Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/080643

Development of Simulation Task Unit Model based on Role and State Mode

Zhiteng Wang[1,*], *Hongjun Zhang*[1], *Rui Zhang*[1], *Yong Li*[2] *and Youliang Zhang*[1]

[1] PLA University of Science &Technology, Nanjing 210007, China
[2] Nanjing Artillery Academy, Nanjing 210110, China

Received: 13 Nov. 2013, Revised: 11 Feb. 2014, Accepted: 12 Feb. 2014
Published online: 1 Nov. 2014

Abstract: Currently, service-oriented model and simulation has become a main development direction for modeling and simulation. In order to separate the simulation service and state and effectively reuse simulation resources, this paper proposed a simulation task unit (STU) model method based on role and state mode, in which role could call the simulation service by simulation member and the state value of simulation service could be saved in the state of role. This paper also analyzed the relations among role, state and logic, and described the simulation member model and simulation task model in detail. Besides, an STU model tool was developed for effectively modeling for STU, and this tool was proved to be useful for saving simulation service state in service-oriented model and simulation.

Keywords: Role,State, Plot, STU, Model, Simulation

1 Introduction

Currently, service oriented architecture (SOA) has been paid more and more attention by many big computer companies, such as BEA [1], IBM [2], HP [3], Microsoft [4], CISCO [5]Oracle [6], SAP [8], Sun microsystems [9], and DOD [10]. Service-oriented concepts and distributed technologies (grid and cloud computing) are now widely applied in the field of engineering and non-engineering, which boosts the sharing and reuse of all kinds of simulation resources, and enhances the interoperability in collaborative, dynamic optimal schedule. How to apply these concepts to build new type of distributed simulation model and simulation system is becoming an important and hot topic for researchers in the modeling and simulation (M&S) field.

In recent years, a significant change is taking place in the M&S field, that is, more and more service-oriented simulation frameworks have been established with technologies and ideas of web service, SOA, cloud computing and grid computing, such as XMSF [11] [12], Cosim-Grid [13] and GridSim [14], and many other research [18] [19] [20] [21], etc. Simulation resource packaged for simulation service is the fundamental instrument in the area of service-oriented M&S, and the transformation can be achieved through publication service and subscription service in the service bus. It is worth noting that web service has become the mainstream in service-oriented M&S. For example, in XMSF, web-based RTI (Runtime Infrastructure) can be achieved by web-enable technology.

There are many unique features in the service-oriented M&S on the background and technical connotation, and the service-oriented models in M&S architecture and enterprise are also different. On the one hand, the service-oriented model in enterprise is based on SOAP protocol for web service; while that in the field of M&S is based on web service of broad sense which requires only the use of WSDL to describe service, and allows message to be sent in any format. On the other hand, the service-oriented model in enterprise is more concerned on stateless service; while that in the field of M&S not only emphasizes the stateless service but also involves a large number of state services. Although this problem can be solved through saving service state by the way of WSRF, it is still limited to some specified frames. In order to achieve the purpose of saving stateless services in a SOA, this article tries to develop a simulation task unit (STU) model based on the role and state mode, which is much more flexible and not limited to specified frames. In addition, this paper also describes the simulation member model and simulation task model

* Corresponding author e-mail: wangzhiteng168@163.com

in detail, and analyzes the relations among role, state and logic. Finally, to effectively model for STU, an STU model tool is also established.

2 Simulation task model architecture

STU is defined as a simulation task unit that one or more roles in the simulation engine act as the base element to complete a simulation task according to the sequence of events in the simulation plot, and this model can be used to describe simulation task by element collections including role, simulation members, plot, events, the role of information, and the role state data.

Figure 1 shows the work architecture of STU model. It can be seen that simulation member is responsible for supplying simulation service for the role by binding simulation service in service bus. Most simulation service exists in the form of web service, but it is not suitable for supplying state service in the process of complicated interactive simulation application in the field of M&S. Hence, according to the simulation plot, the concept of role and simulation member is proposed for binding many roles to effectively reuse simulation resources. As shown in Figure 2, role 2 can be bound to the simulation members 1 and 2, and simulation task members can take part in simulation task by configuring its role ownership for the reason of using role mode. In order to achieve loose coupling relationship between simulation task and simulation member, simulation task member and simulation task should be modeled separately.

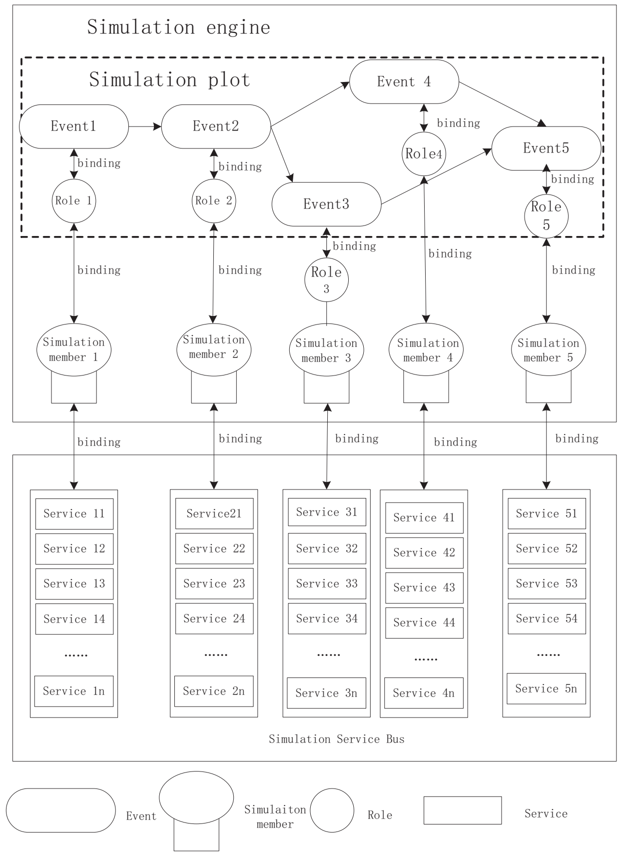

Fig. 1: STU work architecture

3 STU model description

The model of simulation task member and simulation task are described in detail in the following part.

3.1 Model description of simulation task member

Simulation task members may come from different simulation fields with different service descriptors or different particle size heterogeneous encapsulated service models, and these different structures will lead to many problems. Hence, it is very important for simulation members to work together to accomplish complex simulation task based on the reasonable effective model standard. In order to build a simulation task model with characteristics of friendliness, effectivity, integrality, reusability and other specific characteristics, simulation task member model should be a standard model independent of field and platform, and it should support the standard simulation model schema description and simulation development in the whole modeling process. Under the simulation task required, simulation task member (STM) can be described with four unit groups as follows: STM=DO,FS,MS,QOS

DO (Domain Ontology): it represents the domain ontology of the simulation member, and the ontology properties are added by experts according to their respective domain knowledge.

FS (Function Set): it represents simulation task member function sets. Simulation task member might be a combination of one or more services to accomplish one or more functions, and it can be described by certain function sets and then stored in a list of functions.

MS (Metadata Set): it represents the metadata collection of simulation task member. Simulation task member may have one or more properties, and the collection is used to describe it.

QOS (Quality of service): it represents the quality of service of simulation task member.

The element of simulation task member model can be designed as Figure 3 shows.

DO includes many elements as follows: (1) Description: it is the text description for simulation task member function. (2) Application domain: it describes the application domain of simulation task member. (3) Reuse

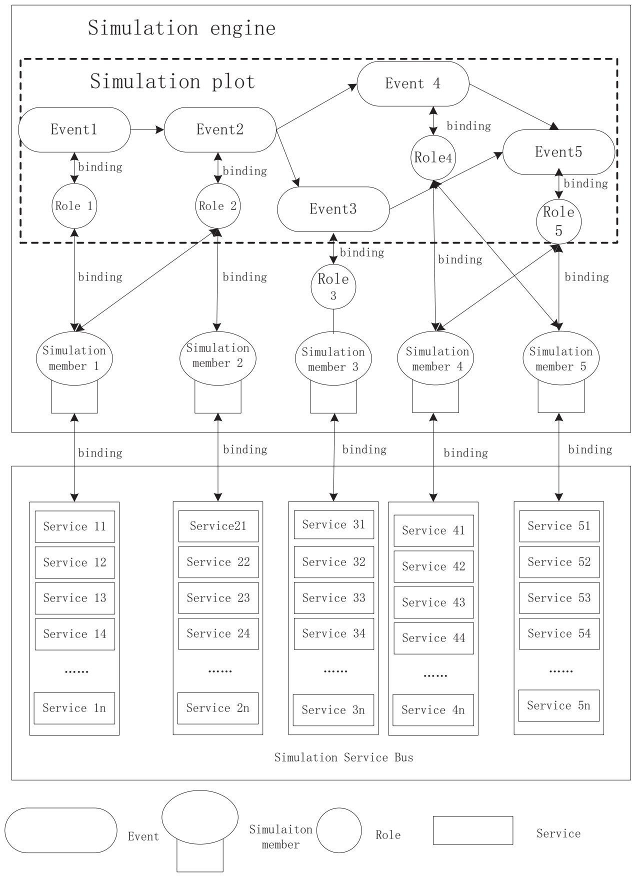

Fig. 2: The relation between role and simulation member

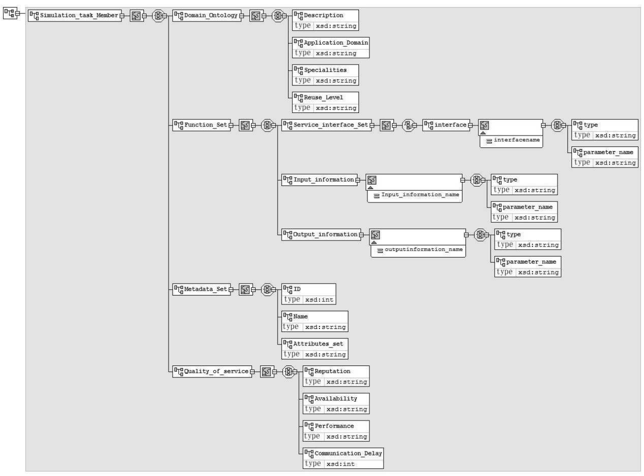

Fig. 3: The element of simulation task member

Level: it describes the reuse level of simulation task member. (4) Specialties: it describes the specialty of simulation task member.

FS includes many elements as follows: (1) Service interface set: it describes the service of interface in simulation task member. Simulation task member is composed of one or more simulation services, which contain more than one service interfaces that can be called. In order to call FS expediently, FS should includes relevant parameters. (2) Input information: it describes the input information parameters and types of simulation tasks member. (3) Output information: it describes the output information parameters and types of simulation tasks member.

MS includes many elements as follows: (1) ID: it describes the unique identifier of simulation task member. (2) Name: it describes the name of simulation task member. (3) Attribute set: it is a variable length vector of describing the attribute set of simulation task member such as space attribute, movement attribute, etc.

QOS includes many elements as follows: (1) Reputation: it describes the reputation of service in simulation task member, which represents the reliability degree of simulation task member to accomplish the task. (2) Availability: it describes the availability of simulation task member. (3) Performance: it describes the performance of simulation task member. (4) Communication delay: it describes the communication delay of simulation task member.

3.2 Simulation task model description

In order to accomplish special simulation task, simulation task unit is responsible for the coordination and management of simulation members. Hence, it is important to describe simulation task with standard description and unified modeling standard, which can describe simulation tasks, and the obligation and relation of different members clearly and accurately to make different simulation members with a concerted effort under a unified modeling standard. Task collaboration (TC) in STU can be constituted with six unit groups as follows: TC=T,G,R,M,P,C T (Task): it describes the task of simulation task unit. G (Group): it describes the group of simulation task unit, which is composed of many roles. R (Role): it describes the role of STU. STU can have many roles and each role is composed of many simulation members. M (Member): it describes the member of simulation task unit. P (Plot): it describes the plot of STU which is a unique factor for STU to arrange different complete events under assigned time and place. It can be designed as Figure 4 shows.

Task includes many elements as follows: (1) Name: it describes the task name of STU. (2) Description: it is simple description for the task in STU (3) Subtask: it is a collection of subtask in STU. Group is composed of one

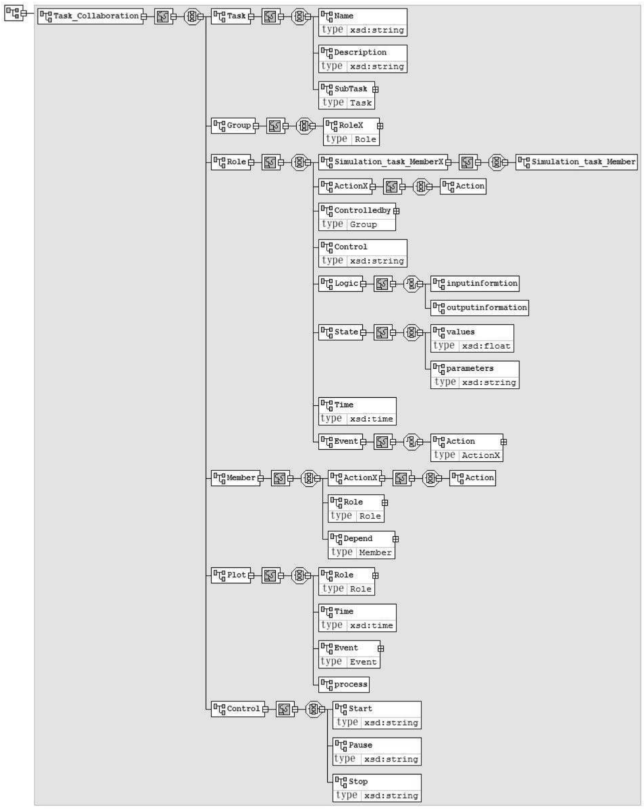

Fig. 4: The element of simulation task

or more roles, and it is significant to coordinate different roles to accomplish a task in simulation.

Role includes many elements as follows: (1) Member: role is composed of one or more simulation task members and each simulation task member can act as many roles. (2) Action: it describes the action of role according to service supplied by simulation member. (3) Controlledby: it describes the role and group to control role in process of simulation. (4) ControlUnit: it describes the role and group controlled by role in process of simulation. (5) Logic: in simulation task unit, logic represents the logical rule or law of role event which can be established by specialist in different fields according to role characteristic in reality to express the personality of the roles. The role state is changed according to the effect of other role states. In addition, this logic can be acquired by data mining from a large number of effective data collected from experiment or reality. (6) State: simulation task member exists in the stateless form, but it needs service state in the simulation. In order to solve this problem, this paper uses role state to separate service and state. The role state is got by the output information from simulation task member and then stored in the value of role state. (7) Time: the time of role in STU is controlled by simulation engine, and the role should complete special action in assigned time according to simulation plot. (8) Event: it describes the event of role including subscribing and publishing message.

Member includes many elements as follows: (1) Action: it describes the action of simulation task member according to many services. (2) Role: it describes the role of simulation task member to act as, and each simulation task member can act as many roles. (3) Depend: it represents the depending input information of simulation task member. Plot includes many elements as follows: (1) Role: it is an important element in plot, and many roles will take part in the process of simulation task. (2) Time: it is used to mark time for plot in the simulation task process, which is an important reference element for simulation task advance. (3) Event: simulation tasks run under the event-driven mode, and multiple roles may participate in the interactive simulation according to the events in simulation plot. (4) Process: it describes the relations between events of role, which include sequence relation, condition relation, parallel relation, circular relationship, AND relationship, OR relationships, logic decision, etc. Control is used to control the simulation process such as start, pause, stop, etc.

4 The relations among role, state and logic

Role simulating things in real-world necessarily involve the state of things, which can be described by parameters and their corresponding values with data representation. In fact, everything has its characteristic and law which can be expressed by the change of parameter values according to certain laws, and this characteristic and law can be called as logic. If not completely understanding the law, logic is just like a black box for inputting parameter data and outputting parameter data. Although it is very difficult to grasp the law in things due to less data available in most cases, the law of things can be gradually recognized and corrected according to continuous accumulation and analysis of new data.

Generally speaking, things should be a unity of description, data, and logic, and the process of understanding one thing for human is a process of data search, processing, mining, and the innovation. Based on these viewpoints, the logical relationship between them can be gotten, and logical relationship between different elements can finally be expressed by mathematical expressions. For example, if an object is described by threes parameters of Parameter 1, Parameter 2 and Parameter 3, the relations of them can be acquired by data mining in Figure 5.

From the perspective of service-oriented simulation, the roles can bind simulation members of the data acquisition and call acquisition and storage services to obtain and store related data values for the parameter. Then, data analysis services can be called for data mining to obtain the relationship among the different parameters. At last, simulation member for logical storage calls the

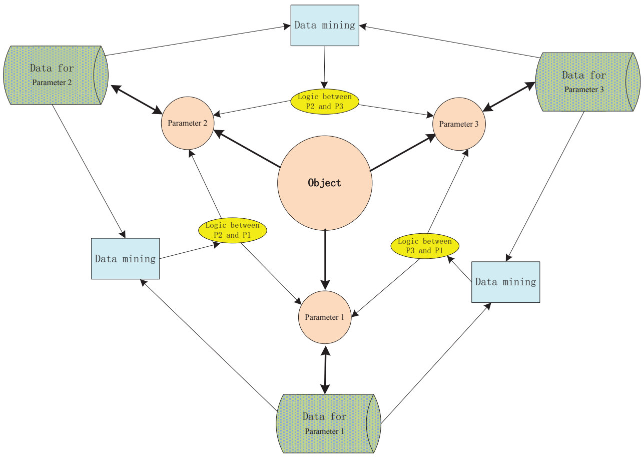

Fig. 5: The relation of parameter, data and logic

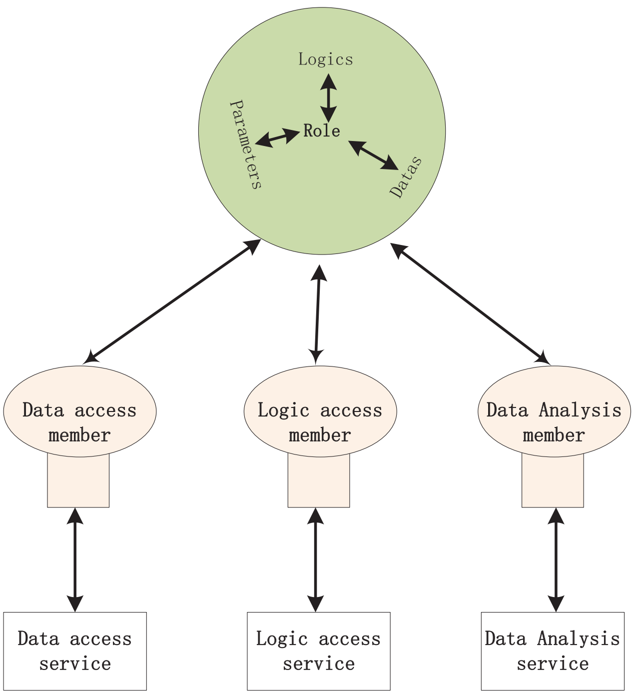

Fig. 6: Role state data and logic access

logical relevant storage service to store the logic. The whole process can be shown in Figure 6.

In fact, the simulation data and logic can interact with each other, which means that the relation between parameter and logic can be acquired by data mining and data can be produced by logical relation between different parameters in simulation. Continuous mutual authentication can achieve unification of data and logic,

```
<?xml version="1.0"?>
<xsd:schemaxmlns:xsd="http://www.w3.org/2001/XMLSchema">
<xsd:element name="Simulation_task_Member" type="Simulation_task_Member_type">
<xsd:complexType name="Simulation_task_Member_type">
<xsd:all>
<xsd:element name="Domain Ontology">
<xsd:complexType>
<xsd:all>
<xsd:element name="Description" type="string"/>
<xsd:element name=" Application Domain" type="string"/>
<xsd:element name="Specialities" type="string"/>
<xsd:element name="Reuse Level" type="string"/>
</xsd:all>
                    ......
<xsd:element name="Quality of service">
<xsd:complexType>
<xsd:all>
<xsd:element name="Reputation" type="string"/>
<xsd:element name="Availability" type="string"/>
<xsd:element name="Performance" type="string"/>
<xsd:element name="Communication Delay" type="int"/>
</xsd:all>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
 </xsd:schema>
```

Fig. 7: simulation task member model described by STMDL

and ultimately, descriptions, data and logical will become harmony and unity.

5 The model description language of STU

Simulation task model description language (STMDL) supplies the frame for simulation task description and simulation member description based on high abstract level, which is between scripting languages and object-oriented languages. STMDL is able to accurately model the frame in accordance with the standard model description for simulation task and simulation member, and deal with the relationships among simulation members, roles, tasks, etc. Its format is XML with good versatility, which can be edited according to different simulation applications. For example, simulation task member model can be described by STMDL with XML schema as Figure 8.

6 The model tool of STU

Mapping the XML schema to XML is important in the process of STU model, and there are many schema mapping tools such as Microsoft BizTalk Mapper [15],

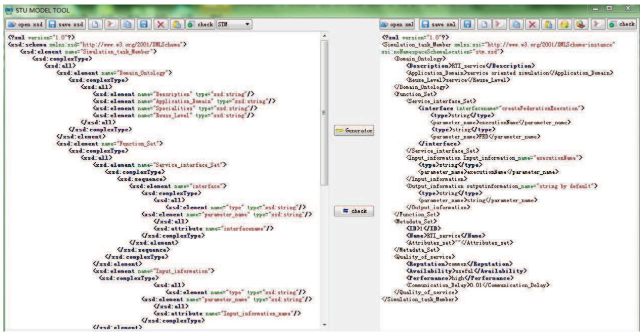

Fig. 8: STU model tool

Stylus Studio XML Mapping Tools [16], and SAP XI Mapping Editor. However, these tools cannot edit and check between XML schema and XML in time. In order to rapidly establish STU model, an STU model tool is developed, which can dynamically check and edit between XML schema and XML. Its function can be shown in Figure 7, and it has many excellent merits for STU model as follows: (1) It can create and edit model frame according to XML schema files and model description XML files. What's more, it can load and edit existed model files in format of XSD or XML. (2) Model frame in XSD format and model in XML format can be automatically mapped by specific mapping algorithm, which allows developers to conveniently edit or write XML comparison with model frame description according to constrain condition. (3) It can check the validity of XML schema or XML to help developers quickly rectify errors, and the consistency of the model framework description and model description can also be checked. This paper uses this tool to build model framework for simulation task member and simulation task unit, which automatically generates XML files for corresponding model. For example, if a run time infrastructure service is bound to simulation task member, the content of this simulation task member can be edited with the STU model tool as Figure 9 shows.

7 Conclusions

This paper described the composition mode of STU from the perspective of service-oriented modeling and simulation based on role state mode, which supplied a new method to save service state without considering frame. Under this model, role could be associated with simulation service by simulation member, and the state of simulation service could be saved in the state value of the role. Furthermore, this model ensured simulation services could be fully reused and that the efficiency of simulation services could be improved. Besides, this article also analyzed the relations between the role state and logic in simulation of task model, which provided a good idea for finding logical relationships between the various parameters and producing simulation parameter data in a service-oriented modeling and simulation. Finally, an STU model tool was established to build model framework for simulation task member and simulation task unit, which was useful for developers to quickly model for STU. In further study, the role state mode will be applied and improved in the simulation application.

Acknowledgement

This work was supported by National Natural Science Foundation of China (Grant no. 70791137).The authors are grateful to the anonymous referee for acareful checking of the details and for helpful comments that improved this paper.

References

[1] BEA, SOA Resource Center. http://www.bea.com/framework.jsp?CNT=index.htm&FP=/content/solutions/soa/.
[2] IBM Service Oriented Architecture. http://www-306.ibm.com/software/solutions/soa/.
[3] HP's approach to Service-Oriented Architecture. http://h71028.www7.hp.com/enterprise/cache/329749-0-0-225-121.html.
[4] Microsoft SOA.http://www.microsoft.com/soa/.
[5] CISCO Service-Oriented Network Architecture. http://www.cisco.com/en/US/netsol/ns629/networking_solutions_market_segment_solutions_home.html.
[6] Oracle Service-Oriented Architecture. http://www.oracle.com/technologies/soa/index.html.
[7] J.R Banavar, J. Damuth, A. Maritan, A. Rinaldo, Proceedings of the National Academy of Sciences of the United States of America, **99**, 10506-10509 (2002).
[8] SAP Enterprise SOA. http://www.sap.com/platform/esoa/index.epx.
[9] Sun Microsystems, Service-Oriented Architecture.http://www.sun.com/products/soa/index.jsp.
[10] R. Paul, DoD towards Software Services, in: Proceedings of 10th IEEE International Workshop on Object-oriented Real-time Dependable Systems(WORDS), February, 3-6 (2005).
[11] D. Brutzman, M. Zyda, M. Pullen, K.L. Morse, Extensible Modeling and Simulation Framework (XMSF): Challenges for Web-Based Modeling andSimulation: XMSF 2002 Findings and Recommendations Report: Technical Challenges Workshop and Strategic Opportunities Symposium, October (2002).
[12] XMSF SAIC Web-Enabled RTI, 2003. http://www.movesinstitute.org/xmsf/projects/WebRTI/ XmsfSaicWebEnabledRtiDecember2003.pdf(27.02.07)
[13] B.H. Li, X. Chai, Y. Di, H. Yu, Z. Du, X. Peng, Research on service oriented simulation grid, in: Proceedings of the Eighth International Symposium onAutonomous Decentralized Systems (ISADS 2005), April 4-8, 7-14 (2005).

[14] R. Buyya, A. Sulistio, Service and utility oriented distributed computing systems: challenges and opportunities for modeling and simulationcommunities, in: The 41st Annual Simulation Symposium (ANSS), Ottawa, April 14-16, 68-81 (2008).
[15] http://msdn.microsoft.com/en-us/library/ee253382(v=bts.10).aspx.
[16] http://www.stylusstudio.com/xml_schema.html.
[17] C. Bianca and L. Fermo, Computers & Mathematics with Applications, **61**, 277-288 (2011).
[18] L Zhang, B Zhang, The quotient space theory of problem solving, Fundamenta Informaticae, IOS Press, **59**, 287-298 (2004).
[19] Heng He, Ruixuan Li, Xinhua Dong, Zhi Zhang, Hongmu Han.An Efficient and Secure Cloud-Based Distributed Simulation System. Applied Mathematics & Information Sciences, **6**, 729-736 (2012)
[20] A Amstutz, D Carr, M Kelleher, L Miele, M Neff, Monetaire wealth management platform, Google Patents, (2002).
[21] SJ Fink, SB Baden, SR Kohn, Efficient run-time support for irregular block-structured applications, Journal of Parallel and Distributed Computing, Elsevier, **50**, 61-82 (1998).

Zhiteng Wang was born in Heilongjiang province in 1982, china.He received the masters degree in Nanjing artillery academy in 2010. Currently,He is a doctorial student in Technology Military Simulation Technology and data engineering Lab in PLA University of Science. He research on service-oriented model and simulation.He has joined in many scientific research items on model and simulation.

Hongjun Zhang is a chief professor and doctoral supervisor at PLA University of Science and Technology. His researches areas are military simulation, model and military operational research,and system integration.

Rui Zhang received Ph. D. oncommunication engineering at school of PLA University of Science and Technology.He is currently an adjunct professor in PLA University of Science and Technology. His researches area is simulation engineering

Yong Li is a professor and doctoral supervisor at Nanjing Artillery Academy. His researches areas are science of tactics and military education and training.

Youliang Zhang was born in Hebei province in 1986, china.He received the master's degree in Nanjing PLA University in 2011. Currently,He is a doctorial student in Technology Military Simulation Technology and data engineering Lab in PLA University of Science. He research on service-oriented model and simulation.He has joined in many scientific research items on model and simulation.