B2C Web Service Discovery

David Aiken

Digital Enterprise Research Institute (DERI),
National University Ireland, Galway, Ireland

david.aiken@deri.org

Abstract

Research into Semantic Web Services is producing new technologies all the time. It is important that these technologies, such as Web service discovery, are made accessible to end-users as technologies are adopted based on the ease with which they can be applied.
Keeping this in mind, this paper discusses one possible implementation for end-users to adopt, to dynamically discover Web Services.

Introduction

Today, if an end-user wants to discover a Web Service they must use a search engine, based on a keyword matching, to locate html pages that in turn must be parsed by the user to determine if the specific Web Service satisfies their requirements. While Web Services technologies have positively influenced the potential of the Web infrastructure by providing programmatic access to information and services, they are hindered by the lack of rich and machine-processable abstractions to describe service properties, capabilities, and behaviour. This lack of rich semantics places a heavy burden on the user to search, parse and compare Web Services that best satisfy their requirements. So, how does one alleviate this burden?

An initial step would be to describe both the Web Service and the end-users goal, based on the most current promising technologies such as the following; for describing interfaces (WSDL [1]), describing semantic web services (WSML [2]), providing basic functionality for discovering services (UDDI [3], ebXML [4]). Once the web services are described and located in registries that enable efficient discovery, one can now concentrate on how to involve the end-user with as little burden as possible to discover their specific Web Service.

Related Work

This work is being applied to WSMX [5], a reference implementation for WSMO [6]. Currently WSMX provides a logic-based approach to discovery to Web Services that have been registered to WSMX. Concurrently, to the mentioned approach, support is also being implemented to discover Web Services that are not registered to WSMX. This is achieved by the means of a Focused Crawler [7], which is set to dynamically locate WSDL interfaces and query UDDI and ebXML registries. The initial step has begun to discover Web Services based on keyword matching and then increase the complexity to simple semantic discovery, and then rich semantic discovery of Web Services.

Implementation and Architecture

The end-user opens a browser and logs onto a specific site (all running locally at present). They are presented with the Form html page consisting of a text area to enter in some abstract keywords which best describe the Web Service they are looking for. When they click ‘Find my Web Service’, the Focused Crawler is invoked, which is sitting on a server as an executable Web Service. The Focused Crawler uses the Google API [8] to locate the WSDL interfaces based on the end-users entered keywords. The WSDLs are then returned and stored in YARS [9] as triples and ready for querying. YARS is RDF store and is also running as a web service. At this stage the user refines their query using the Query html page that sends N3 format [10] queries to YARS and returns results to the Results html page.

At present the N3 query format is still raw and needs to be presented to the user in an easy to read format. The next stage of development would be to present the user with the keyword textfield and behind the html page a Javascipt would put the keywords into N3 format and query YARS. This also is necessary for the returned triples. Currently, the end-user views the entire triple without any intermediate step to present the triples in an easy to read format. The intention is to parse the triple and post to the Results page the URI of the web service. The user would also be given the option to view

The implementation is at the very early stages of development but the fundamental architecture is in place (see Figure 1).

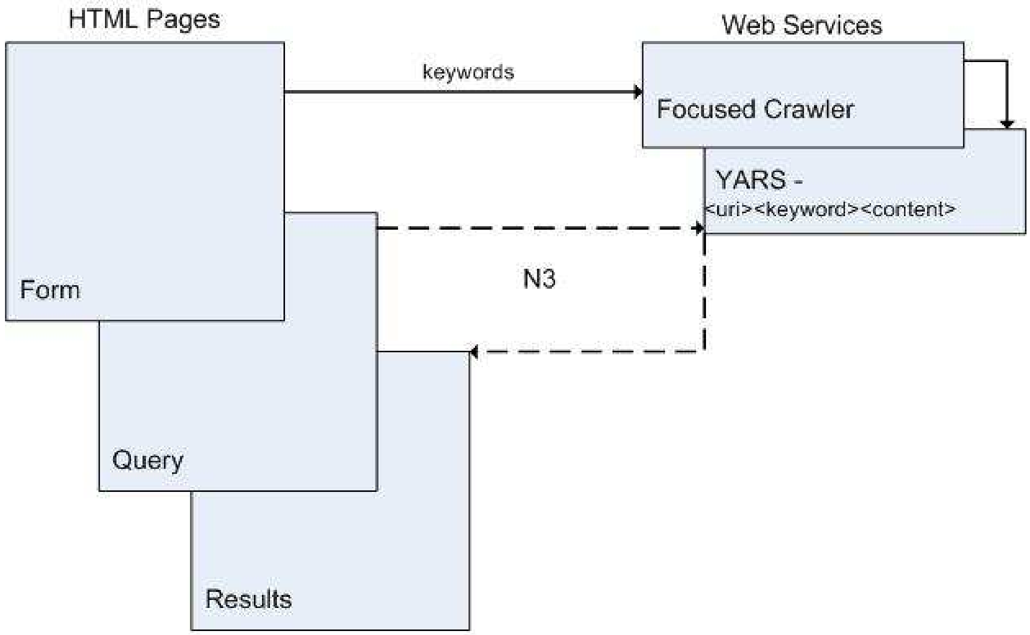

Figure 1: Architecture for B2C Web Service Discovery

Conclusion

The next stages of development would be to begin querying the registries with keywords initially. Working with WSML to enrich the semantic descriptions of web services, the registries could then be queried using these rich descriptions.

References

[1] WSDL-http://www.w3.org/TR/wsdl
[2] WSML-Web Service Modelling Language, D. Fensel, D. Roman
http://www.wsmo.org/2004/d7/d7.2/v1/
[3] UDDI-http://www.uddi.org/
[4] ebXML-http://www.ebxml.org/
[5] WSMX-Web Service Modelling Execution M. Zaremba environment
http://www.wsmo.org/2005/d7/d7.3/v1.0/20050109/
[6] WSMO-Web Service Modelling Ontology D. Roman, S. Arroyo
http://www.wsmo.org/2004/d7/v1/
[7] Focused Crawler-D10.1 Focused Crawler for Web Service Discovery, D. Aiken
http://www.wsmo.org/TR/d10/d10.1/v0.1/
[8] Google API-http://www.google.com/apis/
[9] YARS-Yet Another RDF Store
http://sw.deri.org/2004/06/yars/yars.html
[10] N3-http://www.w3.org/DesignIssues/Notation3.html