Available online at www.sciencedirect.com
SCIENCE DIRECT®
The Journal of Systems and Software 75 (2005) 193–204
The Journal of Systems and Software
www.elsevier.com/locate/jss

Enhancing Service Location Protocol for efficiency, scalability and advanced discovery

Weibin Zhao *, Henning Schulzrinne

Department of Computer Science, Columbia University, 1214 Amsterdam Avenue, Mailcode 0401, New York, NY 10027, USA

Received 18 March 2004; received in revised form 21 March 2004; accepted 3 April 2004
Available online 7 June 2004

Abstract

This paper presents three new mechanisms for the Service Location Protocol (SLP): mesh enhancement, preference filters and global attributes. The mesh enhancement simplifies Service Agent (SA) registrations and improves consistency among Directory Agents (DAs) by defining an interaction scheme for DAs and supporting automatic registration distribution among peer DAs. Preference filters facilitate processing of search results (e.g., finding the best match) in SLP servers (DAs and SAs) to reduce the amount of data transferred to the client for saving network bandwidth. Global attributes allow using a single query to search services across multiple types. These mechanisms can improve SLP efficiency and scalability and support advanced discovery such as discovering multi-access-point services and multi-function devices. We expect that these techniques can also be applied to other service discovery systems.

Keywords: Service discovery; Service Location Protocol; Peer relationship; Full mesh; Registration distribution; Preference filters; Global attributes

1. Introduction

As computing continues moving towards a network-centric model, finding and making use of available services such as printing, display and file sharing in the network becomes increasingly important. To use a service, a device such as a computer or personal digital assistant needs to know the access point [1] of the service, which traditionally depends on a priori knowledge or manual configuration. As more devices are network enabled and more services are available on networks, properly configuring devices for better utilizing available services involves non-trivial administrative overhead. Moreover, administrative configuration becomes difficult or even impossible when devices move from a fixed managed network to a constantly changing or unmanaged network. Consider the following three application scenarios. First, as a mobile device often relies on services provided by other devices, it needs to discover available services when it moves to a new network. Secondly, in an ad-hoc network (e.g., a disaster rescue setting), administrative configuration is unlikely to be possible and effective since devices need to learn about each other dynamically and cooperate. Lastly, for home networks, low cost and ease of use are dominant design considerations, making administrative configuration unsuitable. In recognizing the need to reduce administrative configuration as much as possible and enable automated service discovery, many companies, standards bodies and consortia are actively developing service discovery technology. As a result, various service discovery systems and protocols are emerging in recent years, such as the Service Location Protocol (SLP) (Guttman et al., 1999b), Jini (Waldo, 1999), Universal Plug and Play (UPnP, 2004), Rendezvous (Apple, 2004), Salutation (Salutation, 2004), Universal Description Discovery and Integration (UDDI, 2004), and the Bluetooth Service Discovery Protocol (SDP) (Bluetooth,

* Corresponding author. Tel.: +1-2127490908; fax: +1-2126660140.
E-mail addresses: zwb@cs.columbia.edu (W. Zhao), hgs@cs.columbia.edu (H. Schulzrinne).

[1] In IP networks, a service access point is specified by a tuple: IP address, port number and access protocol (such as FTP or HTTP), which can often be encoded into a URL.

0164-1212/$ - see front matter © 2004 Elsevier Inc. All rights reserved.
doi:10.1016/j.jss.2004.04.011

2004). Although different systems address the service discovery problem at various levels in various ways, they all support the same basic functionality, namely mapping service descriptions or specifications to service access points. By using service discovery technology, a device no longer needs to know its service access points via a priori knowledge, instead it can just specify the characteristics of its desired services, which will be automatically mapped into available service access points in any network that supports service discovery.

One of the widely used service discovery protocols is SLP, which is an IETF (Internet Engineering Task Force) proposed standard for service discovery in IP networks. As more applications (Kempf and Montenegro, 2001; Naugle et al., 2001; Bakke et al., 2003; Zhao and Schulzrinne, 2004c; Poynor, 2001) employ SLP for various discovery purposes, we saw a need to improve SLP efficiency and scalability, and support new discovery scenarios such as discovering multi-access-point services and multi-function devices. In this paper, we present three new mechanisms for SLP: mesh enhancement, preference filters and global attributes. The mesh enhancement simplifies Service Agent (SA) registrations and improves consistency among Directory Agents (DAs) by defining an interaction scheme for DAs and supporting automatic registration distribution among peer DAs. Preference filters facilitate processing of search results (e.g., finding the best match) in SLP servers (DAs and SAs) to reduce the amount of data transferred to the client for saving network bandwidth. Global attributes allow using a single query to search services across multiple types.

The rest of this paper is organized as follows. We first give some background for service discovery and SLP in Section 2. Then we describe three proposed SLP mechanisms: mesh enhancement, preference filters and global attributes in Section 3–5, respectively. Finally, we discuss our implementation in Section 6, give experimental results and their evaluation in Section 7, list related work in Section 8, and conclude in Section 9.

2. Background

2.1. Service discovery

In a service discovery system, a common service description framework is needed for service providers, referred to as servers, and service users, referred to as clients, to describe service characteristics so that they can understand each other properly. In general, each service can be described using a set of attribute-value pairs, with each attribute-value pair specifying one property of the service. There are two ways to organize attributes: a flat structure where all attributes are at the same level, and a hierarchical structure where attributes can be at different levels. For examples, SLP simply puts attribute-value pairs into a list, whereas UPnP and UDDI use XML to describe a hierarchy of attributes. Although Resource Description Framework (RDF, 2004) has been proposed as the service description format for interoperability between service discovery systems (Reynolds, 2001), so far there is no service description standard yet.

While service advertisements from servers usually include all attributes of services, service search requests from clients only include attributes of interest, which may just specify a desired service type or class such as printer, or give additional desired service properties such as color printer and printing speed. In general, any service search request can be specified by a search filter such as those used in SLP and Lightweight Directory Access Protocol (LDAP) (Howes, 1997), which is a logical expression about attributes of interest and their desired values. The matching of a service search request with a service advertisement leads to a discovery. Although service discovery systems bear similarity to web search engines as they both provide matches for search requests, they differ in that service discovery uses attribute-based matching whereas web search uses keyword-based matching, and thus a match in the former has a specific meaning while a match in the latter may have different meanings in different contexts.

Multicast and directories are two widely used mechanisms for service discovery. When only multicast is used, services are discovered in a peer-to-peer fashion in two ways. In passive discovery, servers periodically multicast their service advertisements, and clients listen to these advertisements. Clients compare received service advertisements with their desired service requirements to determine matching services. In active discovery, clients multicast their service search requests, and servers listen to these requests. A server compares received service search requests with its service advertisement; if there is a match, the server unicasts its service advertisement to the client. Although multicast can enable a device to be fully auto-configured (IETF, 2004) in a network segment, it usually cannot scale to a large number of devices, and it is not generally supported in wide-area networks. In the directory-centric service discovery model, directory services accept service advertisements from servers, and answer service search requests from clients. Servers register their services with directories, and clients search services at directories, all using unicast. In order to discover directory services, multicast can be used for an intranet, such as in SLP and Jini, but well-known directories are often assumed for the Internet, such as in UDDI. In addition, Dynamic Host Configuration Protocol (DHCP) (Droms, 1997) can be used to get directory service information in local-area networks (Perkins and Guttman, 1999), and DNS SRV (Gulbrandsen et al., 2000) can be used to obtain