Design of Resource Repository in Virtual Test Architecture

Lian-Lei Lin
Department of Automatic Test and Control, Harbin Institute of Technology, Harbin, China
Email: linlianlei@hit.edu.cn

Yue Ma and Ling-Yu Li
Department of Automatic Test and Control, Harbin Institute of Technology, Harbin, China
Email: lilingyu@hit.edu.cn

***Abstract*— In reference to other architectures such as HLA(high level architecture) and TENA (test and training enabling architecture), etc., combined with new demands in the test and training field, we propose a new virtual test architecture HIT-TENA (Harbin Institute of Technology-TENA). This paper expounds the design of HIT-TENA resource repository in detail. It uses a four-tier architecture, which is mainly composed by the underlying distributed database, repository manager and Web Service. After the abstract analysis of different types of resources, we built the distributed database and developed the repository manager based on B/S (browser/server) for the users to manage the resources. In addition, we prepare database access interferences for other parts of the HIT-TENA architecture. At present, the resource repository has been running in the HIT-TENA, to good effect.**

***Index Terms*— HIT-TENA, resource repository, database, B/S, Web Service**

I. INTRODUCTION

As a new kind of low-cost, high efficiency and short cycle test method, virtual test is widely used in military industry, and gradually has equally importance with physical test. In terms of technical background, the virtual test architecture is a kernel throughout the virtual test verification techniques developing life cycle [1].

High Level Architecture (HLA) is a part of modeling and simulation public technology framework advocated by the Department of Defense of USA's M&S Master Plan [2]. HLA does not specify any particular application to meet its needs, it is defined in a neutral fashion, and thus applies to the development and integration of all areas (such as analysis, test, training, etc.) in modeling and simulation.

In order to overcome the current stovepipe structure and make test resources interoperable, re-useful, and resultant, America's Department of Defense developed a new virtual test framework–the Test and Training Enabling Architecture (TENA) through the Foundation Initiative 2010 (FI2010) project [3]. According to the particular demand in test and training field, TENA expanded HLA to provide more specific ability about test and training.

The development of modeling and simulation promotes the research on the management of simulation resources. In 1995, the United States Department of Defense put forward the Modeling and Simulation Master Plan (MSMP), in which they proposed to construct Modeling and Simulation Resource Repository (MSRR)[4], in which a new method was proposed to unify the resources distributed in different server, and provided users services to manage the resources [5]. The Australian Defense Department also proposed the establishment of Distributed Simulation Resource Repository (DSR), which was based on P2P structure [6]. The domestic research on resource management springs up recently, especially after the HLA becoming the mainstream of simulation technology, National University of Defense Technology, Beijing simulation center and other research units have done research on MSRR, and proposed their own resources management system [7][8].

In reference to other architectures such as HLA and TENA, etc., combined with new demands in the test and training field, we propose a new virtual test architecture HIT-TENA (authors' affiliation is Harbin Institute of Technology, HIT for short). In HIT-TENA, we draw lessons from the resource management concept of advanced virtual test architectures to solve the problems appeared in the process of simulation, the use and management of resource.

II. HIT-TENA ARCHITECTURE

An overview of HIT-TENA is shown in Fig.1. HIT-TENA mainly consists of four basic types of software:

1) **HIT-TENA Resource Application:** HIT-TENA test system is usually built in specific test tasks, using all sorts of HIT-TENA Auxiliary Tool on the basis of common infrastructure. HIT-TENA Resource Application is the basic unit of the test

Manuscript received April 22, 2013; revised July 19, 2013; accepted September 2, 2013.
Project Number: National Science Foundation of China under Grant No. 61201305, Heilongjiang Provincial Postdoctoral Foundation of China under Grant No. LBH-Z11170, and the Fundamental Research Funds for the Central Universities of China under Grant No.HIT.NSRIF.2012015.

doi:10.4304/jsw.9.4.814-819

system. Resource application is built by the test system developer to each node to execute all important tasks in process of the test.

2) **HIT-TENA Common Infrastructure:** HIT-TENA Common Infrastructure means software sub- systems to achieve rapid construction of basic software in HIT-TENA applications. These include HIT-TENA Middleware, for real-time information exchange; HIT-TENA Resource Repository , as a means for storing resources, such as component model, object model and other information; and HIT-TENA Data Archive, for the storage of scenario data, data collected during the test, and so on [3].
3) **HIT-TENA Utilities:** HIT-TENA Utilities is designed for the management of HIT-TENA Common Infrastructure and basic resources, including Resource Encapsulation Tool, Resource Repository Manager, Data Archive Manager, HIT-TENA Gateway and Data Collectors. Resource Repository Manager provides the configuration management and security of the HIT-TENA Resource Repository. Data Archive Manager assists the user in managing and ensuring the consistency of the diverse distributed data in HIT-TENA Data Archive. Data Collectors record public LROM information, and collect data during a test. Resource Encapsulation Tool achieves the encapsulation of all kinds of resources, and then the resources will be interoperability, reuse, and composability. HIT-TENA Gateways are bridges between other isomerous systems and HIT-TENA system [3].
4) **HIT-TENA Auxiliary Tool:** HIT-TENA Auxiliary Tool is designed for HIT-TENA Application, to achieve rapid construction of application with HIT-TENA tools, including Resource Application Integrated Development Environment, Integrated Display Software, Data Analysis and Processing Software and Integrated Environment Supporting Software. Integrated Environment Supporting Software provides different information of environments. Data Analysis and Processing Software provides expended application about data analysis and processing model of some software in HIT-TENA, such as MATLAB and SIMULINK.
5) **Non-TENA Applications:** Systems those are not built in accordance with HIT-TENA, but needed in a logical range [3].

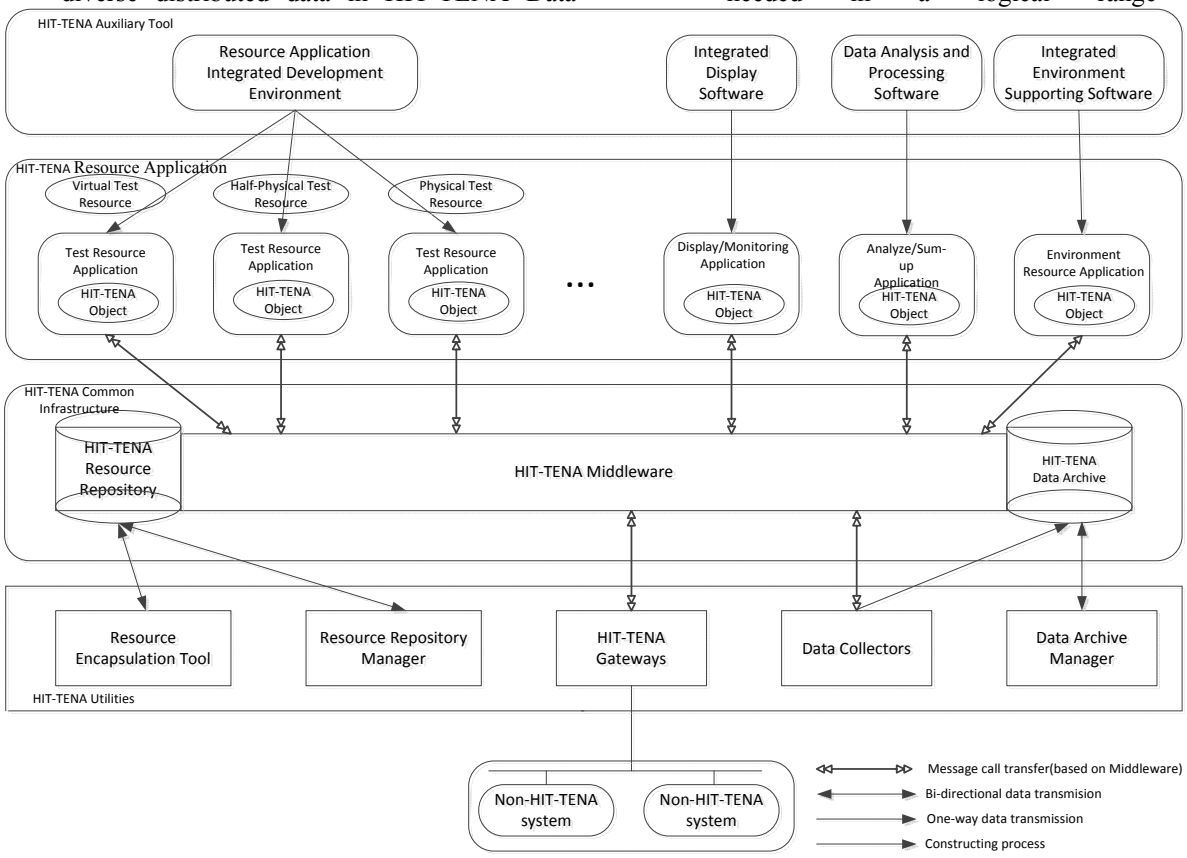

Figure 1. An overview of HIT-TENA

As we can see from Fig.1, this paper will describe the construction of resource repository in HIT-TENA Common Infrastructure, and the design of resource repository manager in HIT-TENA Utilities.

III. Organization of HIT-TENA Resource Repository

HIT-TENA Resource Repository is mainly for the storage of HIT-TENA resources. It is essentially a large, unified, secure database-of-database. Fig.2 shows the organization structure of HIT-TENA Resource Repository.

It appears to the users as a single "logic" repository, and has a mechanism to unify the interface to different categories of information necessary for a range to fully use HIT-TENA [3].

The second tier provides for unification of the schemas of information, and is mainly responsible for establishing "federated multi-database". The Federated Broker share the original data served in Tier 1 through mutual communication, and provide other application software standard service to access information in HIT-TENA Resource Repository.

The third tier is responsible for serving information to the users. Browser can get information from these servers. The servers are in form of Repository Manager.

The fourth tier provides server-side program based on B/S for the users to visit the base information in resource repository through the browser. External program (such as middleware, etc.) can directly visit the repository service provided in second layer.

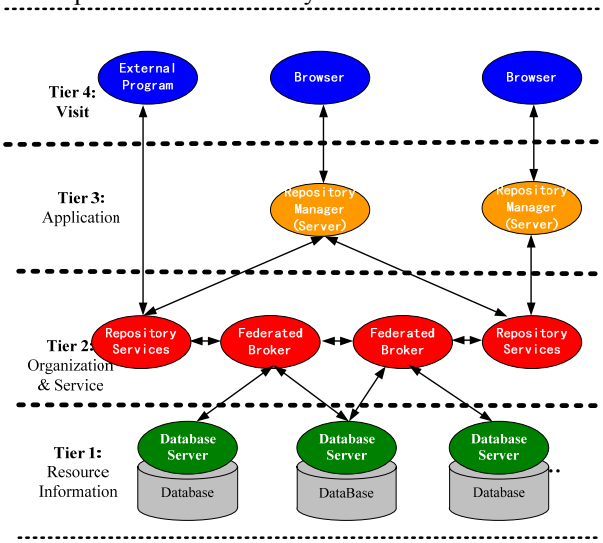

Figure 2. The organization structure of HIT-TENA Resource Repository.

IV. Design of Databases in Resource Repository

A. Database Needs Analysis

The functions of the databases in resource repository are primarily reflected in the management of the various resource information, including storage, update, query, delete. The information need to be recorded mainly includes:

1) **Resource Information:** to describe the characteristics of the resources and make the resource management become convenient. All resources can be divided into two kinds: component model and object model.
2) **User Information:** to describe the users' information such as unit, username and password, make user management and authorization management easy to complete.
3) **Database Information**: to provide the link information of the database services, including server name, IP, etc., to manage the databases.
4) **Authorization Information:** to record the relationship between users and resources, and define the demand and available resources for users.
5) **Database Backup Information:** to record the distributed databases' backup conditions, in order to restore the databases later.

After the analysis, we can design different entities and relationships those we need. Then we can describe them through E-R diagram, and form the foundation of logical structure design. Fig.3 shows the E-R diagram of Resource Repository.

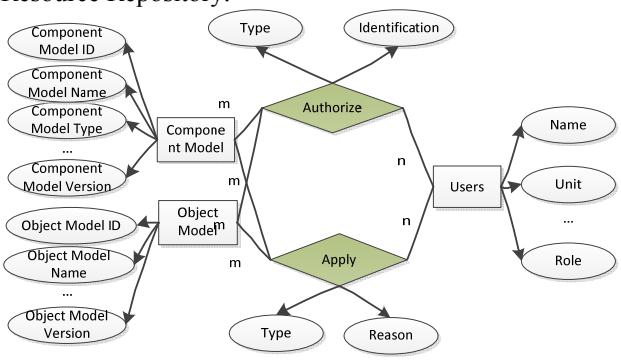

Figure 3. The E-R Diagram of Resource Repository.

B. Design of Database Tables

Through the analysis about the storage information of the HIT-TENA Resource Repository above, we designed the concept structure and logical structure of the database, and finally transformed them into physical structure (design of database tables). Fig.4 shows the relationship between the database tables.

Table Resource and Table ResourceFile can completely describe a component model. Table Resource describes the basic information of component model, such as production information, using information, component version. Table ResourceFile records the file information of component model. They are interrelated through ResID.

Table ObjModel describes object model. Compared to component model, object model's structure is simple, including object model ID (ObjModID), object model name (ObjModName) and three related documents.

Table Users contains the login information of users, including username, password and user roles information (relate to permissions).

Table Userlicese defines each user's different operation authority to resources. On the one hand, ordinary users and administrators own various permissions on visiting and using resources; On the other hand, when it comes to single resource, different users' permissions on it are also different. In addition, this table records ordinary users' demands for resources those beyond their edit

permissions, they can apply for them, and this table stores the application information.

For the rest, they are similar to above tables, no longer introduce in this paper.

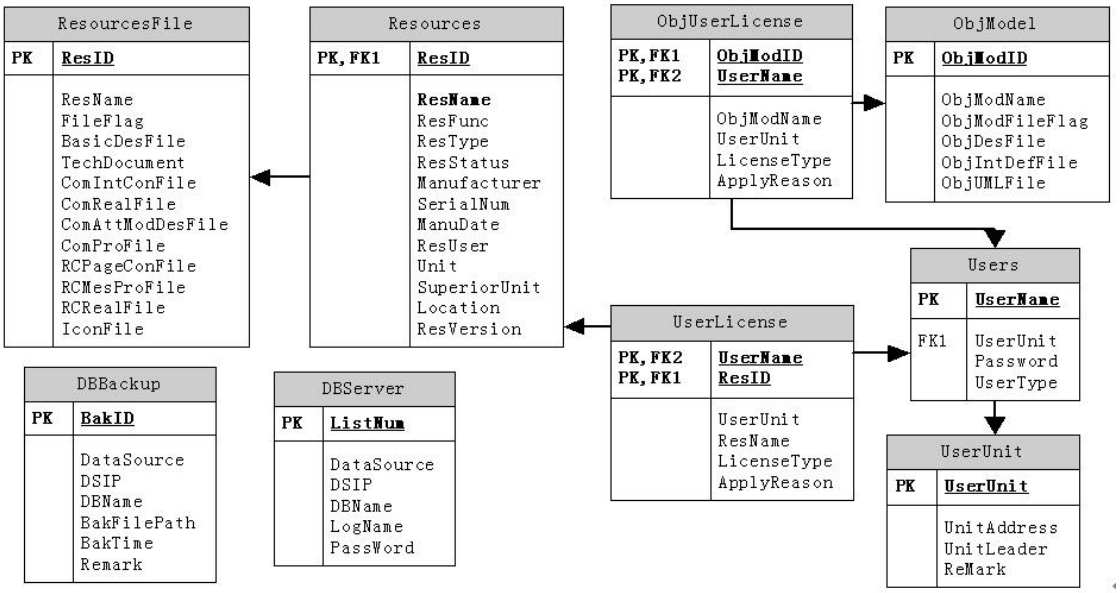

Figure 4. Relationship between the database tables.

V. Design of Resource Repository Manager

Repository Manager provides users visual interface for resource management, including resources adding, update, deleting, database backup and restoring, etc. Repository Manager consists of three parts, data layer realizes the storage of all kinds of resources, logic layer is responsible for resource repository services and unified management, presentation layer provides visual management interface and visits the repository through interface services. Fig.5 shows the functional partitioning of the resource repository manager.

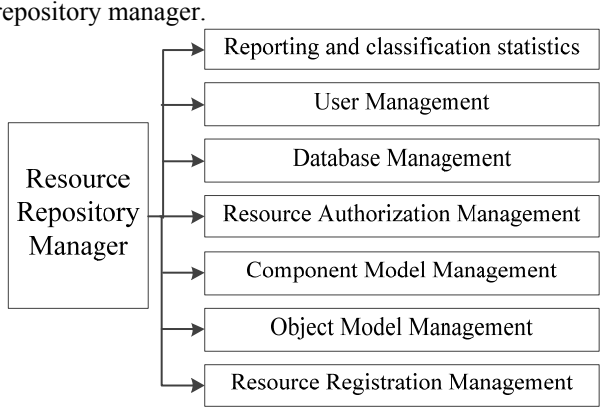

Figure 5. The functional partitioning of the resource repository manager.

A. Design of Repository Manager's Main Modules

1) Resource Registration: For new resources, users can store them through the registration mechanism provided by the resource manager. Whether component model or object model, users must first choose the database to register (prerequisite: the database should be online).

The component model is registered in two ways, one is copy registration, which copy from the existing basic description XML file of the component model, another one is manual entry of all the component model information. Component model information includes component model basic information (such as name, ID, function and state), component model production information (such as manufacturers, serial number and date of production), component model using information (such as users, unit, superior, etc.) and the relevant documents of the component model.

The description of object model is simple and similar to component model, we no longer introduce it.

2) Resource Retrieval: As a lot of resources are stored in databases in practical application, we need a powerful retrieval tool to help users fast lock the resources. The retrieval fields include basic information of resources, production information, as well as the using information. The resource retrieval also supports retrieval of joint fields, in which the different retrieval relations between various fields include and, not, and or. At the same time, users can add matching degree (accurate or fuzzy) to the retrieval.

Resources Repository Manager provides four conditions and forms different SQL query through the set of retrieval fields to visit the databases, finally obtain the specified resources.

3) Resource Authorization management: Resources authorized process is that administrators grant or cancel ordinary users' operation permission of resources. For authorization management, there are four main ways in Resource Repository Manager.

- **Single resource authorization management**: Administrators can enter the authorization management interface of the resource they are

managing, then the system will display the usage of the resource, according to it, administrators can grant or cancel users' authorization. In the system design, the use of the AJAX local refresh technology make two table refresh automatically when the authorized type changed.

- **Single user's resources authorized management**: Administrators can view user's use of all resources, that is authorized or not, then enter the authorization management interface of the specific resource to modify the permission.
- **Multiple resources and multiple users' authorization management**: Administrators can carry out resources batch authorized for multiple users through the component model authorization management. Administrators can give multiple users the edit permissions of any component model.
- **Application and management of authorization**: Ordinary users can apply for resources those beyond their authorized limit when they do resource browsing, then administrators will handle the applications.

B. Repository Manager Access Speed Optimization

As Repository Manager appears in forms of web pages, the access speed of web pages, resource query and loading speed will determine the users' experience, so it is necessary to optimize the access speed of the web pages.

All distributed databases have the same structure. If users want to login, they must via the center server's validation, at the same time, the program will read the database link information and judge if the databases are online, then present users the resource information located in online databases, finally realize the distributed access. But this mechanism has its deficiency. The program must judge the connection conditions of all distributed databases when we want to get the resource information, the offline database detection will spend a lot of time, lead to the low speed of resources' present.

In order to solve the problem, we make the following design: The program will automatic open a thread timer, and then detect the connection of each database on time, then return the online databases as result, and refresh. When users want to obtain the resource information in each database, only visit the online databases through the information returned by the thread.

VI. Design of Resource Repository Web Services

A. Resource Repository Web Services Needs Analysis

Resource Repository must exchange information with other tools in HIT-TENA architecture, such as Resource Application Integrated Development Environment needs to call the resource retrieval service to genetic the list of resources, also obtains the information of specific resources; Data Collectors, HI-TENA Gateway, Data Analysis and Processing Software, Integrated Environment Supporting Software and Integrated Display Software need to call the resource retrieval service to obtain the selected resources' information; The middleware needs to call the resource retrieval service to obtain the selected resources' information in system modeling and system operation process. Therefore, it is necessary to design specific Web service interfaces for other tools in platform to call and get relevant information through them.

B. Design of Resource Repository Web Services

TABLE I.

Type Sizes for Camera-Ready Papers

Funcname	Function	Input	Return
DetectDBConnection	Detect the connection conditions of distributed database	None	Number of efficiency links
SetDectPeriod	Set time of database detection	Password, Time	Success or not (True, false)
GetAllRes	Get basic information of component models in Resource Repository	None	Arrays of structures (all basic information of different component models)
GetAllResNum	Get the number of all component models	None	Number of all component models
GetAllObj	Get basic information of all object models	None	Arrays of structures (basic information of all object models)
GetAllObjNum	Get the number of all object models in Resource Repository	None	Number of all object models
GetSpeObjFile	Get the specific file of selected object model	Given condition	Document binary flow (selected file object model)
GetSpeRes	Get basic information of specific component model	Given condition	Structure (information of selected component model)
GetSpeFile	Get the specific file of selected component model	Resource name, File type	Document binary flow (selected file of component model)
StatByColumn	Get statistical information of specified component models	Given condition	Arrays of structures (information of selected component models)
GetSpeResNum	Get the number of selected component models in Resource Repository	Given condition	Number of the selected component model
GetResLibStatus	Get the status of special database	None	String about the detail information of connection conditions of databases in Resource Repository
ConnectDatabase	Link given database	None	None
StatByColumn	Get number of selected special component models	Given condition	The number of selected component models

Web Service is a new branch of Web applications, self-contained, self-describing, modular application [9]. A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP-messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards [10]. The data format of Web Service is XML, which has the advantages of platform independence, so the Web Service's work processes do not need to consider the system and platform of the client and server [11][12]. In fact, Web Service is to realize the communication between applications, or is to achieve the cross-platform interoperability. Therefore, HIT-TENA Repository Service was written by web service mechanism. Table I shows the main functions.

VII. CONCLUSIONS

This paper mainly introduces the design and realization of HIT-TENA Resource Repository. After abstract analysis about the distributed virtual and physical resources, describe them through database tables, and use the database management system to plan and organize, finally construct the distributed database system and provide users Repository Manager to manage the whole database system. The above is the innovation point of this design. HIT-TENA Resource Repository has many functions such as adding, updating, authorized resources and users, and provides database access interfaces for other parts of HIT-TENA architecture to call, to good effect.

ACKNOWLEDGMENT

This work is supported by National Science Foundation of China under Grant No. 61201305, Heilongjiang Provincial Postdoctoral Foundation of China under Grant No. LBH-Z11170, and the Fundamental Research Funds for the Central Universities of China under Grant No.HIT.NSRIF.2012015.

REFERENCES

[1] DAI Kun, ZHAO HaoLong, "TENA Based Implementation Technology for Virtual Test", Journal of System Simulation, no.5: pp.20-23, 2011.
[2] IEEE Recommended Practice for High Level Architecture (HLA) Federation Development and Execution Process (FEDEP), IEEE Std, 1516.3TM-2003, April 2003.
[3] The Test and Training Enabling Architecture Reference Document, 2002.
[4] Zhang Peng, "Research on Key Technologies for Resource Management System under the integrated environment of M&S", National University of Defense Technology, pp.2-10, 2011.
[5] Zhang Yong, "Research on the Integration Management Method of Military Simulation Resource", National University of Defense Technology, pp.17-20, 2009.
[6] Shawn Parr, Anthony Jones, "The Distributed Simulation Repository (DSR): Realizing Collaboration and Reuse through P2P Communities", Calytrix Technologies Pty Ltd Technology Park.
[7] Zhao YaoCheng, "Research on the mechanism of simulation resources management and sharing", National University of Defense Technology, pp.15-18, 2002.
[8] LIU HanBing, SU Hong-yi, CHAI Xu-dong, "Research and Application of XML in Modeling and Simulation Resource Repository". Journal of System Simulation. no.11, pp.18-19, 2009.
[9] Zhengqiu He, Lifa Wu, Huabo Li, "Semantics-based Access Control Approach for Web Service", Journal of Computers. Vol. 6, No 11, pp.1152-1154, June 2011.
[10] W3C, Web Services Glossary, http://www.w3.org/tr/2004/note-ws- gloss-2004 .0211.
[11] Zhiqiang Yao, "Implementation of the AutoComplete Feature of the Textbox Based on Ajax and Web Service", Journal of Computers, Vol.8, no.9, pp.2197-2199, September 2013.
[12] Jianbo Bai, Yuzhe Hao, Guochang Miao, "Integrating Building Automation Systems based on Web Services", Journal of Software, Vol.6, No.11, pp.2209-2210, Nov 2011.

Lian-lei Lin was born in Gaotang, Shandong Province, China in July 1980. He received the Doctoral degree in Instruments Science and Technology from Harbin Institute of Technology (HIT), Heilongjiang, China in 2009 and 2004 respectively. Now he is associate professor of HIT. His scientific research interests include virtual test technology, machine learning, signal processing and automatic test techniques.