

Enhancing OSGi Framework for Enterprise Distributed Systems

Jae-Hyun Choi¹, Jea-Won Park² and Jong-Bae Kim^{3*}

^{1,2,3*} Graduate School of Software, Soongsil University, Seoul 156-743, Korea
¹jaehyun@ssu.ac.kr, ²jwpark@ssu.ac.kr, ^{3*}kjb123@ssu.ac.kr

Abstract

Recently, OSGi has been used in many areas of smart phones, software applications, and home network. However, it cannot be used for enterprise distributed systems because it does not have auto-managing schema for them and does not provide a backup strategy. A service location protocol which is a dynamic configurations protocol can be used to auto-manage the distributed systems, but the protocol cannot ensure the stability of distributed systems. In this paper, we present a technique of automated service discovery in distributed systems based on OSGi. The proposed technique defines four components and four type profiles to achieve automated service discovery and backup functions. By using the technique, OSGi could be widely used for developing enterprise distributed systems.

Keywords: OSGi, Backup Strategy, Service Discovery

1. Introduction

Recently, a Java-based service platform OSGi (Open Service Gateway Initiative) [1, 4] has been used in many areas including mobile phones, embedded applications, and home area networks [4, 5]. Although, it is not suitable for enterprise areas because it does not support for auto-managing distributed systems and does not provide a backup strategy. A service location protocol [8, 9] as a dynamic configurations protocol can be used to auto-manage the distributed systems, but the protocol cannot ensure the stability of distributed systems because it does not provide a correct strategy to backup a service provider and it only is used in the local network.

In this paper, we present a technique of automated service discovery in distributed systems based on OSGi. It also defines a backup schema at software level. The proposed technique defines four components: discovery implement, actor admin implement, export services admin implement, and topology admin implement. Also, it defines four type profiles: two profiles about the service and two profiles about the service provider, and six type messages. These implementations have been driven by the six type messages to achieve auto-discovery and backup functions. Lastly, we verify the functions and execution efficiency of this technique through an experimental scenario. The results of the experiment show that the functions are reliable. The proposed technique of automated service discovery will be widely used for developing distributed systems based on OSGi.

2. Related Works

OSGi has been defined as a standardized, component-oriented, computing environment for networked services that is the foundation of an enhanced service-

* Corresponding author. Tel. : +82-10-9027-3148.
Email address: kjb123@ssu.ac.kr(Jong-Bae Kim).

oriented architecture, and OSGi platform has a layered model that includes execution environment, security, modules, life-cycle, service, and bundle. Services can communicate each other and make service community for more special purpose on OSGi. The OSGi provides a modular runtime environment which has predefined boundary. Services are discovered and used in that boundary only. This implication may not have been a problem in local area network like a home network. However, it can be large problem in distributed computer system that the services interaction is more important across the platform. Therefore, we believe that there is a need to solve this problem for using OSGi in distributed systems.

In other researches, Service Location Protocol(SLP) has been used to solve this problem [8, 10, [11]. The SLP defines some type of message to communicate and two organization forms. The first form is discovering services without a Directory Agent (DA). In this form a DA is not present on the network; UAs communicate their requests indirectly to Service Agents (SAs). The second form is discovering services with a DA. In this form a DA is present in the network; User Agents (UAs) and SAs communicate with DA directly for service requests and advertisements. However, SLP has three shortcomings for using distributed systems base on OSGi. Firstly, there is not a backup scheme for DA. Secondly, there is not a backup scheme for SA. Lastly, there is only operating in local area networks. Therefore, new approach is needed for using OSGi framework in distributed systems.

3. Design of the Service Discovery and Backup Technique

The technique bases on the multicast protocol, and establishes the relationship in the distributed systems based on OSGi. The endpoints can send or receives the member-to-group and member-to-member messages using a multicast group address (an Internet Protocol address in the range 224.0.0.0 to 239.255.255.255) or sign cast address. The technique defines the four actors that are AE(Admin Endpoint), BAE(Backup Admin Endpoint), AE-Other(the common service provider) and BE(Backup of AE-Other). Table 1 shows the actors defined by the technique.

Table 1. Actors in the Technique

Actor	Description
AE	The manager of a distributed system.
BAE	The backup the AE.
AE-Other	The service provider
BE	The backup of service provider.

As a dynamic discovery technique it provides dynamic schema for discovering the endpoint (service provider) and service by attributes rather than by name or address. It is the endpoint and the service discovery method for TCP/IP (Transmission Control Protocol/Internet Protocol) based communications. It is a simple and lightweight technique for automatic advertisement and maintenance the system and minimization the communication flux of the network. When the system has been created, the AE/BAE election process will run and the AE and BAE will be elected for a system. Only the AE will be viewed by the user application. The technique uses the multicast technique to achieve the suppleness and lightweight. When a new endpoint joins the system, it sends a message to a group address – Announce address by the active mode and the all endpoints have been recovered the message and registers the endpoint into the Endpoint Table by the way the

endpoint of the system should know the change of environment. When an endpoint has been removed from the system, the other endpoints will find it by listening to the Announce address. So it provides a dynamic configuration mechanism of the system. It also uses a extend attribute – BackupEndpoint to provide an endpoint backup mechanisms

In this scenario, the technique will be explained. There are four main steps that it has to establish. The first step is that obtaining information about the distributed systems, the second step is that is electing the AE and BAE, and the third step is monitoring the system a dynamic configuration when changing. The last step is the backup module.

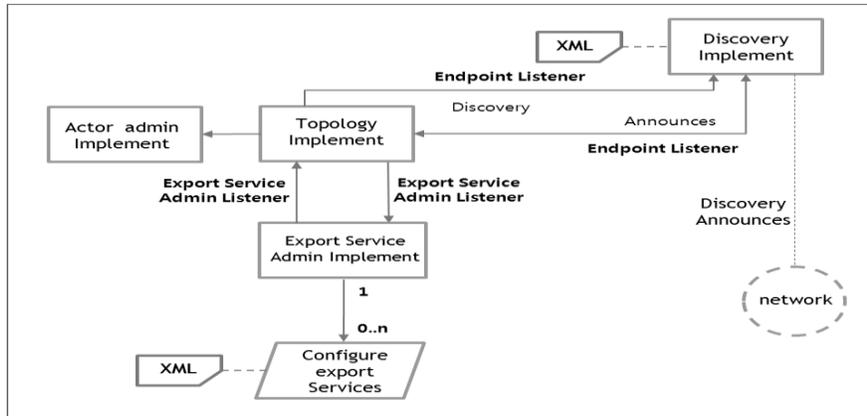


Figure 1. Work Principle

Figure 1 shows the features and running way of the technique. The discovery implement first announces its endpoint information and receive the others information which is used by the topology implement that maintenance the topology of the system, and it also is used by the actor admin implement that decide the endpoint’s actor. Second it announces its export service information and receives others export services information which is used by the export service admin implement that manages the export service.

3.1. Endpoint Relationship

Before communication between members, the OSGi must discovery their peer endpoints and establishes the relationship with each other. The information of endpoints is recorded in Endpoint table by each endpoint. It is used for the maintenance of the relationship. Figure 2 shows the content about table and that will be formalized later.

XML			
Endpoint_Table			
Endpoint (6)			
	EndpointID	EndpointIP	EndpointState
1	Center-1	192,168,0,10	AE
2	Center-2	192,168,0,11	BAE
3	Branch1-1	192,168,0,12	AE-Other
4	Branch1-2	192,168,0,13	BE
5	Brnach2	192,168,0,14	AE-Other
6	Brnach3	192,168,0,15	AE-Other

Figure 2. Work Principle

The technique defines some attributes of the endpoint; it includes EndpointID, IPAddress, SystemID, Priority, AE, BAE, AnnouceInterval, EndpoinDeadInterval, WaitInterval,

AuthenticationKey, EndpointList, and BackupEndpoint. These are recorded in the file of EnpointInfor.XML. Figure 3 shows the content of it.

EndpointInformation	
EndpointID	Center-1
IPAddress	192.168.0.10
ClusterID	cluster01
Priority	80
AE	
BAE	
AnnounceInterval	5
EndpointDeadInte...	15
WaitInterval	15
AuthenticationKey	123456
EndpointList	
BackupEndpoint	

Figure 3. Work Principle

EndpointID and IPAddress are property of the Endpoint. The tracking of other endpoints requires that each endpoint have an EndpointID and IPAddress by each endpoint and the EndpointID is uniquely indemnified within the OSGi domain. SystemID which the endpoint is belongs to.

Priority of the endpoint is base of AE and BAE election. AE, BAE is record the Endpoint ID of AE and BAE, and the default value is EndpointID himself. The attributes AnnounceInterval and EndpointDeadInterval are period in seconds. AnnounceInterval-between transmissions two announce message in the system. EndpointDeadInterval – the endpoint declare a peer endpoint down in which it does not receive the announce message of the endpoint. WaitInterval – the length time of the endpoint will wait before its AE and BAE election begin. AuthenticationKey is a password to join the system. Only the verification if successful, the announce message of the endpoint will be accepted by peer endpoints of the system. EndpointList is a list of its all valid peer endpoints. BackupEndpoint is the tag of backup endpoint, if the value is not empty it is a backup endpoint and the value is the EndpointID of backup to endpoint.

The announce message has been used for communication for endpoints in the system. The technique defines some information the message like Table 2.

Table 2. Content of the Announce Message

Content	Description
EndpointID	The endpoint ID of the origination endpoint
IPAddress	The IP Address of the origination endpoint
SystemID	The system ID of the origination endpoint
AE	The endpoint ID and IP of the AE
BAE	The endpoint ID and IP of the BAE
AuthenticationKey	The authentication type and information of the origination Endpoint
AnnounceInterval	The announce interval time of the origination endpoint.
EndpointDeadInterval	The endpoint dead interval time of the origination endpoint
Priority	The priority of the origination endpoint.
EndpointList	The list of the endpoints that can be communicated.
BackupEndpoint	The backup of service provider.

The announce message has some purpose. It is used to fond endpoint, and it advertises parameters before the endpoint communication with other endpoints. The announce message

acts as keep live between endpoints, and it can be used to select AE and BAE. Endpoint periodically sends an announce message to peer endpoints. The period is `AnnounceInterval` and its default values are 5 sec. If an endpoint has not received the announce message from the endpoint within a period time then it will declare down, that period is the `EndpointDeadInterval` and its default values are three times of `AnnounceInterval`. If matching the announce message is declare valid an if the endpoint ID of originating endpoint is already listed in the endpoint table, the dead-interval timer is rest, and if the endpoint ID is not listed, it will be added to endpoint table. If an OSGi receives a valid announce message in which it finds its own endpoint ID is listed. The relationship with tow endpoints is two-way. The two-way relationship is a two-sway virtual channel. The announce message transmit by the channel. Figure 4 shows the relationship.

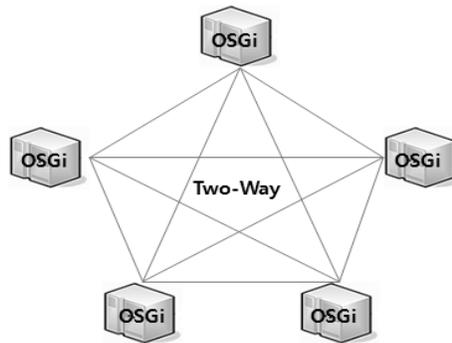


Figure 4. Two-way Relationship

Using multicast to sent message about service has two problems for the distributed systems. One the formation of a two-way between every endpoint would great may necessary message. If there are n pieces of endpoints in the system, there would be $n(n-1)/2$ adjacencies, and each endpoint has $(n-1)$ two-way. Another flooding in the system itself would be chaotic. An endpoint would flood a message to all its peer endpoints, which would create may copy of the same message in the system. The relationship of adjacency is establish the endpoint (AE) with others (AE-Others). The adjacency relationship is a unidirectional channel. The AE-Other send the message about service transmit by the channel.

3.2. AE and BAE

The AE has been selected in the system to resolve the two problems. The AE has following two duties, first reducing the adjacencies relationship with all endpoint in the system and second controlling the flooding of the OSGi application network.

The concept of AE is that the relationship of it and each other endpoints in the system if adjacency and only the AE sends announce service update message to the system. A significant problem with the AE scheme as described so far is that if the AE fails, a new AE must be elected. New adjacencies must be established, and all endpoints on the application network must synchronize their databases with the new AE. All endpoints (AE-Others) have the adjacencies not only with the AE but also with the BAE. The AE and BAE also become adjacent with each other. If the AE fails, the BAE becomes the new AE. Because the other endpoints in the system are already adjacent with the BAE, the system unavailability is minimized.

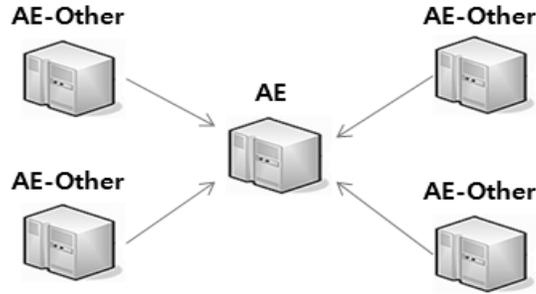


Figure 5. Adjacency Relationship

The election of the AE and BAE is triggered by the endpoint state, which will be described in a later section. For the election process to perform properly, the following preconditions must exist. First, each endpoint has a priority, which is an unsigned integer ranging from 0 to 99. The default priority is 0 and can be changed. An endpoint with a priority of 0 is ineligible to become the AE or BAE, and the backup endpoint (its priority is 0) also is ineligible. An announce message includes SystemID for the originating endpoint which belongs to and specifies its priority, IP address, AE ID and BAE ID. When an endpoint first becomes active in the system, it sets the AE and BAE to itself EndpointID, and also sets a wait timer with a value equal to the WaitInterval. The election process of the AE and BAE is as follows

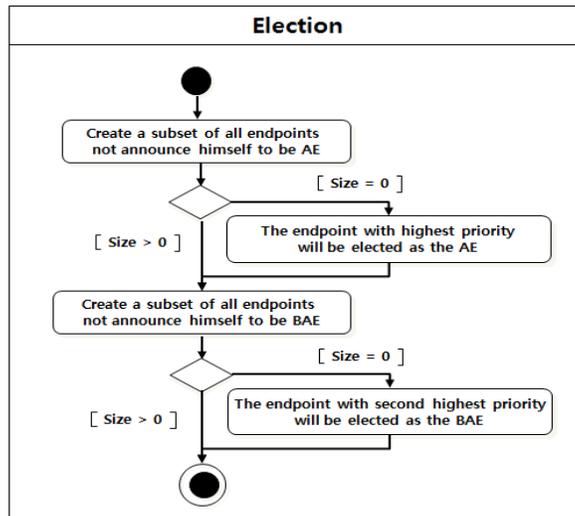


Figure 5. Election Process

After Two-Way relationship with every endpoints in the endpoint table has been established, and the WaitInterval has come on. To list all endpoints for election which priority is not 0, create a subset of all endpoints do not claim himself as AE (their own EndpointID in the AE field of announce message) in the list. If one or more endpoints are in this subset, the endpoint with the highest priority will be elected as the AE. If not only endpoint have the highest priority, the endpoint with the highest EndpointID will be chosen. If the subset is empty, the system has the AE already, and then this endpoint accepts and sets it. Create a subset of all endpoints do not claim himself as BAE (their own EndpointID in the BAE field of announce message) in the list. If one or more endpoints are in this subset, the endpoint with

the second highest priority will be elected as the BAE. If not only endpoint have the second highest priority, the endpoint with the highest EndpointID except AE will be chosen. If the subset is empty, the system has the BAE already, and then this endpoint accepts and sets it.

Explaining in simple language, when an endpoint becomes active and discovery its peer endpoints, it checks for an active AE and BAE. If an AE and BAE exist, the endpoint accepts them. if there is no AE, an election is held in which the endpoint with the highest priority becomes the AE, then BAE will be elected. If more than one endpoint has the same priority, the one with the numerically highest EndpointID wins.

If there is no active AE, the BAE is promoted to AE and a new election is held for the BAE. It should be noted that the priority can influence an election, but will not override an active AE or BAE. That is, if an endpoint with a higher priority active after an AE or BAE has been elected, the new endpoint will not replace either of AE or BAE. Once the AE and BAE have been selected, the other endpoints will establish adjacencies with the AE and BAE only. All endpoints continue to multicast announce using the address-announce address, the AE will flood the message of update service to AE-others endpoint using this address. AE-others endpoints to multicast message about service using the address-AE/BAE address, only the AE and BAE will listen to this address. If only one eligible endpoint (others priority is 0) is attached in the system, it will become the AE and there will be no BAE. Any other endpoint will form adjacencies only with the AE. If none of the endpoint attached, there will be no AE or BAE and no adjacencies will be establish. The endpoint states of all endpoints will remain Two-Way.

3.3. Endpoint State

The endpoint will transition through several states before it becomes balance state. Those states are Down, Waiting, AE, BAE, AE-Others like Table 3 and Figure 6 shows the endpoint states transition.

Table 3. Endpoint States

Content	Description
Down	The endpoint does not start at this state.
Initialization	This state is the initial state. The endpoint is not functional; all endpoint parameters are set to their initial values, and no traffic is transmitted or received.
Waiting	When the endpoint transitions to this state, it will begin sending and receiving announce message. The endpoint will attempt to identify the AE and BAE while in this state.
AE	In this state, the endpoint is the AE in the system and the adjacencies will establish with other endpoints.
BAE	In this state, the endpoint is the AE in the system and the adjacencies will establish with other endpoints (except the AE)
AE-Other	In this state, the endpoint is neither the AE nor the BAE in the system. It will form adjacencies only with the AE and BAE and register the service to AE and BAE.
BE	In this state, the endpoint is the backup for one of AE-Other.

When the endpoint is in the waiting state, it first checks the BackupEndpoint attributer. If it is the backup, the endpoint state will transition to BE, otherwise when the election process is triggered; it will transition to AE or BAE or AE-Other by result of election. The BE state can transition to AE-Other, but it can transition to AE or BAE state. The AE state only transition the Down state and the BAE state transition the Down state or BE state. It ensure the AE or

BAE does not be replaced, when new endpoint that has the higher priority than AE. Figure 6 shows the states transition.

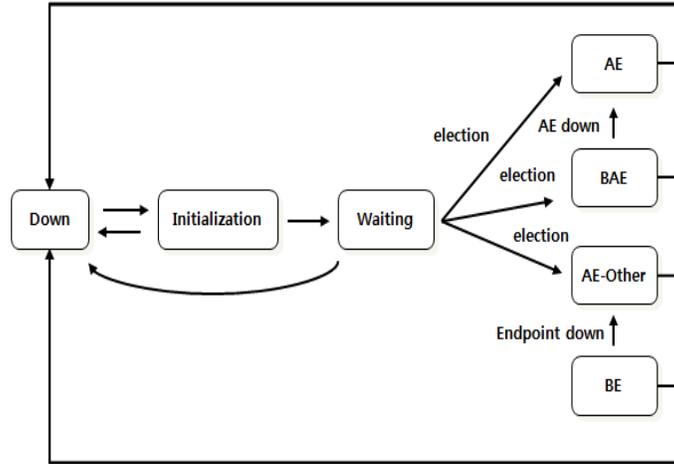


Figure 6. Endpoint State Transition

3.4. Service Synchronization

ServiceRegister, ServiceChange, and ServiceUpdate message are exchanged to ensure that both service providers have the same service information in their service description table - ServicesTable.XML. This section explains the service synchronization process that includes the service register and service change process. When the election of AE/BAE process has finished, service register process will run. The endpoint sends the service description list to the group address - AE/BAE address (the AE and BAE will be listening). When AE receive endpoint's service register message, it will response the ACK (acknowledge character) message and sent service update message to all endpoints. In the process the endpoint send ServiceRegister message and AE will send the RegisterACK message to response and send the ServiceUpdate message. Figure 7 shows the process.

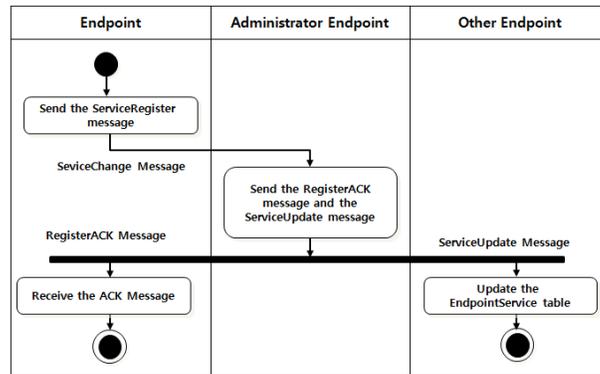


Figure 7. Service Register and Synchronization Process

When the service of the endpoints has been changed, the endpoint will send the ServiceChange message to the group address - AE/BAE address (will be listening by the AE and BAE). When AE receive endpoint's service register message, it will response the ACK message and sent service update message to all endpoints. Figure 8 shows the process.

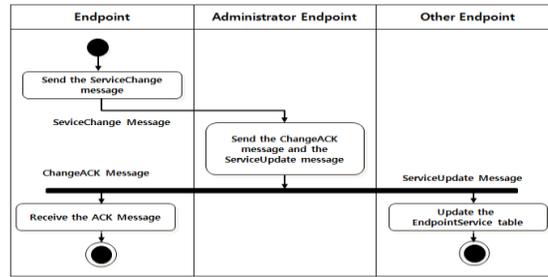


Figure 8. Service Change and Synchronization Process

3.5. Backup Schema

The technique provides service provider backup schema. The backup provider monitors the endpoint which it backup to. It will be active when its monitoring provider leaves the distributed systems. Figure 8 shows the backup process.

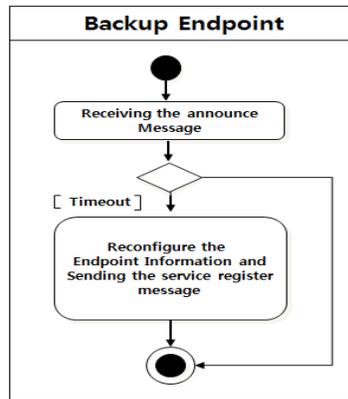


Figure 8. Backup Endpoint Process

3.6. Messages of the Technique

The technique defines some type message for communication. It includes Announce message, ServiceReigster message, ServiceChange Message, RegisterACK message, and ServiceUpdate message like Table 4.

Table 4. Messages of the Technique

Message Type	From	To	Description
Announce	Endpoint	Announce address (239.1.1.1)	It has been used to discovery the endpoints and maintenance the topology of the system.
ServiceRegister	Endpoint	AE/BAE address (239.1.1.2)	The endpoint uses it to register service.
ServiceChange	Endpoint	AE/BAE address (239.1.1.2)	The endpoint uses it to change service.
ServiceUpdate	AE	Announce address (239.1.1.1)	The AE send the update info for endpoint by this message.
RegisterACK	AE	Endpoint	The ACK message for service register.
ChangeACK	AE	Endpoint	The ACK message for service change.

4. Verification of the Discovery and Backup Technique

The technique provides stability, scalability, and auto-management functions in the distributed systems based on OSGi. It achieves centralized management of OSGi platforms and realizes the distributed intelligent network management based on OSGi platforms.

4.1. Environment

The verification environment includes network, hardware and software. The network and hardware and software all support the multicast, and the hardware is personal computer that has 32 bits x86-based processor. The software includes operation software, Java software development kit, and OSGi framework. Table 5 shows the environment.

Table 5. Verification Environment

Type	Environment
Network	Local Area Network
Hardware	PC (AMD Althorn 4200 / 2G)
Software	Windows 7 Java 1.6.2, OSGi 4.2

Figure 9 shows a distributed system based on OSGi platform and it includes a center and the three branches and the branch 1 has a backup. The application uses the technique to manage the network topology and dynamic synchronize services in the system. An endpoint is an OSGi platform and all endpoint provide three service that are GetStock, GetRecord, and GetAccount services.

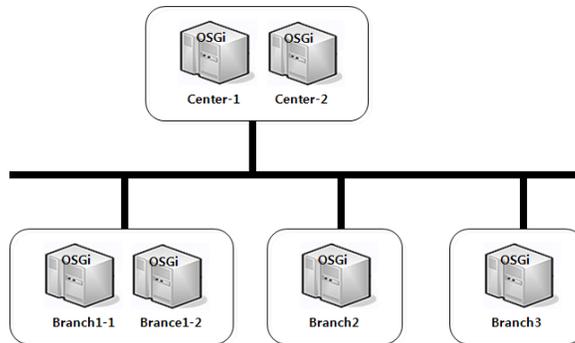


Figure 9. Topology of Test

4.2. Experiment and Data Analysis

The technique bundle will install in center endpoints and all branch endpoints, and then the all endpoints will set the files - Endpointinfo.xml, EndpointService.xml, EndpointsTable.xml and the ServicesTable.xml. The endpoint Center-1 has the highly priority and the Center-2 has the second highly priority and the Branch1-2 is the backup of the Branch1-1.

Test 1: Testing the functions of the technique.

In this test, first of all, we start the all endpoints and then inspect the configuration file EndpointsTable.xml of Center-1 and ServicesTable.xml.

XML			
Endpoint_Table			
Endpoint (6)			
	EndpointID	EndpointIP	EndpointState
1	Center-1	192.168.0.10	AE
2	Center-2	192.168.0.11	BAE
3	Branch1-1	192.168.0.12	AE-Other
4	Branch1-2	192.168.0.13	BE
5	Branch2	192.168.0.14	AE-Other
6	Branch3	192.168.0.15	AE-Other

Figure 10. Initial Configuration File (EndpointsTable.xml)

XML							
ExportService							
Endpoint (5)							
ID	Service	ServiceName	Input	Output	Description	Path	
1	Center-1	Service (3)	1	GetStock	String	String	Center-1:Center-1:GetStock
			2	GetRecord	String	String	Center-1:Center-1:GetRecord
			3	GetAccount	String	String	Center-1:Center-1:GetAccount
2	Center-2	Service (3)	1	GetStock	String	String	Center-2:Center-2:GetStock
			2	GetRecord	String	String	Center-2:Center-2:GetRecord
			3	GetAccount	String	String	Center-2:Center-2:GetAccount
3	Branch1-1	Service (3)	1	GetStock	String	String	Branch1-1:Branch1-1:GetStock
			2	GetRecord	String	String	Branch1-1:Branch1-1:GetRecord
			3	GetAccount	String	String	Branch1-1:Branch1-1:GetAccount
4	Branch2	Service (3)	1	GetStock	String	String	Branch2:Branch2:GetStock
			2	GetRecord	String	String	Branch2:Branch2:GetRecord
			3	GetAccount	String	String	Branch2:Branch2:GetAccount
5	Branch3	Service (3)	1	GetStock	String	String	Branch3:Branch3:GetStock
			2	GetRecord	String	String	Branch3:Branch3:GetRecord
			3	GetAccount	String	String	Branch3:Branch3:GetAccount

Figure 11. Initial Configuration File (ServicesTable.xml)

Endpoints in the system have reached balance state. The endpoint Center-1 finds the all endpoints in the system, and registers all services of the system. The Center-1 that has the highly priority is the AE, and the Center-2 that has the second highly priority is the BAE of the system. And Branch1-2 is the backup endpoint. Then we shutdown the endpoint Branch3 and inspect the configuration file EndpointsTable.xml of Center-1 and ServiceTable.xml.

XML			
Endpoint_Table			
Endpoint (5)			
	EndpointID	EndpointIP	EndpointState
1	Center-1	192.168.0.10	AE
2	Center-2	192.168.0.11	BAE
3	Branch1-1	192.168.0.12	AE-Other
4	Branch1-2	192.168.0.13	BE
5	Branch2	192.168.0.14	AE-Other

Figure 12. Configuration File (EndpointsTable.xml) after shutdown Branch3

XML							
ExportService							
Endpoint (4)							
ID	Service	ServiceName	Input	Output	Description	Path	
1	Center-1	Service (3)	1	GetStock	String	String	Center-1:Center-1:GetStock
			2	GetRecord	String	String	Center-1:Center-1:GetRecord
			3	GetAccount	String	String	Center-1:Center-1:GetAccount
2	Center-2	Service (3)	1	GetStock	String	String	Center-2:Center-2:GetStock
			2	GetRecord	String	String	Center-2:Center-2:GetRecord
			3	GetAccount	String	String	Center-2:Center-2:GetAccount
3	Branch1-1	Service (3)	1	GetStock	String	String	Branch1-1:Branch1-1:GetStock
			2	GetRecord	String	String	Branch1-1:Branch1-1:GetRecord
			3	GetAccount	String	String	Branch1-1:Branch1-1:GetAccount
4	Branch2	Service (3)	1	GetStock	String	String	Branch2:Branch2:GetStock
			2	GetRecord	String	String	Branch2:Branch2:GetRecord
			3	GetAccount	String	String	Branch2:Branch2:GetAccount

Figure 13. Initial Configuration File (ServicesTable.xml) after Shutdown Branch3

When the Branch3 leave the system, the endpoint Center-1 finds the change of the topology and reconfigures EndpointsTable.xml, and then unregistered the service of the

Banch3. Then After, we shutdown the endpoint Branch1-1 and inspect the configuration file EndpointTable.xml of Center-1 and ServiceTable.xml.

ID	EndpointID	EndpointIP	EndpointState
1	Center-1	192.168.0.10	AE
2	Center-2	192.168.0.11	BAE
3	Branch1-2	192.168.0.13	AE-Other
4	Brnach2	192.168.0.14	AE-Other

Figure 14. Configuration File (EndpointsTable.xml) after shutdown Branch1-1

ID	EndpointID	ServiceName	Input	Output	Description	Path
1	Center-1	1 GetStock	String	String		Center-1:Center-1:GetStock
		2 GetRecord	String	String		Center-1:Center-1:GetRecord
		3 GetAccount	String	String		Center-1:Center-1:GetAccount
2	Center-2	1 GetStock	String	String		Center-2:Center-2:GetStock
		2 GetRecord	String	String		Center-2:Center-2:GetRecord
		3 GetAccount	String	String		Center-2:Center-2:GetAccount
3	Branch1-2	1 GetStock	String	String		Branch1-2:Branch1-2:GetStock
		2 GetRecord	String	String		Branch1-2:Branch1-2:GetRecord
		3 GetAccount	String	String		Branch1-2:Branch1-2:GetAccount
4	Branch2	1 GetStock	String	String		Branch2:Branch2:GetStock
		2 GetRecord	String	String		Branch2:Branch2:GetRecord
		3 GetAccount	String	String		Branch2:Branch2:GetAccount

Figure 15. Initial Configuration File (ServicesTable.xml) after shutdown Branch1-1

When the endpoint Branch1-1 leave the system, the endpoint Center-1 finds the change of the topology and reconfigures EndpointsTable.xml. it unregistered the services of the Branch1-1 and register services of the Brnach1-2 that is the backup of the Branch1-1.

Test 2: Service synchronizes time of the distributed system with increasing a number of the service provider.

In this test, the synchronize time of the system has been recorded when the services provider increase from 5 to 40 (five provider offer 15 export services). Figure 16 shows the results.

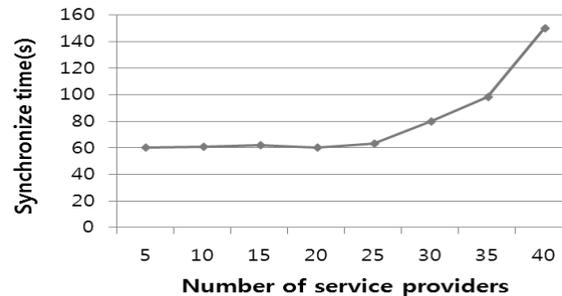


Figure 16. Result of Test 2

Through analysis of the result, when a number of service provider is less 20 the discrepancy of the synchronize time is slight difference. When the number of service provider exceed 20 the synchronize time increase sharply.

Test 3: Service synchronize time of the distributed system with increasing a number of the export service.

In this test, the synchronize time of the system has been recorded when the export services increase from 3 to 10 (the system includes five service provider). Figure 17 shows the result.

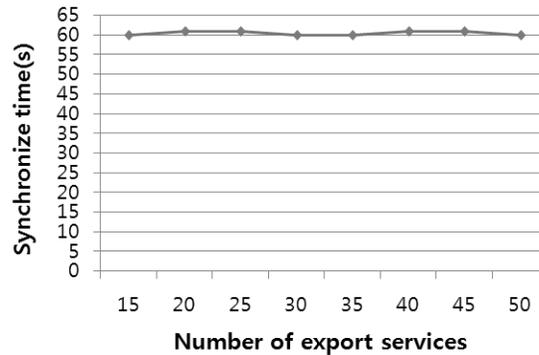


Figure 17. Result of Test 3

Through analysis of the result, when a number of export service increasing, the synchronize time is slight difference.

4.3. Interpretations

The result of experiment shows that the technique can satisfy the design and it realizes expects functions. It can dynamically configure the topology of the distributed systems and synchronize the services, and supply the software level backup schema. It provides AE – the manager of the distributed systems and BAE – backup of AE by election. The execution efficiency of the technique is relevant to a number of the service provider, and when an number of the provider exceeds 20 the execution efficiency is sharp decline. The experiment does not explain how to implement the service in a real scenario. It can implemented as simulate Java RMI. (Remote Method Invocation) technology.

5. Conclusions

This paper presents the technique - dynamic configuring and offering the software backup solution in the distributed systems based on OSGi platform. The technique provides a scalable framework for discovery the service provider and the service. When a new service provider joins or leaves the distributed systems, the system is able to automatically regulate the new provider. It realizes the distributed intelligent network management based on OSGi platform and auto-manage the network topology. It is a dynamic configuration technique for applications operating in local area networks or wide area networks. From the service provider's view, it finds and remains the all export services of the distributed systems, so it can offer the new service that used the existed service of other service providers in the distributed systems, and it can monitor the state of its import services. As the service

discovery technique [12, 13] that research discovering the web services and it does not support the OSGi.

The technique is able to improve the performance, operation, maintenance and stability of the distributed systems. It uses the multicast to reduce the some type messages transaction. It's all configuration file is XML type that make the operation simple and ensure east maintenance. Through the experiment, the technique is powerful enough and able to get the information of the service provider and its services, and it achieves the self-configuration, auto-discovery the service and the service provider in the distributed systems based on OSGi. The backup schema has been proved effective in this experiment. The consequent research is to improve the execution efficiency in large scale distributed systems. With the scale increasing, the noticeable decline in performance. On the other hand how the services dynamic synchronize between different distributed systems based on OSGi is the other study aspect.

References

- [1] J. Choi, J. Kim and J. Park, "Service Discovery and Backup Technique for Distributed Systems based on OSGi", 2014 International Conference on Future Information & Communication Engineering (ICFICE), (2014), pp. 42-45.
- [2] J. Lopez de Vergara, "An autonomic approach to offer services in OSGi-based home gateways", Computer Communications, vol. 31, (2008), pp. 3049-3058.
- [3] K. Kyuchang and L. Jeunwoo, "Extended Service Registry for Distributed Computing Support in OSGi Architecture" February 20-22, ICACT 2006, (2006), pp. 1631-1634.
- [4] D. Marples and P. Kriens, "The Open Services Gateway Initiative: An Introductory Overview" IEEE Communications Magazines, vol. 39, no. 12, (2001), pp. 110-114.
- [5] C. Lee, D. Nordstedt and S. Helal, "Enabling Smart Spaces with OSGi", IEEE Pervasive Computing, vol. 2, no. 3, (2003), pp. 89-94.
- [6] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek and H. Balakrishnan, "Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications", In: Proceedings of the Annual Conference of the Special Interest Group on Data Communication (SIGCOMM) (2001).
- [7] Knopflerfish - Open Source OSGi <http://www.knopflerfish.org>
- [8] E. Guttman, C. Perkins, J. Veizades, M. Day, "Service Location Protocol", Version 2. IETF RFC 2608 (1999)
- [9] J. Kempf and J. Goldschmidt, "Notification and Subscription for SLP. IETF RFC 3082, (2001).
- [10] flowSGi Project, <http://flowsgi.inf.ethz.ch>
- [11] C. Lee, Y. Jongkyu and L. Wonjun, "Bridging OSGi islands through SLP protocol", Ubiquitous Intelligence and Computing. Springer Berlin Heidelberg, (2007), pp. 787-797
- [12] C. Lin, Z. Wu, S. Deng and L. Kuang, "Automatic service matching and service discovery based on ontology, in: Proceedings of the GCC 2004 Workshops, Lecture Notes in Computer Science, vol. 3252, Springer Verlag, (2004), pp. 99-106.
- [13] R. Lara, M. Corella and P. Castells, "A flexible model for locating services on the web International Journal of Electronic Commerce, vol. 12, no. 2, (2007), pp. 11-41.
- [14] I. Foster, C. Kesselman, J. Nick and S. Tuecke, "The Physiology of the Grid: an Open Grid Services Architecture for Distributed Systems Integration", Technical report, Global Grid Forum. (2002).
- [15] National Center for Biotechnology Information, <http://www.ncbi.nlm.nih.gov>.

Authors



Jaehyun Choi, he received the Ph.D. degree in Computer Science from Soongsil University in Korea, (2011). He is a professor at Graduate School of Software, Soongsil University. His research interests are in areas of Data Processing, Service Engineering, Software Engineering, and Text Mining.



Jeawon Park, he received the Ph.D. degree in Computer Science from Soongsil University in Korea, (2011). He is a professor at Graduate School of Software, Soongsil University. His research interests are in areas of Software Testing, Software Process, Web Services, and Project Management



Jong-Bae Kim, he received his bachelor's degree of Business Administration in University of Seoul, Seoul(1995) and master's degree(2002), doctor's degree of Computer Science in Soongsil University, Seoul(2006). Now he is a professor in the Graduate School of Software, Soongsil University, Seoul, Korea. His research interests focus on Software Engineering, and Open Source Software.

