A Novel Architecture for Dynamic Invocation of Web services from Mobile

Srinivasu Yelsuri [*], G.B.V Padmanadh [#]
[*]Final M.Tech Student, [#]Assoc.professor
[*#]Department of CSE, Kakinada Institute Of Engg & Tech, Kakinada .

Abstract: We are proposing an efficient proxy based approach during the access of service oriented application information through mobile devices. Researchers are interested in invocation of functionalinaties to the user with minimal over head by using dynamic invocation and with language interoperability approach. Even though various traditional approaches introduced in the traditional mechanisms those are not optimum and used for only few simple query or minimal parameters. For dynamic invocation we introduced a novel Proxy based approach and for the language interoperability, we introduced Web services with proxy implementation in android

I.INTRODUCTION

Mobile devices have become highly popular in recent years. A computing device is usually considered a mobile device when it is small enough so that carrying it around is effortless. Traditionally, this could be considered the only constraint, but with the considerable rise in in the use of radio communications, particularly cellular phone networks and Wireless Local Area Networks (WLANs), mobile devices are typically expected to have some sort of wireless networking technology. SOAs deal with networked communication and require some sort of network access from the participants. Furthermore, mobile devices are meant to be mobile, not wired, so the focus here is specifically on mobile devices with wireless network access. For these reasons, the mobile devices discussed here will be assumed to have some sort of means for wireless networking and being able to access the Internet regardless of the underlying wireless networking technology.[1]

The question of at which point a device is small and light enough to be a mobile device clearly demonstrates the problems with the mobility criteria presented. To demonstrate the problem, modern laptop computers typically have wireless network connections and some laptops are quite small, weighing in at about 1 kg. In contrast, mobile phones and Portable Digital Assistants (PDAs) tend to weigh less than 300 grams and have wireless networking which work at higher ranges. Small laptops are the size of large books, while mobile phones are usually smaller than pocket-sized books. Should laptops, phones and PDAs all be considered mobile devices?[2][3][4] However, one must consider that laptops, PDAs and mobile phones vary considerably in size, especially when older models are compared to current ones, and phones and PDAs offer much better standby times than laptop computers. Due to these factors, it becomes increasingly difficult to determine what are the requirements for a mobile device, particularly since the sizes and wireless capabilities of laptops have been improving, while the processing capabilities and software sophistication of mobile phones have improved significantly in recent years.[5][6]

From the viewpoint of this thesis, devices with Internet connectivity are of particular interest, as well as the low power requirements of battery operation and the limited processing power of smaller devices. Considering this viewpoint, the actual nature of the device is not so important, particularly since implementing SOAs demands sophisticated software capabilities. However, long standby time and good portability are key concerns to users of mobile devices and by sheer market share, mobile phones are the most popular devices in use today and will continue to be so for the foreseeable future. Given these considerations, this thesis focuses primarily on mobile phones, but the results and conclusions should be easily applicable to PDAs and mobile phones.[7]-[10].

II. RELATED WORK

In the mobile web services domain, the resource constrained mobile devices are used as both web service clients and providers, still preserving the basic web services architecture in the wireless environments. While mobile web service clients are quite common these days, the research with providing web services from smart phones is still sparse. In our mobile web service provisioning project one such Mobile Host was developed proving the feasibility of concept[11]-[14]

In this chapter, the currently used implementation techniques for SOAs are discussed. The focus is on the traditional model of Web services, using SOAP for transferring data, WSDL for describing the services and Universal Description, Discovery, and Integration (UDDI) for service discovery. In order to evaluate service description, the current status of XML schema description languages is described. The description of the traditional Web service model is augmented with a

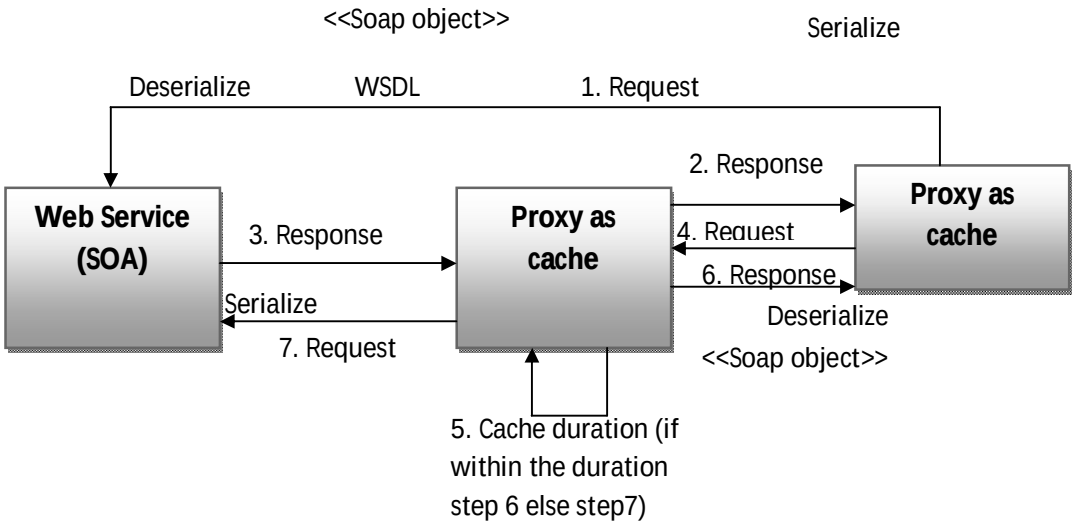
description of the REST architecture and the use of the REST model in building SOAs. The issue of underlying protocol stack is taken up with considering how Web services implementations need to interact with lower level protocols, in particular HyperText Transfer Protocol (HTTP).

Generally, the WSDL document, that defines and describes a web service, consists of information specific to the location of the service (binding information) and the operations (methods) the service exposes. The web services are published by advertising these WSDL documents in a UDDI registry. The registry maintains a reference of the WSDL documents. Any potential web service client searches for the service in the public registry, gets the description of the service and tries to access the service using the information specified by the WSDL. Since the Mobile Host is implemented on the smart phone, mostly by using the basic web services architecture, the standard WSDL and UDDI registry can theoretically be used to describe and publish the services. Obtaining the binding information of the mobile web services can be tricky as it needs the IP address of the Mobile Host, where the services are deployed. Different means of accessing the services deployed on the Mobile Host are observed in [16].

Mobile devices are getting more pervasive, and it is becoming increasingly necessary to integrate web services into applications that run on these devices. We introduce a novel approach for dynamically invoking web service methods from mobile devices with minimal user intervention that only involves entering a search phrase and values for the method parameters. The architecture overcomes technical challenges that involve consuming discovered services dynamically by introducing a man-in-the middle (MIM) server that provides a web service whose responsibility is to discover needed services and build the client-side proxies at runtime. The architecture moves to the MIM server energy-consuming tasks that would otherwise run on the mobile device. Such tasks involve communication with servers over the Internet, XML-parsing of files, and on-the-fly compilation of source code.

But in a commercial environment with Mobile Hosts, and with each Mobile Host providing some services in the wireless network, the number of services expected to be published could be quite high. In such a situation, a centralized solution is not a best idea, as they can have bottlenecks and can make single points of failure. Besides, mobile networks are quite dynamic due to the node movement. Nodes can join or leave network at any time and can switch from one operator to another operator. This makes the binding information in the WSDL documents, inappropriate. Hence the services are to be republished every time the Mobile Host changes the network. This process leaves many stale advertisements in the registry. Keeping up to date information of the published mobile web services in centralized registries is really difficult.

III. PROPOSED SYSTEM



web services are the service oriented applications, where we maintain business logic which is accessed by the clients or users remotely. Service oriented applications provides language interoperability because it converts the native language to web service description language, here everything transmits in terms of soap object. At sender side it serializes the native language can be converted to web service description language and at the receiver end it

Deserialize in to receiver native language. Usually web services registered at UDDI(Universal Description, Discovery and Invocation) .This architecture involves service requestor, who is accessing the service. Service provider, who is providing service to the consumers. Service broker who is mediator between Provider and consumer. Here our Service consumer is a mobile client, Mobiles are not like desktops are laptops while we are developing mobiles as client ,we need the consider the resource of the device, Web service provides language interoperability, which has the capability of converting the language to native language. For implementation purpose we design the mobile client in android. For accessing the service, Initially it needs to register the service, Mobile client should know ,where the service is running and it requires to construct the soap object, which contains parameters ,Envelop and other. For continuous discovery and invocation of service ,power consumed by the mobile more ,so that is the major issue during the discovery and consuming, so that is the reason we are implementing proxy between the service and mobile client.

Proxy can be a computer or an application. In our paper it works as cache for dynamic invocation of the service. It helps the consumer to consuming the services in an efficient way. If consumer (mobile client) try to invoke the previously invoked content, client need not to process the same request again and again until a specific time. After the specified time mobile client request directly sends to the service and service update the cache and sends the response to the mobile client.

A)Business Logic

Service contains the business logic of the operations, which can be accessed by the mobile users dynamically and there are registered in the server and these business language uses the web service description language (wsdl) for the communication and uses the protocol soap(Simple access protocol),Every thing is transmitted in the form of soap objects.

B)Mobile Users

Mobile users requests the specific string or operation sends a request to the server, initially requset checks in cache if it not availble in cache,request forwards to the web server afetr processing the request user recieves the response.This mobile application is developed using the android for efficient data access from the service.

C)Proxy Generation and Dynamic Invocation

We are generating client side proxy, it caches the previously accessed information from the server, when ever user makes a request again for the same information, and user need not access the information of wsdl files from the server. Whenever data updated in the server.

D)Consuming services

User consumes the service by creating the soap object make a call to the service with the specified name space and operation ,service process the request sends the response in the format of web service description language, at receiver end it can be converted to native language.

SOAs are based on loose coupling and open, transparent communication interfaces. To achieve these goals, the dominant choice for building SOAs has been the use of XML technologies for data transfer. Consequently, tool support for building SOAs must be based on XML handling capabilities , full use of XML requires the knowledge of several different data formats and specifications. However, the typical uses in data processing are most dependent on having basic XML parsing capability in order to be able to retrieve the relevant data from an XML document

IV.IMPLEMENTATION DETAILS

Mobile Host is a light weight web service provider built for resource constrained devices like cellular phones. It has been developed as a web service handler built on top of a normal Web server. The SOAP based web service requests sent by HTTP tunneling are diverted and handled by the web service handler component. Android mobile version 2.3.3 is used as mobile client ,Web services are implemented in C#.net,It efficiently shows the language interoperability .

Business logic runs at specific location that can be identified by the url and which is used by the mobile client to access the business logic by forwarding the input parameters to respective url in the form of soap object, implicitly it serializes the object and converts into the web service description language again at the receiver end it converts the wsdl into receiver native language by using the deserializer, that shows the language interoperability due to wsdl.

For implementation purpose we designed an application which uses the business logic as bank common manipulations if any recent accessed information is there, it is available in cache otherwise forward the request to the server, during the reply before displaying to the user place the accessed content in cache for next time access.

V. CONCLUSION

Finally our approach generates an efficient results than the traditional approaches of static and continuous discovery and invocation of web services from the mobile client. If data available in the proxy cache Mobile client need not access the service again, it reduces the consumption of the battery power of mobile phones.

REFERENCES

[1] C. Aggarwal, J. Wolf, and P. Yu, "Caching on the World Wide Web," IEEE Trans. Knowledge and Data Eng., vol. 11, no. 1, pp. 94- 107, Jan./Feb. 1999.
[2] R. Boyer and J. Moore, "A Fast String Searching Algorithm," Comm. ACM, vol. 20, pp. 762-772, 1977.
[3] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web Caching and Zipf-Like Distributions: Evidence and Implications," Proc. IEEE INFOCOM, pp. 126-134, 1999.
[4] J. Cao, M. Andersson, C. Nyberg, and M. Kihl, "Web Server Performance Modeling Using an M/G/1/K*PS Queue," Proc. 10th Int'l Conf. Telecomm., 2003.
[5] Celimaris Vega Citrix Consulting, MetaFrame XP Oracle 11i Application Scalability Analysis, http://support.citrix.com/ article/CTX101887, 2002.
[6] A. Chakravarti, G. Baumgartner, and M. Lauria, "The Organic Grid: Self-Organizing Computation on a Peer-to-Peer Network," IEEE Trans. Systems, Man, and Cybernetics, vol. 35, no. 3, pp. 373- 384, May 2005.
[7] M. Chatti, S. Srirama, D. Kensche, and Y. Cao, "Mobile Web Services for Collaborative Learning," Proc. IEEE Int'l Workshop Wireless Mobile and Ubiquitous Technology in Education, pp.129-133, Nov. 2006.
[8] CodePlex, ProxyFactory Home Page, www.codeplex.com/ ProxyFactory, 2011.
[9] R. Costello, "Building Web Services the REST Way," http:// www.xfront.com/REST-Web-Services.html, 2011.
[10] G. Dattatreya, Performance Analysis of Queuing and Computer Networks. CRC Press, 2008.
[11] I. Duda, M. Aleksy, and T. Butter, "Architectures for Mobile Device Integration into Service-Oriented Architectures," Proc. Int'l Conf. Mobile Business (ICMB '05), 2005.
[12] R. Fielding and R. Taylor, "Principled Design of the Modern Web Architecture," ACM Trans. Internet Technology, vol. 2, no. 2, pp. 115-150, 2002.
[13] J. Flinn and M. Satyanarayanan, "PowerScope: A Tool for Profiling the Energy Usage of Mobile Applications," Proc. IEEE Second Workshop Mobile Computer Systems and Applications, p. 2, 1999.
[14] G. Gehlen and L. Pham, "Mobile Web Services for Peer-to-Peer Applications," Proc. IEEE Conf. Consumer Comm. and Networking, pp. 427-433, Jan. 2005.
[15] GeoTrust Corp, http://www.geotrust.com/enterprise-slcertificates/ georoot, 2011.
[16] X. Gu and K. Nahrstedt, "On Composing Stream Applications in Peer-to-Peer Environments," IEEE Trans. Parallel and Distributed Systems, vol. 17, no. 8, pp. 824-837, Aug. 2006.
[17] V. Gupta, "Finding the Optimal Quantum Size: Sensitivity Analysis of the M/G/1 Round-Robin Queue," ACM SIGMETRICS Performance Evaluation Rev., vol. 36, pp. 104-106, 2008.
[18] A. Halteren and P. Pawar, "Mobile Service Platform: A Middleware for Nomadic Mobile Service Provisioning," Proc. IEEE Int'l Conf. Wireless and Mobile Computing, Networking and Comm. (WIMOB), 2006.
[19] R. Heffner, "SOAP versus REST: A Comparison," http://www.
forrester.com/research/document/excerpt/0,7211,35361,00.
html, 2011.

BIOGRAPHIES

Srinivasu Yelsuri pursuing Mtech on Kakinada Institute Of Engg & Tech Kakinada . He completed his graduation in Prasad Institute of Technology and Science.

G.B.V Padmanadh is working as Assoc. professor in dept of cse in kakinada institute of Engg and tech. He completed his post graduation in gitam university Visakhapatnam