Investigating SOA Based Sharing of Peripheral Network Resources

Wandile Sibewu, Mamello Thinyane and Paul Kogeda
Department of Computer Science
University of Fort Hare, Private Bag x1314, Alice 5700
Tel: +27 40 6022464, Fax: +27 40 6022464
email: {wsibewu, mthinyane, pkogeda}@ufh.ac.za

Abstract-The growth of the internet sees it being available to people in marginalized rural areas. But the cost of network peripheral resources such as scanners, printers, and fax machines that are suppose to give a complete user experience to the people of these disadvantaged communities are still a bit expensive even though prices are decreasing. For this reason there are normally a limited number of such resources available for use by a large community. This paper introduces work that is being undertaken in the Dwesa community to make network peripheral resources available as Web Services. This paper will introduce the Service Oriented Architecture (SOA) paradigm as it relates to the work that we are doing, it will also introduce the proposed solution to the problem that the community currently face when it comes to sharing resources. It will also give an overview of the work that will be done to achieve the goal of the project.

Index Terms—SOA, web services, peripheral resources/devices, access node

I. INTRODUCTION

In trying to provide a ubiquitous computing environment for sharing network resources, we looked into Services Oriented Architecture (SOA). According to Sun Microsystems SOA is an IT approach in which application or other services make use of services that are available on the internet to complete a task [1]. SOA is ideal for this project because it provides an environment that is loosely coupled, that has interoperable components and message-based, asynchronous communication [2].

The most commonly used implementation of SOA is Web Services which provides a distributed computing approach for integrating heterogeneous applications over the internet, Web Services are based on open standards which makes them platform independent, programming language independent and even hardware independent [3]. There are four main technologies that makeup the Web Services stack, these are eXtensible Markup Language (XML), Simple Object Access Protocol (SOAP), Universal Description, discovery and Integration (UDDI) and Web Services Description Language (WSDL) [4].

The project is undertaken within the Siyakhula Living Lab (SSL) and is based in a region in the Eastern Cape called Dwesa which is a marginalized rural area (MRA). In Dwesa the internet availability is limited and is still only accessible at the local schools (network access nodes). Provisions are being made to extend the network and create more network access nodes by making Wi-Fi hotspots around the areas that surround the current network access nodes, and by investigating the use of mobile WiMAX[5].

The aim of the project is to develop Web Services wrappers for standard computing peripheral devices making them accessible as Web Services. Web Service wrapper is a development approach that makes it easy to integrate existing code into a services-based framework [6].

II. RELATED WORK

A considerable amount of work has been done by private companies for making resources available as Web Services, one example of this is the joint venture by HP and PUCRS a university in Brazil, their goal was to create a Web Service application for HPs LaserJet 4600 series printers [7]. The main focus of this application was to develop two ChaiServices, one is used to provide the printing URL and the other is to announce the device hosting the web service as a Universal Plug and Play (UPnP) device in the network. ChaiServer is HP's web services provider and it is implemented using Java, and so runs on a Chai VM [7]. UPnP was used to dynamically discover ChaiServices on the network.

A number of papers have been published that suggest that the automation industry is moving to a more SOA based infrastructure by adapting Web Services. An example of this is research done in the University of Loughborough to provide a SOA capable of supporting reconfiguration of production lines and Enterprise level devices management for the Ford powertrain assembly machine [2, 8]. This project was implemented using Device Profile for Web Services (DPWS), a specification that is promoted by Microsoft with a number of other companies [4]. It is a profile of web service protocols of a minimal set of implementation constraints to enable secure web service messaging, discovery, description, and eventing on resource-constraint endpoints [9]. The main drawbacks of this profile are that, it is relatively new; devices must support meaning that there should be a way to load the application into the devices [4].

We believe that by using the open standard that are encouraged by Web Services and wrapping existing code for controlling these devices, we can achieve the required out come. The existing code for controlling the device could be code written in Java, C++ or the actual device drivers.

III. SYSTEM DESIGN

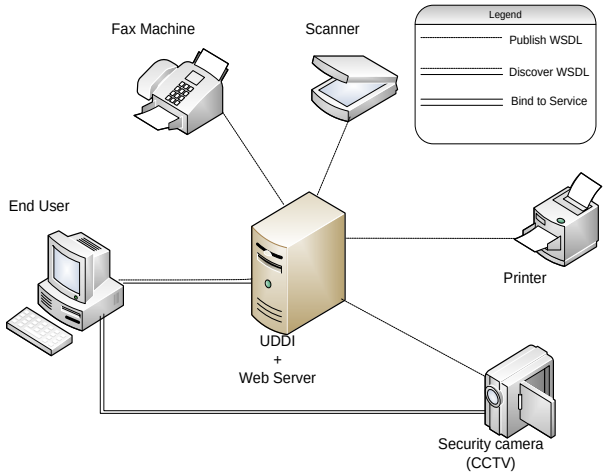

Figure 1: Proposed System Design

Each resource will be published to the local affiliated UDDI registry as a service. An affiliated UDDI registry is a registry that is deployed in a controlled environment but with limited access by authorized clients meaning that data may be shared with other registries in a controlled manner [10]. The service description which will be written in WSDL will be accessible by client devices who query the registry. After discovering the resource, a user or another service will communicate with the discovered service using XML-based messaging.

The Web Services stack is not limited to XML, WSDL, UDDI and SOAP, there are other specifications that need to been in cooperated when developing Web Services. Quality of Service (QoS), Security and management have to be addressed at all layers of the Web Services stack from the network which will use HTTP, XML-Based Messaging which will use SOAP, Service Description which will use WSDL to Service publication and Discovery which will use UDDI [3].

Currently in Dwesa the network access nodes are the schools in the area. The proposed application will be deployed in each network access node to manage interaction with local network resources. The affiliated UDDI registries that will be available in each access node will share data (services description) with each other to promote sharing of network peripheral devices in Dwesa. This will make it easy for the community to share these crucial but limited resources

IV. CONCLUSION AND FUTURE WORK

Open standard are the foundation onto which interoperability is achieved and Web Services a perfect example. Developing this application using Web Services technologies will save the community of Dwesa and other MRAs the much needed money by improving the sharing of resources amongst the community.

DPWS can be implemented using Java or C++ [11], the Java implementation supports both Java SE and Java ME meaning that it can be used to develop a client application that can consume Web Services from cellphones that support Java – this is a possible future extension of the project.

V. REFERENCES

[1] Ed Ort, *Services-Oriented Architecture and Web Services: Concepts, Technologies, and Tools*, April 2005 [online] http://java.sun.com/developer/technicalArticles/WebServices/soa2/soa2.pdf [Accessed 21/03/2010]

[2] F. Jammes *et al*, *Use of Web Services for next-generation automation system*, Loughborough University, January 2009

[3] M Endrei *et al*, *Patterns: Service-Oriented Architecture and Web Services*, pp 31, IBM corp., International Technical Support Organization, 2004

[4] Gunzer, H Engineer, S, *Introduction to Web Service*, Borland Developer network 2002, [online] http://bdn.borland.com/article/images/28818/webservices.pdf [Accessed 30/05/2010]

[5] Thinyane, M. Slay, H. Terzoli, A. Clayton, P., *Investigation into the Implementation of ICTs in Marginalized Communities*, 2006, SATNAC conference proceedings paper No: 213

[6] Glatard, T. Emsellem, D. Montagnat, J., *Generic Web Service wrapper for efficient embedding of legacy cides in services-based workflows*, Grid-Enabling Legacy Applications and Supporting End Users Workshop (GELA'06), Paris, France, 2006

[7] da Silva, OA. *et al*, *Providing printing Web services*, 2002, IEEE 5th International Workshop on Network Appliances. Liverpool Proceegings

[8] Phaithoonbuathong, P. *et al.*, *Web services-based automation for the control and monitoring of production systems*, 2010, International Journal of Computer Integrated Manufacturing, Vol. 23, No. 2, pp 126-145

[9] Jeyaraman, R. *et al.*, *Understanding Devices Profile for Web Services, Web Services Discovery, and SOAP-over UDP* [online] http://xml.coverpage.org/WSDD-UnderstandingDevicesProfile-20080910.pdf [Accessed 04/05/2010]

[10] OASIS, *Introduction to UDDI: Important Features and Functional Concepts*, 2004, [online] http://uddi.org/pubs/uddi-tech-wp.pdf [Accessed 9/04/2010]

[11] Schall, D. Aiello, M. Dustdar, S., *Web Services on Embedded Devices*, 2006 International Journal of Web Information Systems, Vol. 2, No. 1, p45

Wandile Sibewu received his BSc and BSc honours (Computer Science) degrees at the University of University of Fort Hare and is presently studying towards his Master of Science (Computer Science) degree at the same institution.