RC 21897 (98516) 30 November 2000 **Computer Science**

IBM Research Report

Advertisement and Discovery of Services (ADS)

William Nagy, Francisco Curbera, Sanjiva Weerawarana
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598

Research Division
Almaden - Austin - Beijing - Haifa - T. J. Watson - Tokyo - Zurich

Advertisement and Discovery of Services (ADS)

William Nagy, Francisco Curbera, and Sanjiva Weerawarana
Component Systems Group
IBM T.J. Watson Research Center

September 22, 2000

1 Introduction

Web services directories accessible to both application developers and applications will be central pieces of the future B2B infrastructure, in much the same way as business directories are critical pieces of today's economies. New standards will emerge to provide universal access to the directories, and most businesses will register their information with well known directory service providers. The transition to this new infrastructure will have to take place at many different levels and at different times.

This document deals with the problem of how to effectively enable the emergence of B2B directories using existing Web infrastructure, and aims at outlining the path for a smooth transition between today's user-centric Web and a new business-enabled one. The starting point is thus the current HTML oriented Web, while the (moving) target is the future services directory infrastructure. The connection between the two is achieved using XML descriptions of service interfaces.

The proposal contained in this document describes a simple mechanism by which a Web server can become a read-only directory advertising its own services, thus enabling discovery of services by B2B search engines. The proposed mechanism uses existing HTML pages to aid in the location of XML descriptions of business services. It introduces a basic HTTP discovery protocol, but architects an extension mechanism that uses WSDL documents to allow flexible integration with upcoming directory standards.

2 Describing Services

This proposal supports the following three classes of services descriptions:

1. The description of a single service.
2. The description of a collection of services.
3. The description of a repository of services information.

2.1 Single Service

For a single service, the description proposed here is composed of two parts: a non-functional characteristics description and an access mechanism specification.

The non-functional description is described by a WDS document, see [1]. It includes non-operational information directed mainly to human users or search engines. This information is variable by nature, but must at least include provider name and contact information, textual and keyword based descriptions of the service, and service categorization. It must also provide a reference to a WSDL document including access information.

The access mechanism to a service is described by a WSDL document. It provides all the information required to connect to the service, and includes abstract interface definition, access protocol information and service endpoints. WSDL is described in detail in [2].

2.2 Multiple Services

For a collection of services, the provided description is contained within a new type of document, a service description list. The service description list document contains both links to a number of WDS documents, as well as pointers to other service description list documents, enabling the creation of a hierarchy of descriptions. This aggregation document may be identified by its root element `<serviceListDef>`, qualified by the corresponding ADS namespace `http://www.ibm.com/namespaces/ADS`. Its children include a single `<description>` element which provides information about the set of services listed, and a variable number of `<serviceDef>` and `<serviceListDef>` elements. `<serviceDef>` elements are used to reference a WDS document and have a required `serviceRef` attribute containing the URL at which the service description can be located. They may also have `targetNamespace` and `name` attributes which provide the unique qualified name of the service. The `serviceListDef` element includes a `serviceListRef` attribute containing the URL where another service description list document can be found. A simple schema for service description list documents is included in the Appendix.

2.3 Service Repository

The third type of service description, the description of a repository, is based upon the use of a WSDL document to describe the access mechanism which should be used to retrieve the contents of a services directory. The WSDL document represents what is hopefully an agreed upon interface for communication with the repository. For example, the WSDL document could describe how to access an LDAP directory which contains information about a set of services. Because the WSDL document is simply a description of an access mechanism, it provides a point of continuing extensibility for this architecture that enables the use of richer access interfaces as they become available.

3 Advertising Services

This section describes the method by which a provider can advertise the existence and location of service descriptions. Other discovery mechanisms have been proposed in [3, 4]. The ADS method utilizes the well-defined service description documents described in the previous section. In addition, the proposal supports two mechanisms by which service providers can announce the availability of services descriptions. This can be done

1. Through the creation of a `svcsadvt.xml` file, which is placed in the root of the server.
2. Through the use of a `<meta>` tag carrying the value

```
serviceDescriptionLocation
```

in its `name` attribute, which can be included on any HTML page of a given host.

Service providers may create a file named `svcsadvt.xml` in the root directory of their server in order to advertise their offerings. The contents of this file may be a WDS document describing a single service, see Section 2.1, a service description list document describing a set of services, see Section 2.2, or a WSDL document describing the information necessary to contact a services repository. The use of a `svcsadvt.xml` file provides a consistent location for discovery tools to examine when searching for services.

The second method by which services may be advertised is through the use of a `<meta>` tag carrying the value

`serviceDescriptionLocation`

in its `name` attribute on an HTML page. Typically, one would include this tag on highly visible pages, like the root document of the server. The corresponding `content` attribute must be a valid URL indicating the location of the document listing the service descriptions for that business. Like the contents of the `svcsadvt.xml` file, the referenced document may either be a single WDS document (Section 2.1), a service description list document (Section 2.2), or a WSDL document containing information about the access mechanism for a services repository. In each of these cases, an HTTP GET request against this URL should return the desired information.

Note that it is possible to encode two (or more) `<meta>` tags within a single document, as in the following example.

```
<meta name="serviceDescriptionLocation"
      content="http://www.getquote.com/quote.wds"/>
<meta name="serviceDescriptionLocation"
      content="http://www.getquote.com/services-wsdl"/>
```

This way a search engine can choose the most appropriate access interface according to its capabilities.

The following section provides scenarios to illustrate the usefulness of the different advertisement mechanisms.

4 Usage Scenarios

A consumer of services information, such as a portal, may make use of both advertisement mechanisms. The `svcsadvt.xml` file provides a consistent starting point for performing a search for services provided by a site, while the second mechanism may be used during the standard indexing of web pages, and allows for the categorization of services based upon contextual information.

An offerer of services, such as a small business, may also utilize both mechanisms. The `svcsadvt.xml` provides a means by which a business can provide access to an aggregation of the information about the services which it provides. The second method, the reference of a services description from within an HTML document, can be used to advertise available services when a customer comes to visit the providers site. The visiting party does not need to examine any other locations to learn about services which may be available, but instead can extract that information directly from the given page.

5 Discovery Services

Assuming that services have been advertised using the methods described above, applications can then be developed to utilize the new information. Two main discovery patterns are likely to

arise from the availability of services advertisements: discovery through crawling and discovery through context.

Discovery through crawling follows the same approach that many search indices use today to gather information. Seeded with one or more starting points, the algorithm follows links present within the information until it reaches some predefined ending points, with interesting information being recorded along the traversal. An algorithm for a simple crawling engine for locating advertisements may be as follows:

```
For each starting URL:
  Store the URL in a queue if it isn't already there
  Store the URL corresponding to the svcsadvt.xml file for the
    location that is referenced by the starting point URL

Repeat until the queue is empty:
  Pop a URL off of the queue
  Load the data specified by the URL
  If the data is an XML file, see if we can recognize it as a WSDL or
    WDS file and do the appropriate thing
  If the data is an HTML file, process the HTML file as follows:
    If we find meta tags with the desired name attribute, process the
      referenced document
    If we find links, add them to the queue if they are not already
      there.  Then, take the host portion of the link and create a URL
      which looks for a svcsadvt.xml file in the root of the server
      referenced in the link.  If that new URL has not previously been
      placed onto the queue, do so now.
```

Discovery through context occurs when an HTML page containing an advertisement, via the meta tag, is visited. The context in which the link appears can provide information about the referenced service in addition to what may be contained within its description. This mechanism allows content providers to directly couple available services with the provided content. For example, a Web page selling flowers may offer a service to allow for a purchase order to be submitted through some type of electronic interchange. The service may be advertised directly from this web page, in which case not only can contextual information about the service be gathered, such as the fact that it has something to do with a business that sells flowers, but in addition, a smart browser may be able to modify its interface to allow for the user to interact with the offered service.

6 UDDI and Future Directions

The mechanisms in this proposal are complementary to what can be found in the Universal Description Discovery and Integration (UDDI) specification [5]. While UDDI provides for a fixed entrypoint for the discovery/location of a wide range of services, its responses are constrained by the information which has been registered in the directory. For those services whose presence has not been announced to the directories, for any number of reasons, the ADS mechanism provides a means by which their existence can be discovered and their functionality utilized. It is easy to see how one possible use of ADS may be to populate a UDDI directory, although further work needs to be done before that task is made easy. In addition, ADS provides for a means by which services and content can be tightly coupled by the content provider, whereas UDDI requires that a consumer actively search for a service.

One possible migration path to more tightly couple the ADS mechanisms and the UDDI directories is for them both to utilize a common description format. A single document, constrained by the UDDI schema, could perhaps be used to replace the current WDS and service description list documents which are referenced by ADS, without requiring a change of the discovery mechanism.

Appendix A: Schema for Service Description Lists

```
<xsd:schema targetNamespace="http://www.ibm.com/namespaces/ADS"
            xmlns:xsd="http://www.w3.org/1999/XMLSchema"
            xmlns:wsdl="http://www.ibm.com/namespaces/WSDL"
            xmlns:wds="http://www.ibm.com/namespaces/WDS"
            xmlns:ads="http://www.ibm.com/namespaces/ADS">

 <xsd:element name="serviceListDef" type="serviceListDefType"/>

 <xsd:complexType name="serviceListDefType">
   <xsd:element name="description" type="xsd:string"/>
   <xsd:choice maxOccurs="unbounded">
     <xsd:element name="ads:serviceDef" type="serviceRefType"/>
     <xsd:element name="ads:serviceListDef"
                  type="serviceListRefType"/>
   </xsd:choice>
 </xsd:complexType>

 <xsd:complexType name="serviceRefType">
   <xsd:attribute name="targetNamespace"
                  type="xsd:uriReference" minOccurs="0"/>
   <xsd:attribute name="name" type="xsd:string" minOccurs="0"/>
   <xsd:attribute name="serviceRef" type="xsd:uriReference"/>
 </xsd:complexType>

 <xsd:complexType name="serviceListRefType">
   <xsd:attribute name="serviceListRef"
                  type="xsd:"uriReference"/>
 </xsd:complexType>

</xsd:schema>
```

References

[1] N. Uramoto and K. Kosaka. *The Well Defined Services (WDS) Specification*. Downloadable as part of the Web Services Toolkit, available at `http://www.alphaworks.ibm.com/tech/WSTK`.

[2] E. Christensen, F. Curbera, G. Meredith, S. Weerawarana. *The Web Services Description Language (WSDL) Specification*. Download from `http://www.uddi.org`.

[3] Microsoft. *Discovery of Web Services (DISCO)*. Downloadable from
`http://msdn.microsoft.com/xml/general/disco.asp`.

[4] CommerceNet. *The eCo Framework*. Download from
`http://eco.commerce.net`.

[5] International Business Machines, Microsoft, Ariba. *Universal Description Discovery and Integration (UDDI)* Download from
`http://www.uddi.org`