Proceedings of the 5th WSEAS International Conference on Telecommunications and Informatics, Istanbul, Turkey, May 27-29, 2006 (pp1-6)

Connection Management System for Web Services based on a Policy

Woon-Yong Kim, Seok-Gyu Park

Dept of Computer Technology & Application,
Gangwon Provincial University,
Gangneung, 210-804,
Korea

Abstract: - Web services have been recognized as international standard that are feasible for agile and real-time cooperation and integration between individual, corporation and government and offer how to easily integrate IT technologies which are based on Internet. These technologies have been accommodated to the business environment which is based on Service Oriented Architecture quickly. However, the part of problems which are under Web services construction and application are made a decision by quality of services depended on service providers, and these services lack efficient management. To achieve it, we propose the system which provides dynamic connection of Web services based on a policy using Web services pool. It will provide stable and reliable services and construct efficient services management environment.

Key-Words: - Web Services, SOA, Load balancing, Quality Evaluation model, Connection Management

1 Introduction

The recent IT issues are making fast, real-time collaboration model and creating new business using it by extending existing infrastructure. This purpose is presented notably in SOA. SOA makes processes to components at the point of business view and offers various services flexibly, so that it is recognized as suitable technology for rapidly changing IT environment [7]. Web service has been utilized for ideal implementation of SOA. Web services that are an international standard technology which is able to collaborate between individuals, enterprises and governments can easily and quickly integrate IT environment based on the Internet [10]. Web services consist of Service Provider, Service Broker, and Service Requester. These offer standard methods for describing, registering and discovering services [8][11][12]. Also web avail-ability and reliability are critical factors for Web service quality [4]. There are two possible problems in Web service creation and utilization. First, the service quality is decided with depending on service providers, and second problem is efficient management of services is lacked in current Web service. In general, there could be several Web services which have same purpose on Internet. That is, some service providers might offer and deploy same purpose services, so there are two ways of utilizing services. One is using only one service out of many same purpose services; user authentication service, the other is using several services at the same time; service for comparing a price of products. These services might offer different quality of services depends on the service environment and point of servicing time. Therefore, an application for an integrated management of service and a policy decision can provide an advantage for the reliable service.

This paper proposes dynamic connection system of Web services based on a policy through service pool. It offers service allocation and load balancing policy in service group environment and stable service by managing available services. Removing the direct connections between service providers and users minimizes changes of service and also it is easily adapted to IT environment when they use the connection system.

The rest of this paper is organized as follows: section 2 presents service quality management and related technology. Section 3 describes service connection management system based on a policy. Section 4 presents implement of system and section 5 presents conclusion of this paper and future work.

2 Related Work

2.1 Traditional Approach to Web services

Web services consist of Service Requestor, Service Provider and Service Broker (UDDI). The relation of three factors is shown below (fig 1. Left). Service Requestors use UDDI for discovering services and connect directly with particular service provider for using discovered service [10].

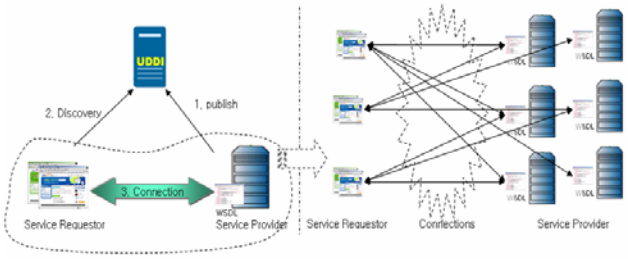

Fig. 1 Relation of Web service composition factors (Left), Connection relation between requestor and provider (Right)

After service is discovered, the service which service requestor uses is connected directly with particular service provider, so the quality of this Web service fully depends on particular service provider. There are various solutions for the service quality problem, such as SLA assurance technique [3] [9] and service grade [5]. These techniques always serve on the fixed paths because services request in a fixed way between Service Requestor and Provider. The right feature of Fig. 1 shows the connection form for service requests between requestors and providers. Due to the direct connection of service requestor and provider, if the availability and reliability of ser-vice provider are declined, the quality of service requestor is declined, too. However, there are several Web services which offer same service; we can use them as alternatives without depending on particular service, so providers can provide the reliable Web service. In the phase of clients, they have to manage the change of service registration. When the requestor uses many of same services, offering fast applicable way to utilize frequent changing Web services without modification of clients also improves utilization of services. We propose service connection management system based on a policy for those purposes in section 3.

2.2 Web service Quality Assurance Technique

Recent interests of Web service quality are embodied in OASIS, National Computerization Agency and so on. For example, OASIS constitutes technical committee for quality evaluation model on last September and National Computerization Agency is constructing Web service quality integration management system. Quality Evaluation model in these systems includes registering Web services which are evaluated more than specific level at national UDDI after evaluating service quality provided by ser-vice provider, and its purpose is to guarantee development cost, quality followed by the authenticated level [5]. However, these quality assurances also fully depend on service provider.

3 Service Connection Management System based on a Policy

3.1 System Environment

In generalized policy based service connection management system, the relation of each factor is illustrated in fig. 1. Service requestor requests specific service which is concerned about service character and environment using service connection management system. At this time, service connection management system provides optimal service concerning with request form of service, policy and current service status through group service management. These services are managed by Service Pool and are grouped by purposes of services in the Service Pool. Service Pool also maintenances usage procedure of service which is allocated to service requestor and utilizes them as property of the service when the service is requested. Therefore, Service Requestors obtain available and reliable services and service provider can solve the problem of service failure and load congestion.

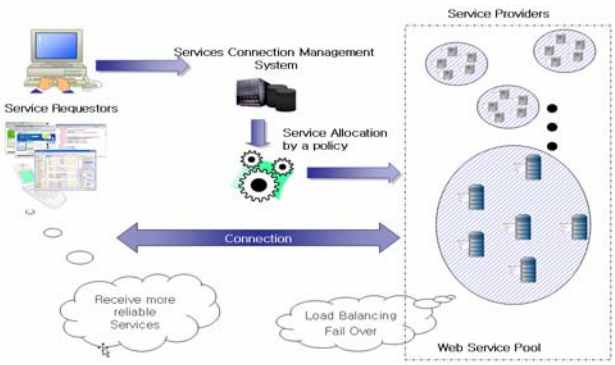

Fig. 2 Relation among Request, Connection Manager and Providers

Service connection policy and operation technique of connection management through Service Pool are discussed in the section.

3.2 Services connection policy

Requirement for making policy of service connection is constructing pool of services. The service pool as shown in fig. 2 can describe as a service group which provides same functionality. Generally, many services deployed on the web include function-ally same services. Therefore we need grouping for providing efficient management of the services. Group also can be divided into two types by request forms of clients: single selection and multi selection. Consequently connection policies are needed as their characteristics. The objective of single selection service type is to select optimal service concerning current environment in service group. For this objective, we use Load Balancing and Fail Over approach and utilize current utilization information of the service and use log information. Multi selection service group, as a form which invokes all services in particular group, provides environment which always dispatches services that are registered before invoking time without modification of client. Table 1 shows service types and approach for reflecting connection policy.

Service Type	Approach	Description
Single Selection	Load Balancing Fail Over	-Select one optimal service considered current environment in service group -Judge with Load Balancing policy and service utilization information
Multi Selection	Plug-in	-Invoke all services in service group -Dispatch newly registered services without modification of client

Table 1 Service Types and Approach

3.2 Services Approach

Load balancing, approach considered in single selection, considers usage type of ser-vices and constructed environment. There are three kinds of Load Balancing techniques: Round-Robin (RR), Weighted Round-Robin (WRR), and Least Connection Scheduling (LC). Round-Robin is accomplished by allocating statically without considering status of server or network and it is efficient when the specification of servers is same. Weighted Round-Robin gives much weight on better servers so that these servers can dispatch more works. Both of them can reduce load of balancing due to construct static environment and dispatch request of service, but they can cause load unbalancing as the time goes on. Least Connection Scheduling manages connection information of server about service request and makes load lessen by connecting to server which has least connections. However, for reliable service approach additional considerations such as response time of service request or aging information except standard Load Balancing policy are needed. This paper improves the existing policy algorithm by adding management of necessary service status information and log data, and utilizes it through allocating each service group. These information is decided when they are deployed in service connection management system and services are decided by the result. The service allocation algorithm by service request of clients is shown at fig. 3.

```
INPUT group
BEGIN
  Service[] services = getAvailableServices(group);
  type = getType(group); // get Group type for request
  if(type == single){
    // select one service from serivces group based on a
policy
    Service properService;
    rtype = getLoadBalancingType(group)
    if(rtype == RR){ // round robin type
      properService = getServiceByRoundRobin(services)
    }else if(rtype == WRR){
      properService =
            getServiceByWeightRoundRobin(services)
    }else if(rtype == LC){
      properService =
        getServiceByLeastConnectionScheduling(services)
    }
    addConnectionList(properService);
    setServiceStartTime(properService);
    setAgingToZero(peroperService);
  }else if( type ==multi){
    // select multi services from service group
    Serivce[] services = getServicesByMultiType(group)
    return services
  }
END
```

Fig. 3 Algorithm to Request Service from Group Service based on a policy

The above algorithm shows process of Web service allocation by service request of clients through connection management system. When clients request services, con-nection manager looks up all services at the current time in the available group, and decides how to dispatch particular service using type (single/multi) information of the group. If the type is single, manager will search optimal services based on load bal-ancing of the registered service groups and then will transmit the services to clients. In this process, the connection manager manages connection information of invoked service, start and end time of service and aging information. The information is util-ized for service selection with existing log information during the process of each load balancing policy. In the case of multi type, all of currently available service list from the service lists that are registered at connection manager is transmitted to clients.

4 System Implementation

The environment of implementation is based on EJB and includes management of service group, maintenance of status information about service request and policy management. This section presents architecture of connection management system, service Deploy Tool and operation process.

4.1 Web services Connection Management System Architecture

Web service Connection Management System architecture consists of service group using Deploy Tool and utilizing service group of clients. Fig. 4 shows architecture of connection management system.

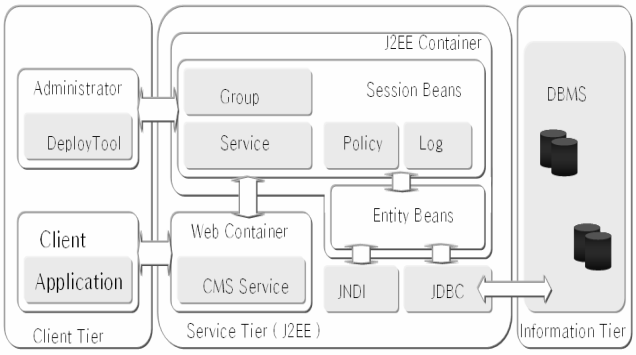

Fig. 4 Web services Connection System Architecture

As shown in Fig. 4, Deploy Tool has a purpose of deploying and managing services per group and manages policy configuration and information management of services. Additional particulars for deployment are presented in section 4.2. Utilization of Web service per group is performed by clients and CMS (Connection Management Service) Service is operated on Web service. Service request and usage procedure are managed by session bean and it manages connection information, error occurrence and request time of requested service, and dispatched results which is completed after performing service are stored as log information. This log information will utilize as a selection decision data of service request.

Policy Session Bean is used for providing reliable service of service request to clients. The Bean provides reliable service concerning with policy information of registered service group by Deploy Tool, collected log data at the process of providing services and service connection information at the current time. Also, service groups can operate under lower coherence, because service connection based on JNDI is offered.

4.2 Deploy Tool

Deploy Tool for Web service group is responsible for registration of Web service per web group, group policy and information maintenance and registered group provides services clients by service connection management system and also monitors current status information of registered service at group. Fig. 5 shows composition factors of Deploy Tool. The tool composites CMS server, connection part, Web service group, Web service registration part and monitors part of registered service.

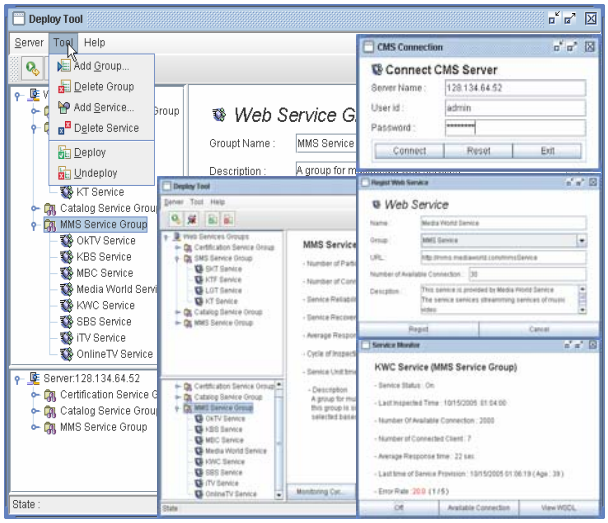

Fig. 5 Web service Deploy Tool per groups

First of all, service group registration configures JNDI name used at WSDL document and program concerned with particular service and decides single or multi service type at a policy. In the case of single type, applicable algorithms are selectable. Provided algorithms are RR, WRR and LC. Identification of Web service list information registered in particular group is possible. Services registered in service group decide ser-vice use by group policy and then allocate to clients.

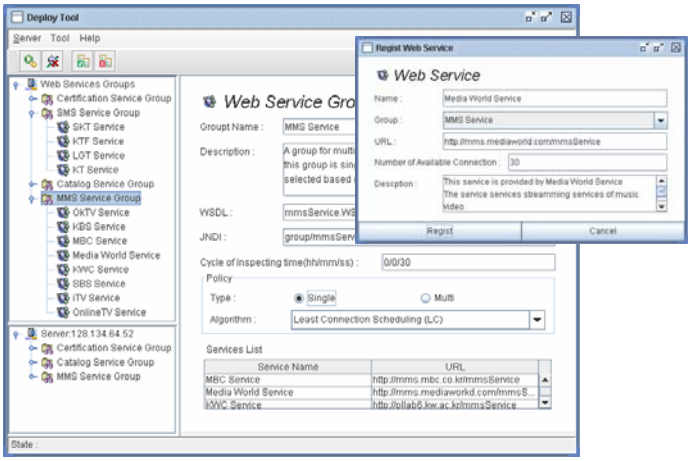

Fig. 6 Service group and service registration

Services which are registered at the services can monitor status of current time.
After registering service following this procedure, we can figure out the status of using services by monitoring service connection information, average response time and recent use list of request of particular service. Fig. 7 shows registered groups and use list of each service.

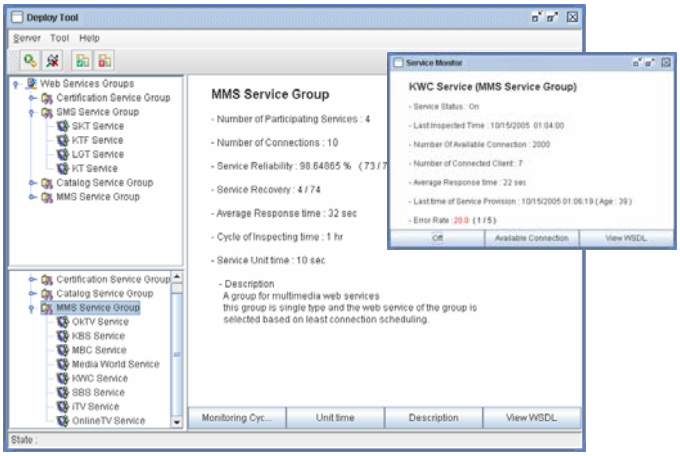

Fig. 7 Registered group and service monitoring

5 Conclusion and Future Work

There are various researches for Web service construction and utilization: quick acceptance of SOA based business environment, necessary interoperation problem for application, evaluation of service itself quality and service search. However, quality of Web service mainly depends on provider of services and usually is served in the only static way. Nevertheless, there are many Web services which provide same services and their efficient utilization is very important factor for improving quality. This paper proposes dynamic service connection system by service pool for efficient utilization of Web service and quality improvement. To archive it, we group and manage Web ser-vices which provide same service and construct system which provides reliable ser-vice based on status information of service and a policy. Therefore users are not only provided more stable and reliable service but also offer efficient service management environment. We need researches for solving service cost problem by service group.

References:

[1] Hammouda, I., Koskimies, K., *"Generating a pattern-based application development envi-ronment for Enterprise JavaBeans,"* Computer Software and Applications Conference, 2002.

[2] Jaiganesh Balasubramanian, Schmidt, D.C., Dowdy, L.; Othman, O, *"Evaluating the per-formance of middleware load balancing strategies,"* Enterprise Distributed Object Comput-ing Conference, 2004

[3] Judith M. Myerson, *"Gurantee your Web service with an SLA"*, IBM developerWorks, April, 2002.

[4]Kalepu, S., Krishnaswamy, S., Loke, S.W., *"Verity: a QoS metric for selecting Web services and providers,"* Web Information Systems Engineering Workshops, 2003

[5] Korea National Computation Agency, *"A Study on Technical Trends and Deployment Strategies of Web service Quality Management,"* Research Report, 2003.

[6] Macaire, A, *"An open and secure terminal infrastructure for hosting personal services,"* Technology of Object-Oriented Languages, 2000

[7] M.P. Papazoglou, *"Service-Oriented Computing: Concepts, Characteristics and Directions,"* Proc.

of 4th Int'l Conf. on Web Information System Engineering, 2003.

[8] Oasis Committee. *Universal Description, Discovery and Integration (UDDI),* Version 3 Specification. http://www.uddi.org. 2002

[9] Schmietendorf, A., Dumke, R., Reitz, D., *"SLA management - challenges in the context of Web-service-based infrastructures, "* Web services, 2004

[10] W3C Web services Activity, http://www.w3.org/2002/ws

[11] W3C. Simple Object Access Protocol (SOAP) 1.2, W3C Candidate Recommendation. http://www.w3.org/TR/soap12-part0/, 2002

[12] W3C. Web services Description Language (WSDL) 1.1, W3C Note. http://www.w3.org/TR/wsdl (W3C Working Draft for version 1.2 is available at http://www.w3.org/TR/wsdl12 , 2001