

# Research and Application of Development Model of Information Service for IOT of Oil and Gas Production Based on Cloud Architecture

Wu Haili<sup>\*1</sup>, Gong Renbin<sup>2</sup>, Wang Congbin<sup>3</sup>, Gong Lei<sup>4</sup>

Research Institute of Petroleum Exploration & Development-Northwest, PetroChina  
No.535 Yanerwan Road, LanZhou, GanSu 730020, China, Ph./Fax: +010-51551230

\*Corresponding author, e-mail: wuhl\_xb@petrochina.com.cn

## Abstract

*On completion of oil and gas production IOT system application system needs expanding and secondary problems such as software development integration service, this paper proposes a scalable cloud platform(called A11-PaaS) based on middleware and ESB(Enterprise service bus). The platform realizes that managment of ESB, controls the service request access on the ESB with the LDAP, use the WAS profile as a sandbox for the development, combine with the Maven plug-in and Nexus, realize the unified management of the secondary development, testing, and deployment of the new system and achieve the purpose of rapid development. The platform has been deployed in some oil company, the experimental results show that this technology has achieved rapid development, integrated the web service from IOT system, and provide effective method to integrate other application system.*

**Keywords:** IOT, PaaS, sandbox, Continuous Integration, ESB

Copyright © 2017 Universitas Ahmad Dahlan. All rights reserved.

## 1. Introduction

### 1.1 Background

With the increasing popularity of cloud computing technology and a large number of industrial applications of cloud computing, cloud computing gains increasing acceptance from the industry in the availability, scalability and other aspects of services. According to the definition of the National Institute of standards and Technology (NIST), services of cloud computing are divided into three levels, including SaaS, PaaS and IaaS. PaaS provides encapsulation of the middleware platform, components and operating environment, push the abstraction of resources into a higher level [1]. PaaS administrator provide development environment, test environment and server platform to support users build their own applications, including design, development, implementation, testing, deployment and maintenance, and help users develop through providing much-needed services as PaaS's service directory.

IOT of oil field maily uses the sensor, radio frequency, communication and other technologies to detect oil and gas production unit, and implement automatic collection and control of production data, production monitoring and management. IOT of Oil field is a unified platform which achieves the common needs of oil and gas production, but due to the different oil field business, there are some personalized requirements in the expansion of the IOT oil field's functions needed some efficient development methods and techniques to achieve.

### 1.2 The Problem

For the traditional system development methods and techniques, they have common points as follows. In the life cycle management of application, the efficiency of every stage is not high enough. The whole life cycle of application development includes 6 stages: development, test, deployment, expansion, upgrading and maintenance. In the development and test stage, needs to build environment and application; in the deployment stage, needs to apply for hardware, install software stack, configure and deploy; in the expansion stage, the need for new hardware applications, software stack installation and cluster configuration; in the upgrading and maintenance stage, the need for independent software and hardware maintenance,

independent version control and independent monitoring configuration. In the aspect of system integration, it is needed to communicate with the service developer several times so as to determine the call interface. At the same time, because of the difference of development language, often also needs to provide a new way of calling the service, and so on, all these have increased the cost of communication and reduced the efficiency of the work.

And the functional modules of the traditional IOT systems and other applications can only be used in these systems themselves, which reduces the diversity of applications. For example, there are many components, such as oil and gas field location, seismic data system and so on, all these components or applications have long development cycle and high complexity, but these components are closed, that is these completions can only be used by their own developers, which reduces the benefits of component reuse. The traditional waterfall software development model has resulted in a large number of duplication of research and development, improved the cost of software development. With the widely use of enterprise PaaS platform, the diversity and convenience of services have gradually become the core difference between PaaS, therefore to explore how to provide a variety of reusable components simply and effectively and agile development become an urgent problem to solve.

### 1.3 The Proposed Solution

Cloud computing and Internet of things both have a lot of advantages and characteristics. Cloud computing is equivalent to the human's brain, belonging to the cognitive system, while Internet of things is equivalent to the eyes, ears and limbs, belonging to the perception system. Article 2 proposed to combine the cloud computing and Internet of things, and put forward a kind of intelligent logistics system, which is applied to the army [2]. Article 3 provided a integrated application of cloud computing and Internet of things technology, put forward a comprehensive port logistics service platform, in order to adapt to the new business needs of the port logistics [3]. Article 4 focus on designing and implementing PESMS(PaaS Environment for Social Multimedia Service) including a transcoding function for processing large amounts of social media in a parallel and distributed manner based on hadoop [4]. Article 5 focuses on the security issues encountered in PaaS clouds [5].

Research and development of cloud platform technology are mainly guided by some of the world's famous enterprises. IBM announced the cloud computing program in 2007, which was the concept of cloud computing first appeared; EMC developed Atmos cloud storage system with Intel in 2009; Red Hat provides 4 cloud computing solutions, including IaaS, PaaS, SaaS and the extension of SaaS, and provides cloud computing platform for Amazon; HP's matrix of blade system provides the basis for cloud computing platform, which reduces the cost of infrastructure and the complexity of data center; Intel proposed the idea of open data center in 2010, and so on.

The existing cloud computing technology contains 3 aspects of the characteristics [6], respectively: a. hardware infrastructure in large scale and cheap server cluster; b. applications and the underlying service collaborative development, to maximize the use of resources; c. achieve high availability of software by redundancy of cheap server, so as to achieve scalability and high availability goals.

In this paper, we combine cloud computing and Internet of things technology in the field of oil and gas production to meets the business development requirements in the field of the oil and gas production. In the three kinds of cloud computing's applications, PaaS can be used in the IOT of oil and gas production system, so we implement a PaaS platform called A11-PaaS. A11-PaaS is a private cloud platform which has functions as follows: application registration, application development, application testing, service registration, service hosting, service routing, service scheduling, service monitoring and service directory.

And after using PaaS, it makes the life cycle management of applications simple as Figure 1 shows.

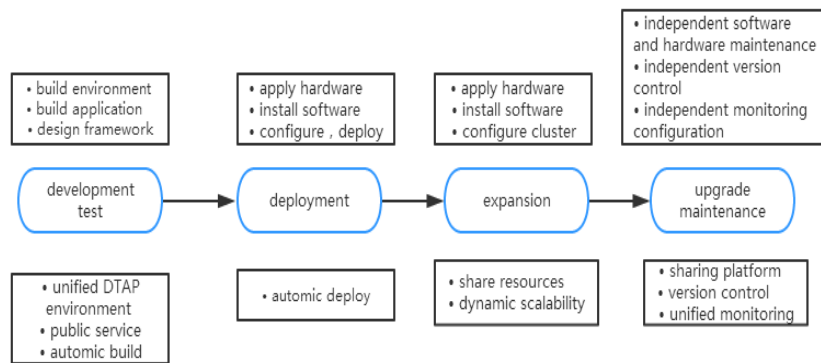


Figure 1. ALM of Software Development Before and After Using PaaS

## 2. Design of System

### 2.1 System Architecture

A11-PaaS is mainly divided into two parts: aPaaS and iPaaS. aPaaS is divided into two parts: platform management and background services. Platform management includes: account management, system management, personnel management, task management, application management, automatic generation of code, version management, service configuration, continuous integration configuration, sandbox and container configuration, resource management and authority management. Background services mainly includes continuous integration service which is included in application's development, testing and release stages. In the development stage, it is mainly responsible for the developers' sandbox and version control. In the test stage, it is mainly responsible for the management of test sandbox. In the release stage, it is mainly responsible for the management of the application container.

iPaaS mainly uses the Enterprise Service Bus to complete the integration of various existing services, which includes data services of IOT for oil and gas production, A2, A5, A8 and other internal business services, PaaS's platform services and other third party services, etc. And Figure 2, iPaaS completes the registration, scheduling, routing, monitoring, directory and security management of these services mentioned.

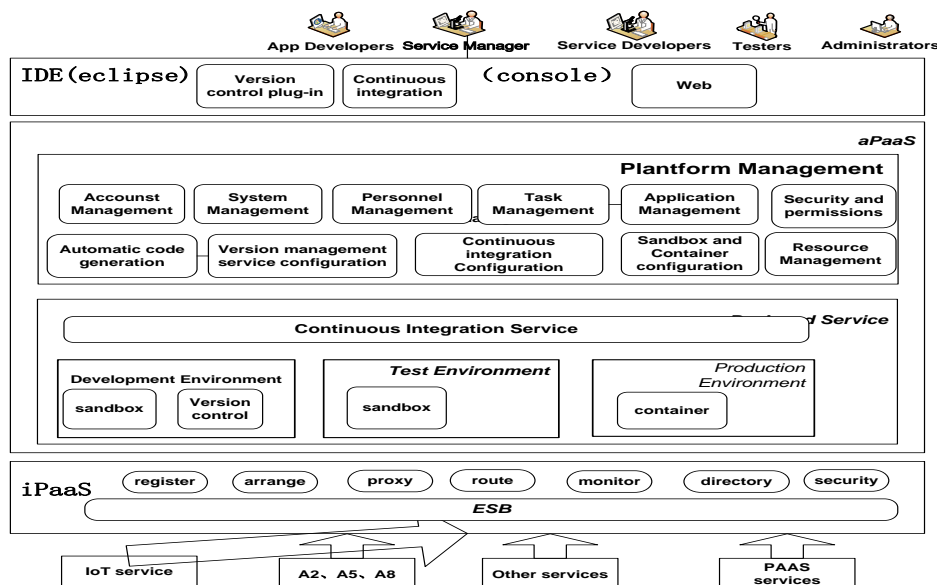


Figure 2. Architecture Diagram of A11-PaaS

## 2.2 Function Design

In order to support the rapid and convenient implementation of the application, the development of services, A11-PaaS supports the following main functions.

- a. Role management, responsible for different tasks. In the process of information service development, different roles, for example, the developer, the tester and the PaaS platform manager and so on, are needed to provide different functions.
- b. Application registration. Developers register a new application before developing, after administrator approve the registration, A11-PaaS automatically provides the basic framework for application code and development template for developers, this greatly reduce the time to build the development environment and development framework.
- c. Application development. A11-PaaS provides the development library and service directory to developers, assists them to develop new applications which have a unified interface style, and supports the rapid development of applications.
- d. Version control. Developers can use SVN to manage their own code through A11-PaaS which provides a tag interface of SVN. developers for the new version of tag tag, label directly in the SVN version. When developers mark the tag of the application's new version, it's directly marked to the version number of the SVN.
- e. Automatic source compilation and deployment. When the developer complete a new version code of an application, and submits to the SVN server, A11-PaaS will automatically compile and deploy this code, and release it to the application container which A11-PaaS provides. support, two development and application of automatic deployment. And A11-PaaS supports automatic deployment of developed applications.
- f. A11-PaaS can utomatically generate test environment. After the new version of the application code compiled and deployed to the sandbox, sandbox as the test environment of the application, developers and testers can start, stop and restart their application sandbox
- g. Real-time monitoring of applications' operation condition. Administrators monitor all applications running on the platform, record their key operation data which are some related hardware and software resources, including CPU, memory, operating system resources, database and application container etc..
- h. Dynamic extension of application resources. According to the monitoring resources usages of the applications, once the usages are more than the set of resource limits, A11-PaaS will automatically extend applications' resources according to the setted rules.

We use the following softwares to implement A11-PaaS.

Table 1. Components of A11-PaaS

Softwares	Functions
J2EE	Using Java language, provides the basic framework for the development of A11-PaaS platform
WAS	Provide sandboxes for the development, test and release processes
Nexus/Maven	Using jar file library to manage files, provides the required jar files for development of applications
Hudson	Completes the continuous integration process, automaticly completes applications' compilation and deployment process
SVN	Source version control
ESB	Rout and distribute messages to the target webservice, complete access adapter, security authentication, data and log records, shield webservice's real location, protocol and other technical details to the upper layer application

## 2.3 Physical Composition

Figure 3 shows the physical architecture of A11-PaaS. The test environment server, provides the physical environment for the test sandboxes. Database server provides database services for applications' development. SVN version control server provides source code version management services for applications' development. Hudson, the continuous integration server, automatically compiles, publishes the source code, which reduces manual operation. ESB

server provides the function of service bus. Production environment cluster provides the operation environment for the final deployed applications.

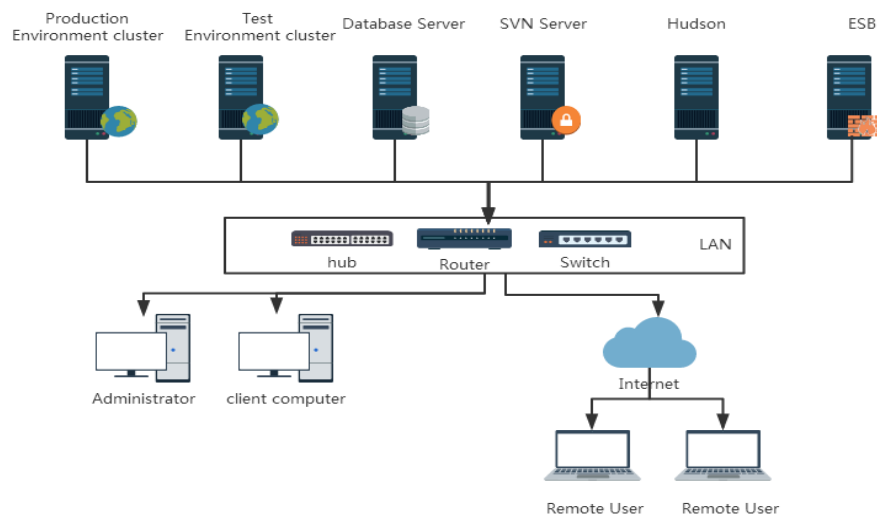


Figure 3. Physical Composition Diagram of A11-PaaS

### 3. Key Techniques and Implementation

#### 3.1 Continuous Integration

Continuous integration is an important practice of agile development. Generally speaking, different developers frequently submit codes to code trunks, and then the new version of the software is frequently delivered to the quality team or the user for review. Finally, the reviewed codes are deployed to the production environment. Continuous integration includes continuous delivery and continuous deployment.

A11-PaaS uses the Hudson components to achieve continuous integration. When a developer submits source code to the SVN, SVN will detect changes in the code and trigger construction system to build automatically. Construction system firstly obtains the latest copy of the code from the SVN, secondly executes the construction script, finally returns the construction results to the relevant developer [7]. After the successful construction, this application is automatically deployed in the sandbox, and the construction system will return the deployed url. Figure 4 shows the specific application of continuous integration in A11-PaaS.

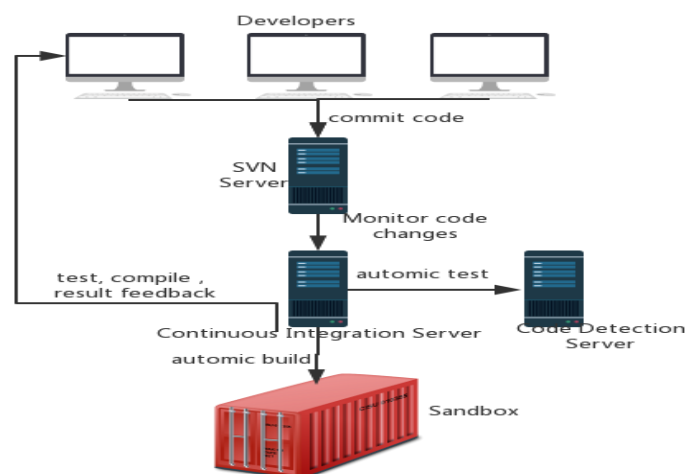


Figure 4. Process of Continuous Integration

Through continuous integration, A11-PaaS can help developers quickly find reasons of compilation or test failure, avoid the risks of traditional software development, improve the efficiency of development, and make the application development lifecycle management convenient.

### 3.2 Security Design of Service Access

The focus of security objectives in the cloud environment are data security and privacy protection. There have been more and more standard organizations began to work on cloud computing's security standards [8], to enhance the interoperability and security and reduce duplication of investment or reinvention, such as the ITU-TSG17 research group, structured information standards and so on. At present, the research on the key technologies of cloud computing's security mainly includes [9]:

- a. Trusted access control;
- b. Ciphertext retrieval and process;
- c. Data's availability;
- d. Data privacy protection;
- e. Virtualization security technology;
- f. Cloud resources' access control.

Cloud computing's security architecture has also become a hot research [10], it includes:

- a. Security architecture based on trusted root;
- b. Security architecture based on isolation;
- c. Security architecture of "Security as a Service";
- d. Controlled and measurable security architecture.

Because A11-PaaS is a private cloud platform, its security requirements are different from the public cloud computing's. A11-PaaS uses a method of finely granular access control in the services registered on the ESB, and uses LDAP to verify the access permissions of applications. All services registered on the ESB are managed by A11-PaaS. All applications doesn't have directly access to the services, to ensure the security of cloud services. For the accesses to the services registered on the ESB, A11-PaaS assigns a service access account to each application. And the number and types of services are assigned according to the need of each account. On A11-PaaS in Figure 5, developers can browse all the services' introductions, but can not view the address of ESB.

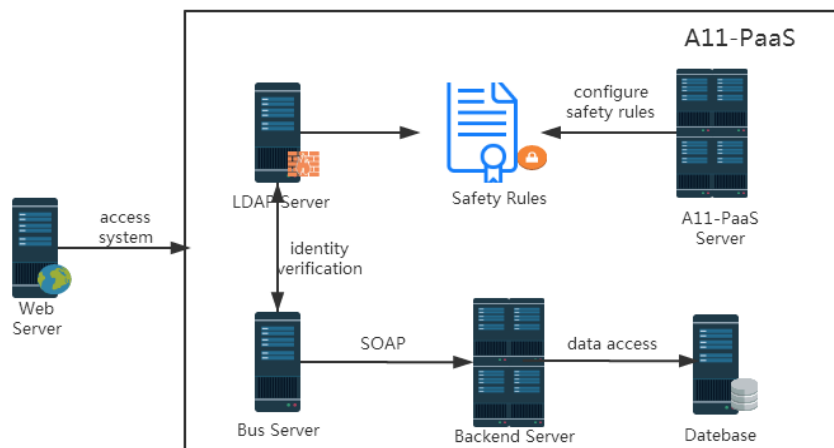


Figure 5. Authentication Process of Service Access

Developers apply for accesses to the services wanted to invoke or use to the administrator as desired. After the administrator passes the application, the platform authorizes the access to the corresponding service to the developer's application account, and sends the

service address to the developer. When the web application requests a service, ESB firstly analyses the service request, extracts authority verification tokens from the request, and sends the tokens to the LDAP server for authentication. LDAP verify permissions according to A11-PaaS's authentication rules and information. If the verification is passed, the application is allowed to access the backend service via the bus, and forward the results from the background to the requester; if the verification is not passed, the application will be rejected.

### 3.3 Development, Testing and Release Processes of Applications

Figure 6 shows how the Application Life Management is applied on A11-PaaS.

In development process, configuration administrator firstly registers an application, then specifies the developers and development language, and sets the SVN code library.

Developers download the framework of codes and develop, select the needed components and services registered in A11-PaaS which need administrator to authorize.

- A11-PaaS sets the version number of applications registered by developers, and provides the framework and plug-ins for development.
- A11-PaaS through controlling WAS server to automatically generate sandbox, and help developers simulate the production environment which is used to debug codes.
- Developers apply for the accesses to services registered in A11-PaaS, with administrators' authorization, developers have permission to invoke.
- Upload the codes to SVN server to achieve source version management.

According to the software development process, after the completion of the applications' development, it's needed to test. Test and release processes include:

- Firstly, developers apply for application test, after administrators review this application, A11-PaaS will start test process automatically.
- Testers receive test tasks, and apply to initialize the test environment. A11-PaaS generates test sandbox automatically, and marks version number to the application's source codes.
- A11-PaaS starts the process of continuous integration, source codes are automatically compiled and deployed to the test environment.
- After the test process is executed successfully, testers apply to publish this application's codes to the production environment.
- After the administrators review this application, start to deploy this application's codes to the production environment. A11-PaaS generates production environment sandbox automatically.
- Lastly, A11-PaaS activates continuous integration process, the new version of the application code is automatically compiled, and deployed to the production environment sandbox.

After completing these above processes, the application's whole life cycle is over.

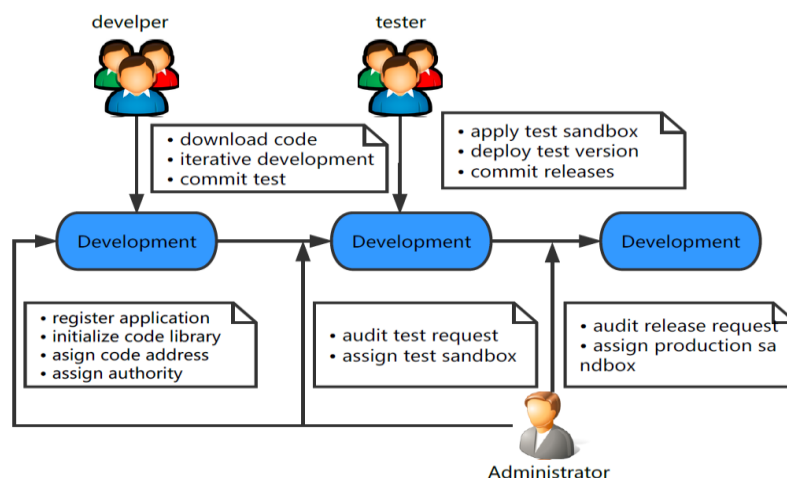


Figure 6. Application Life Management in A11-PaaS

#### 4. Results and Analysis

Analysis from the application's whole life cycle management, in the development process and test process, A11-PaaS provides a unified DTAP environment (Development-Test-Acceptance-Production), manages the public services access and supports applications' automatic construction process; in the deployment process, A11-PaaS supports the applications' automatic compilation and deployment; in the application's extended process, provides a shared resources pool and supports dynamic expansion of applications' required resources; in the maintenance process and upgrading process, you can carry out a unified software and hardware maintenance, version control and monitoring.

Analysis from the system integration, it changes the traditional system integration method, at the same time, simplifies the existing service calls method, and there is no need for complex communication, improves work efficiency. Combining cloud computing with Internet of Things, which can be used for the Internet of things platform to provide a more flexible and scalable platform. From the application results of several test areas, A11-PaaS basically overcomes the shortcomings of the traditional development methods, and meets the goal of the expansion of the Internet of things.

Table 2 shows the comparison of A11-PaaS, Alibaba Cloud and Ge Predix. Alibaba Cloud is based on OpenStack, and GE Predix is based on CloudFoundary. While All A11-PaaS is implemented by ourselves. A11-PaaS has the following advantages: firstly, A11-PaaS is mainly used in IoT of Oil and Gas field, which has many API and services about IoT of Oil and Gas field; secondly, the most important is that A11-PaaS supports Oracle, which is widely used in Petrochina, and Oracle is not supported by many PaaS Platform including Alibaba Cloud and GE Predix; thirdly, we implements some components built in development tools(such as Eclipse), which is easy to use.

Table 2. The comparison of A11-PaaS, Alibaba Cloud and Ge Predix

PaaS	Occurrence Time	Emphasis	DevOps Usability	Supported Databases	dependency
A11-PaaS	2013	IoT of Oil and Gas, DevOps Internet	Components built in development tools CLI, Open API	MySQL, Oracle, Redis, etc	None
Alibaba Cloud	2009			Not include Oracle	OpenStack
GE Predix	2015	Industrial Internet		Not include Oracle	CloudFoundary

#### 5. Conclusion

In the personalized demand of oil field's Internet of things, this paper introduces the development method of information service based on cloud computing. A11-PaaS greatly improves the application development speed, and reuse services of the old system, but also deepens the applications integration between systems. A11-PaaS has been applied in several pilot oil fields, and helps quickly and efficiently complete the application development.

#### References

- [1] Chen Quan, Deng Qianni. Cloud Computing and Its Key. *Journal of Computer Applications*. 2009; 29(09): 2562-2567.
- [2] Zhang Yueling, Li Yanxiao. Design of Intelligent Logistics System Based on Cloud Computing and Internet of Things. *Computer Science*. 2012; 39(Z6):212-213,241.
- [3] Han Haiwen, Qi Deyu, Feng Bin. Construction of Port Logistics Service Platform Based of Cloud Computing and Internet of Things. *Computer Science*. 2013; 40(Z6):232-235,261.
- [4] Jongjin Jung, Myungjin Kim, Hanku Lee. A Study on Efficient Design of a Multimedia Conversion Module in PESMS for Social Media Services. *International Journal of Electrical and Computer Engineering (IJECE)*. 2015; 5(4):821-831.
- [5] Devi T, Ganesan R. Platform-as-a-Service (PaaS): Model and Security Issues. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2015; 15(1):151-161.
- [6] Chen Kang, Zheng Weimin. Cloud Computing: System Instances and Current Research. *Journal of Software*. 2009; 20(5):1337-1348.
- [7] Liu Qianling, Fan Bingbing, Huang XingPing. Research and Application of Continuous Integration Based on Hudson. *Computer Systems & Applications*. 2010; 19(12):151-154.



- 
- [8] Zhu Lei, Zhou Minghui, Liu Tiancheng, Mei Hong. A Permission Management Model in Service-Oriented Architecture. *Chinese Journal of Computers*. 2005; 28(4):677-685.
- [9] Feng Dengguo, Zhang Min, Zhang Yan, Xu Zhen. Study on Cloud Computing Security. *Journal of Software*. 2011; 22(1):71-83.
- [10] Lin Chuang, Su Wenbo, Meng Kun, Liu Qu, Liu Weidong. Cloud Computing Security: Architecture, Mechanism and Modeling. *Chinese Journal of Computers*. 2013; 36(9):1765-178.