Agent-Based Service Discovery Middleware in Ubiquitous Environments*

Hyung-Jun Kim, Kyu Min Lee, Kee-Hyun Choi, and Dong Ryeol Shin

School of Information & Communication Sungkyunkwan University,
Jangangu Chunchundong 300, Suwon Kyunggi-do, Korea
{mairi, kmlee, gyunee, drshin}@ece.skku.ac.kr

Abstract. Service discovery is becoming a hot topic as it becomes more widespread through the Internet. In pervasive environments, there are various services, and users use a service discovery protocol for finding the most appropriate service among them. Thus, over the past few years, many service discovery protocols have been proposed. Even though they provide convenient service discovery functionality with users, complexity is increased because they have different message exchange pattern and architecture. In this paper, we propose a novel middleware based on agent platform for interoperability among heterogeneous services. This has the Discovery Agent for each service discovery protocols and it can detect service advertisement messages for registration, so the services are registered in agent platform. Finally, user can use them. The key point of this is not to require modification of existing service discovery protocols. Near the conclusion of this paper, this is implemented.

1 Introduction

Over the past few years, lots of service discovery protocols have been proposed. Although they provide the same basic functionality of service discovery, they differ significantly in architecture, message exchange pattern, expected operating environment and service representation/description. These differences make they cannot discover services of other service discovery protocols. In this paper, we propose a new middleware that provides agent-based service discovery to support interoperability using a FIPA-compliant agent platform [1]. The proposed middleware is different from the currently available approaches in that users are easily able to use heterogeneous services through activating discovery agent. It does not require modifying existing service discovery protocols, and uses JADE (Java Agent Development Framework) among diverse agent platforms, which is representative and popular, so it is simple to develop an agent. Other benefits are not only interoperability, but also agent collaboration. The rest of this paper is organized as follows. Section 2 presents the proposed architecture. Section 3 describes operation of our system. In section 4, the implementation is presented and this paper is finally concluded in section 5.

* This research is supported by the ubiquitous Autonomic Computing and Network Project, the Ministry of Information and Communication (MIC) 21st Century Frontier R&D Program in Korea.

2 Proposed Architecture Overview

Fig. 1 shows our new middleware architecture.

Fig. 1. Architecture of our middleware

As illustrated in Fig. 1, there are three parts on the middleware.

Discovery Agent

Discovery Agent receives messages from heterogeneous service discovery protocols through each well-known message port. In other words, there is Message Monitor module for the function, and JADE platform is able to receive service registration messages of UPnP (Universal Plug and Play) [2] and SLP (Service Location Protocol) [3] domain. These messages are converted to DAD used on JADE after parsing and converting process through Discovery Adaptor module.

In fact, Discovery Adaptor analyzes messages and then calls appropriate parser and converter. Namely, if the messages are from UPnP domain, it uses the parser and converter for UPnP. Otherwise, if the messages are from SLP domain, it uses the parser and converter for SLP domain. After conversion, newly created services of DAD type are automatically registered in DF. Remote Invocator is a module that invokes the service User Agent wants to use. For example, it creates SOAP message for invocation, the user agent wants to invoke an UPnP service.

JADE Main Container

If users start up JADE platform, main container is basically created, and DF agent, which is a yellow page, AMS, which is a white page, and Remote Management Agent, which is a JADE management tool, are created and executed. DF stores heterogeneous services that are detected by Discovery Agent. Finally, User Agent can use them by searching DF. All agents on JADE should get AID (Agent Identity Description) from AMS and then is executed.

User Agent

User Agent gets information of UPnP and SLP services that are registered in DF by polling method. It can discover services through the process. Message Manager supports communication with other agents and when an agent is created, a user can see the process on GUI (Graphic User Interface).

3 Sequence Diagram

Fig. 2 shows a sequence diagram of using services of UPnP domain and SLP domain. First, UPnP device and Service Agent of SLP advertise their services. Discovery Agent detects each registry message through well-known ports, parses them and converts them to DAD type. Next, it registers them in DF. After that, User Agent can get the list of heterogeneous services that are in DF. If we want to use them, we can send ACL message to Discovery Agent. Finally, Discovery Agent parses the message, and then creates SOAP message to invoke the services.

Fig. 2. Sequence diagram: Service Registration and Discovery

4 Implementation and Application

In this Section we demonstrate the implementation of the new middleware. Implementation of our middleware is based on JADE 3.3 and each agent is created on 1.4.2_07 version.

The prototyped system is based on UPnP development tool with Intel's UPnP SDK, an open source implementation of the UPnP protocol for UNIX, and Siemen's Java UPnP stack, and SLP development using mSLP supported by Columbia Univ.

We did test DVD Shop, Bank and Hotel services on mSLP and TV device, Washer Device, and Light device on UPnP.

As can be seen in Fig. 3, it shows Discovery Agent and User Agent are on JADE where User Agent indicates a user who tries to find the services in DF.

Fig. 3. JADE RMA and User Agent GUI

5 Conclusion

Existing service discovery protocols have shortcoming because they cannot guarantee interoperability among them. To solve the problem, we propose novel middleware based on agent platform, which supports scalability and generality

Until now, we have implemented our middleware, UPnP and SLP Discovery Agent. In the future, we will develop Discovery Agent for Jini. The most important thing is the middleware supports users to be able to use heterogeneous services in ubiquitous environments without any modification of existing service discovery protocols.

References

- [1] Foundation for Intelligent Physical Agents. *FIPA Agent Management Specification*. June 2002.
- [2] UPnP Forum, "UPnP device architecture 1.0," May 2003.
- [3] E. Guttman, C. Perkins, J. Veizades, and M. Day, "RFC 2608: Service location protocol, version 2," 1999, status: PROPOSED STANDARD.
- [4] M. Rajdou. Software agents in business: Steady adoption curve. Technical report, Forrester Research, USA, 2003.
- [5] Michael Wooldridge. *An Introduction to MultiAgent System*. Wiley and Sons, 2002.
- [6] Michael Luck, Peter McBurney, and Chris Preist. *Agent Technology: Enabling Next Generation Computing*. <http://www.agentlink.org/roadmap>, 2003.