Chapter 3.6
Context-Aware Service Discovery in Ubiquitous Computing

Huaqun Guo
Institute for Infocomm Research and National University of Singapore, Singapore

Daqing Zhang
Institute for Infocomm Research, Singapore

Lek-Heng Ngoh
*Institute for Infocomm Research, A*STAR, Singapore*

Song Zheng
Institute for Infocomm Research, Singapore

Wai-Choong Wong
National University of Singapore, Singapore

INTRODUCTION

The decreasing cost of networking technology and network-enabled devices is driving the large scale deployment of such networks and devices so as to offer many new and innovative services to users in ubiquitous computing. For example, when you carry your mobile laptop or personal digital assistant (PDA) around, or drive on the road, various services have been made available, ranging from finding a local printer to print a file, to instantaneously knowing about the traffic situation from traffic-cameras and other sensors along a highway.

To achieve the above, every participating network-enabled end-device must solve an interesting technical problem, i.e., to locate a particular network service or device out of hundreds of thousands of accessible services and devices. Such service advertising and discovery is important as mobile devices and mobile wireless devices proliferate on networks. For this reason, a service discovery and advertising protocol is an

important tool to help these devices find services on the network wherever they connect, and to let other network users know about the services they are offering.

Context-aware service discovery, on the other hand, would help users to find services that are most appropriate based on fast-changing client conditions, such as location. For example, most laptops are statically configured to print to dedicated office printers. With the help of the context-awareness, a laptop could find the nearest accessible printer attached to the network that the laptop is currently plugged into.

SERVICE DISCOVERY PROTOCOLS

As new services are made available, clients would be able to obtain services based on a set of client-defined keywords or attributes. These attributes would allow the client to narrow the scope of the search so that only those services that pertain to its needs would be discovered. Furthermore, the client would be able to automatically start interacting with the newly discovered services without any programming (Intel).

For the past few years, competing industries and standards communities have been hotly pursuing the technologies for service discovery. E-speak (HP, 2002), UDDI (Universal Description, Discovery, and Integration), Sun's Jini (Arnold, ed. 2001), Microsoft's Universal Plug and Play (UPnP), IBM's Salutation (Salutation Architecture Specification), Service Location Protocol (SLP) of IETF (Guttman, 1999), and Bluetooth's Service Discovery Protocol (SDP) (Helal, 2002) are among the front-runners in this new race. E-speak and UDDI are designed specifically for discovering Web services. On the other hand, Jini, UpnP, and Salutation are geared toward services furnished by hardware devices such as printers, faxes, etc. We elaborate each of these in detail next.

- **E-speak** is an open software platform designed by HP (Hewlett-Packard Co.) to facilitate the delivery of e-services (electronic services) over the Internet. Based on Extensible Markup Language (XML), the E-speak Service Framework Specification makes it possible for e-services to advertise, discover, negotiate, and form contracts, learn each other's interfaces and protocols, and invoke each other's services, all without human intervention.
- **UDDI** protocol is one of the major building blocks required for successful Web services. UDDI creates a standard interoperable platform that enables companies and applications to quickly, easily, and dynamically find and use Web services over the Internet. UDDI is a cross-industry effort driven by major platform and software providers, as well as marketplace operators and e-business leaders within the OASIS (Organization for the Advancement of Structured Information Standards) consortium.
- **Jini** is a distributed service discovery and advertisement architecture that relies on mobile code and leverages the platform independent of Java language. Jini entities consist of services, lookup servers that catalog available services, and clients that require services. All service advertisements and requests go through the lookup servers. To register service availability or to discover services, a service or client must first locate one or more lookup servers by using a multicast request protocol. This request protocol terminates with the invocation of a unicast discovery protocol, which clients and services use to communicate with a specific lookup server. A lookup server can use the multicast announcement protocol to announce its presence on the network. When a lookup server makes an announcement, clients and services that have registered interest in receiving announcements of new lookup services are notified.