes the guideline and the architecture for the resource management of the cloud simulation platform. In this framework, in order to support the centralized management for the life cycle of all kinds of simulation resources, we present the formal description of the unified modeling from the perspectives of resource operation and resource running. In the unified modeling, we propose clearly the concept of resource group, so as to distinguish the selection methods in the resource group and inter-groups. The resource management technologies of cloud simulation platform described in this paper have been verified in the collaborative design and simulation process for the multi-disciplinary virtual prototype.

In future work, we will do more research on simulation performance evaluation, and enhancing the flexibility of resource selection, which will select the most appropriate resource instances instead of best performance ones. In addition, we need further research on the on-demand aggregation and high efficient collaboration for the simulation capabilities.

ACKNOWLEDGMENTS

This paper is supported by the National 973 plan (No. 2007CB310900).

REFERENCES

LI Bo-hu, CHAI Xu-dong, HOU Bao-cun, et al., 2009. A Networked Modeling & Simulation Platform Based on the Concept of Cloud Computing – "Cloud Simulation Platform". *Journal of System Simulation* 21(17): 5292-5299 . (in Chinese)

Barham, P., et al., 2003. Xen and the art of virtualization. *Proceedings of the 19th ACM Symposium on Operating Systems Principles,* 164-177. Oct. 2003, New York USA.

Anderson, T., et al., 2005. Overcoming the Internet impasse through virtualization. *IEEE Computer* 38 (4): 34-41.

Apache, 2009. *Axis2/Java - Next Generation Web Services.* Available from: http://ws.apache.org/axis2/[2010].

Platform, 2007. *Platform EGO white papers.* Available from: http://platform.com/Products/platform-enterprise- grid-orchestrator/whitepapers[2010].

Ganglia, 2010. *Ganglia Monitoring System.* Available from: http://ganglia.sourceforge.net/[2010].

Dahmann, J., et al., 1998. The DoD High Level Architecture: An Update. *Proceedings of the 1998 Winter Simulation Conference,* 797-804. 10 Dec. 1998, Washington, D. C., USA

IEEE, 2008. *Draft Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)-Object Model Template (OMT) Specification.* Available from: http://ieeexplore.ieee.org/servlet/opac?punumber=4478265 [2008].

IEEE, 2009. *Draft Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)-Framework and Rules.* Available from: http://ieeexplore.ieee.org/servlet/opac?punumber=5347324 [2009].

IEEE, 2009. *Draft Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)-Federate Interface Specification.* Available from: http://ieeexplore.ieee.org/servlet/opac?punumber=5347330 [2009].

Foster, I., 2002. The Grid: A New Infrastructure for 21st Century Sciece. *Physical Today* 55 (2): 42-47.

Bo Hu Li, Xudong Chai, Baocun Hou, et al., 2006. Research and Application on CoSim (Collaborative Simulation) Grid. *Proceedings of MS-MTSA2006,* 156-163. July 2006, Alberta, Canada.

Pearlman, L., et al. 2004. Distributed Hybrid Earthquake Engineering Experiments: Experiences with a Ground-Shaking Grid Application. *Proceedings of 13th IEEE Symposium on High Performance Distributed Computing,* 14-23. June 2004, Hawaii, USA.

Einstein, A. 2005. FederationX: A Technical Brief. Available from: http://www.magnetargames.com/.

Chang Feng Song, Bo Hu Li and Xudong Chai, 2009. Node selection in simulation grid. *Journal of Beijing University of Aeronautics and Astronautics* 35(1): 56-60. (in Chinese)

AUTHORS BIOGRAPHY

Ting Yu Lin was born in 1984. He received his B.S.degree in BeiHang University. He is currently a Ph.D. candidate at the School of Automatic Science and Electrical Engineering, BeiHang University, Beijing, China. His research interests include multi-disciplinary virtual prototype and intelligent distributed simulation.

Xudong Chai was born in 1969. He is a researcher and deputy director at Beijing Simulation Center of Second Academy of Aerospace Science & Industry Co. and council members of Chinese System Simulation Association and National Standardization Technical Committee. His research interests include automatic control and simulation.

Bo Hu Li was born in 1938. He is a professor at School of Automatic Science and Electrical Engineering, BeiHang University, and Chinese Academy of Engineering, and the chief editor of "Int. J. Modeling, Simulation, and Scientific Computing". His research interests include multi-disciplinary virtual prototype, intelligent distributed simulation and cloud manufacturing.