Hybrid Framework for Effective Mobile Web Service Provisioning

Kishor S. Wagh¹ and Dr. R. C. Thool²

1 Marathwada Mitra Mandal I.O. Technology, Department of Computer Engineering, Pune, India
   Email: waghks@gmail.com
2 SGGS College of Engineering and Technology, Department of Information Technology, Nanded, India
   Email: rcthool@yahoo.com

Abstract— Due to the advancements in smart mobile phone’s capabilities, web technologies and wireless technologies, it is changing the role of mobile devices as a web service consumer to a web service provider. The future of mobile web service provisioning is more exciting because day by day Smartphone users are increasing. In future it is called as Mobile Ubiquitous computing because the smart phone provides services anywhere, anytime. But provisioning web services on mobile devices in a wireless environment are challenging due to resource restrictions of the device and unreliable mobile network. Providing continues web services from a mobile phone, it consumes more resources and battery power. Processing web service request and communicate response will drain the battery rapidly, hence processing and communication overhead should be kept at a minimum. The solution for this is to develop lightweight frameworks for provisioning web services on mobile phones. This paper describes the hybrid framework to overcome the above problems and support continuous provisioning of web services on android mobile host. The main aim is to deploy web services on Android mobile using REST architecture and access web services through SMS and HTTP. The paper presents the detailed study of Android based mobile web service provisioning using light weight web services on real time setting. Our experimental setup and results indicate that hybrid framework effectively reduces resource consumption, processing and communication overhead.

Index Terms— Mobile web service provisioning, Android Host, Mobile Web Services, Mobile Web Service provider, SOAP, REST, Hybrid framework

I. INTRODUCTION

Usually, handheld devices are resource constrained devices. These devices have many resource constraints or limitations like small display screen area, less battery capacity, less processing power, less memory addressing capacity. While designing web service provisioning should consider these resource constraints and investigate lightweight framework to reduce the processing and communication overhead of a mobile host to improve the performance of mobile hosts and reduce the battery power consumption. But amazing advancement in a Smartphone capability in terms of processor, memory, storage capacity, display screens, built-in cameras, Bluetooth technology, infrared port and large variety of sensors are embedded within the devices to expand their capabilities and functionalities. Remarkable improvement in wireless communication
is in the spectrum efficiency to accommodate more number of users and continuous improvement in data transmission in cellular networks. Due to all above parameters and reduced cost of devices and Internet 2G, 3G and 4G plans; there is an amazing growth in Smartphone users. The next generation 4G introduces flexible access of future services and applications from mobile devices. As a response to this growth, a huge amount of applications and services have emerged in the market in order to provide various types of services and information. However it is still in early stages to investigate effective lightweight framework for mobile hosted web service provisioning. There are lots of challenges to provisioning web services in the cellular domain on resource constrained mobile devices. These challenges are reduced resource consumption, mobility security, scalability, Dynamic operating environment and battery consumption.

There are several reasons for hosting a web services on mobile phone to:
- Smartphones always with user, he/she can maintain/update web service anytime anywhere
- If any updates/modification in web service, it is not affecting the client
- Client users can access the web service in an emergency situation if health related web services are available
- User get information on figure touch
- The developer can test locally client web applications without having to consume network access

Web services are self-describing modular business applications that expose the business logic as services over the Internet through programmable interface and where IP can be used finding ways to subscribe and invokes those services in the fixed wired network. Same way Mobile Web services are defined as self-contained modular applications that are defined, published and accessed across the Internet, in a wireless mobile communications environment using standard protocols. This technology has evolved from advances in the mobile device technology, rapid growth of Web services development and progression of wireless communication in parallel to widespread use of internet applications.

Web service defined by[1]: A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine understandable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards. The W3C [1] also states that two major classes of Web services: REST-compliant Web services, in which the primary purpose of the service is to manipulate XML representations of Web resources using a uniform set of “stateless” operations; and arbitrary Web services, in which the service may expose an arbitrary set of operations.

Web services are different from web servers. Web services are the internet based web application that can be deployed independently and invoked by a web client browser. Web Services used in mobile classified into three main categories:

- Network hosted Mobile Services: The mobile operator and third party mobile service provider providing such services. These services used mobile network infrastructure and the various service enablers that they could be offered as web service interfaces to the developer and internet service providers. These services are best provided as web service interfaces (WSI), considering the heterogeneous developer platform requirements available out there. The service enables such as Payment, SMS, MMS interfaces, Locations, Profile services is few of the services that could be provided as a network / operator hosted Mobile Web Services.
- Device hosted Web Services: The smart mobile devices are now computationally capable, such that they can host Web Services on the Mobile devices directly. At the same time also allowing great potential for applications and service innovations that can be provided by individual mobile device owners by utilizing Mobile device as a: Web Services Client, Web Services Provider, Web Services Broker.
- Device Centric Peer-to-Peer Web Services: It uses decentralized architecture, each Mobile device act as service consumer or service provider in this architecture. Web services hosted on a device can communicate with another mobile device vice versa. Mobile devices in P2P call peer, it allows other mobile device to access services. This can also enable mobile user to provide services that can locally interact with services running on other non-mobile devices as well.

The paper is organized as follows: section II describes the general and proposed mobile web service provisioning architecture for mobile host. Section III is a related work; section IV covers the details of hybrid framework for REST based mobile web service provisioning. Section V gives the information about results and last section VI concludes the paper and highlights the future research direction.
II. ARCHITECTURES FOR MOBILE WEB SERVICE PROVISIONING

Mobile web service provisioning allows deploying, publishing, discovering and executing of web services in wireless environment using standard protocols. Mobile web service provisioning can be classified based on architecture as SOA (Service Oriented Architecture or SOAP based Mobile web service provisioning) and ROA (Resource Oriented Architecture) or RESTful based Mobile web service provisioning.

A. SOAP Based Mobile Web Service Provisioning

Web services that use SOA architecture and the SOAP (Simple Object Access Protocol) protocol to communicate between the client interface and provider are called SOAP-based Web services. SOAP-based Web services are used RPC-like interactions while communicating with remote systems. SOAP based service provider and web service client need to follow service syntax and data types. It uses Web Services Description Language (WSDL) for describing functionality offered by deploying web service and Universal Description Discovery and Integration (UUDI) is used as a registry for deployed web services. The main functions of UDDI are to manage, maintain and provide reference to the WSDL file. For interaction between the service provider and service requester XML is used. This type of web service provisioning has only one end point communication address and which is defined by the single URI. This address is stored in UDDI registry and used by the web service to receive SOAP request. The Web service itself is responsible for handling all the communications directed to its internal operations.

SOAP based web service provisioning architecture has three components: Web Service Requestor (Client), Web Service Provider (Mobile Host) and Web Service Registry. The mobile web service provider publishes its Web Services with the service registry. The service requestor client (mobile, PDA, PC or Laptop) searches (“Find”) the UDDI registry for the services, and the UDDI compatible service registry refers the respective WSDL. The service requestor client accesses the described Web Service, using SOAP is shown in figure 1.

![Figure 1. SOAP based Mobile web service provisioning](image1)

Figure 1. SOAP based Mobile web service provisioning

![Figure 2. REST based Mobile web service provisioning](image2)

Figure 2. REST based Mobile web service provisioning

B. REST Based Mobile Web Service Provisioning

SOAP is designed for a fixed network environment where high end servers are used, but REST is alternative to SOAP for the fixed network environment as well as mobile network environment and resource constrained devices. REpresentational State Transfer (REST) is also called Resource Oriented Architecture (ROA), is a style of software architecture that relies on the fact that any resource (such as Web services) can be identified by their URLs. REST is an architectural style [2] defined in his dissertation and it is derived from the Web, and its purpose is to assemble the fundamental design principles that enable the greater scalability, growth and success of the Web services on the web. REST is initially used for publishing hypermedia documents but later it is becoming a common scheme for realizing Web services on fixed network as well as a mobile network. REST follows a different philosophy than SOAP by focusing on data instead of operations and also to provide resource friendly alternative to SOAP. RESTful Web services are tightly coupled with the HTTP protocol however and compromise their flexibility and portability. RESTful Web services gained much attention from the Web community due to their simplicity and scalability. Major Web service providers such as Google, Amazon, Yahoo, and eBay adopted the RESTful Web services approach in their offered
Web services. RESTful Web services use unique URIs for identification of resources. These resources are accessed and manipulated using a set of uniform methods GET, POST, PUT and DELETE.

REST based web service provisioning architecture has two components: Web Service Requestor (Client), Web Service Provider (Mobile Host). In this web client directly send HTTP request to host and host parse the request and web servlet generate response, and then response sends back to the client using the same protocol. REST based web service architecture shown in above figure 2.

As per our previous preliminary comparative study in [3] and Android Mobile Host performance evaluation of SOAP and REST in [4], it showed that REST is a more suitable architecture for provisioning web services on mobile smart phones. The RESTful approach enhances the performance of mobile Web services also described by many researchers. Smartphone prices are reduced and easily available the internet connection in very cheap rate, usability of Smartphone increased. While considering todays scenario of Smartphones is becoming more and more-smart in terms of processing power or more capable processor, memory addressing capability and advancement in mobile wireless technologies and web technologies. Due to this, mobile computing is becoming more and more popular for pervasive computing. Deploying web services on smartphone it is today need, because anyone can access mobile hosted web services any time anywhere and managing mobile hosted web services is also very easy.

C. Proposed System Architecture for Mobile Web Service Provisioning

Web services technology is about providing a standard mechanism for communication between service consumer and service provider that is XML or HTTP. This standardization is independent of the implementation of applications, platforms and operating systems under which they operate. It also does not depend on a particular communication protocol that may be used by communication. The solution for this is to use hybrid development environment. We propose an innovative novel idea of deploying web service on Android Smartphone and it use the short messaging service for an alternative response, i.e. unavailability of GPRS or any non-Java or multimedia mobile won’t be an issue the HTTP request for client side. Although there have been many traditional solution deployment approaches, each of them mainly focuses on a particular product. In this paper, we propose an alternative solution deployment paradigm as hybrid solution deployment, which gives an alternative approach for solution deployment state-of-the-art provisioning tools.

The proposed system architecture provides mobile web services that integrate existing web service standard, wireless technology, mobile computing and mobile inbuilt SMS service. The major difference between previous web service provisioning architecture and the proposed architecture is that the previous architectures are based on XML and HTTP only. SOAP based architectures uses for communication XML or HTTP tunneling and REST based architecture uses for communication simple HTTP protocol, but proposed architecture is based on HTTP as well as SMS (Short Message Service) as an alternative to access web service from a Smartphone. The approach presented here follows a different from a technical and communication point of view, the mobile Web Service provider communicates as a Web Service client with a dynamically generated Web Service with the help of HTTP protocol or SMS mobile service. Proposed Web service provisioning system architecture for mobile host shown in figure 3.

![Figure 3. Proposed REST based Mobile web service provisioning architecture for Hybrid Framework](image)

The main components of proposed system architecture are: Mobile Host, Web service HTTP clients, SMS services client. The mobile host in proposed system provides the service to both clients (HTTP web service client or SMS request from any mobile device). HTTP and SMS based web service requester is normal we service client. The proposed architecture is quickly responding to client and reduces overhead of mobile hosts. It has also solved the problem single point of failure of mobile hosts.
For feasibility and testing purpose we deployed Hospital and Blood bank search web services on Android Smartphone and use the short messaging service for an alternative response, i.e. unavailability of GPRS or any multimedia mobile won’t be an issue web service request from the client side. This approach is very useful those are in the ruler area having connectivity problem due lack of proper infrastructure and this also reduces battery consumption.

III. RELATED WORK

Over the past couple of years, researchers have proposed different architectures and frameworks for providing Web services by mobile devices. These research studies showed the possibility of hosting Web services on resource-limited devices. Most of the research described mobile web service provisioning architectures and frameworks based on SOAP but very less research work for REST based mobile web service provisioning. Each one of these approaches addresses and deals with certain challenges facing mobile service provisioning such as reachability, reliability, and scalability. Most of the current mobile Web services provisioning architectures are immature and still in the early stages. Also not address most important issue such as battery power consumption.

The concept of Mobile Web services was first introduced as a computing paradigm in the early twenty-first century. Since then, many research efforts have focused on enabling reliable mobile service provisioning despite the constraints and peculiarities of mobile environments. However amazing advancement in mobile device manufacturing and wireless technologies triggered a great interest in the research community in implementing mobile services. Several research efforts have studied web service provisioning from resource-constrained mobile host, however, they address specific aspects in isolation which may yield inefficient architectures. Mobile services and used as a mobile phone as a web server started before one decade back in 2002 with the Personal Server [5], a concept that enables people to store and access data and applications hosted on their mobile devices through convenient interfaces found nearby such as public display monitors, information kiosks, and other computers. Personal server was an idea to unleash the potential of user’s mobility, while freeing users from the small and limited displays of their mobile devices. There were similar efforts taken by IBM in [6] mobile devices used as a web service provider presented. This work presents shopper-kiosk specific scenario, where the web service hosted on a mobile device and addressed a of key design issues of hosting web service on a mobile device. The author [7] presented architecture of a Web server running on a Symbian platform. This computing model was further improved by the Nokia Research Center, where they ported Web server to the Symbian/S60 platform [8]. This server permitted users to publish and share photos and other personal information on their terminals, or even web server offer an interface to remotely configure the device itself.

Feasibility of mobile web service provisioning elaborated by [9, 10] is along with performance analysis of the mobile hosts. In [11] author describes a mediation framework for mobile web service provisioning and author used the JXTA for deploying web service on mobile host based on SOAP standard. This approach is same as p2p service provisioning. “Reference” [12] author propose a lightweight Web service provider toolkit that can be implemented in any object oriented language to facilitate the hosting of Web services on a number of handheld devices. The toolkit is based on a stack of four layers. The “Transport Layer” in which HTTP is used as the main transport mechanism for the exchange of SOAP messages. The “Security Layer” uses SSL and TLS to secure the incoming and outgoing SOAP requests/response messages. A “SOAP Engine Layer” parses the SOAP messages and uses RPC as the binding technique. The “Target Web service Layer” maintains a list of deployed Web services in an XML file. The toolkit scalability was tested on a PDA running Windows Mobile 5.0 with an Intel processor at 624MHz and 64MB memory, but it was found to be limited to support only a dozen concurrent Web service client requests.

According to [13] author proposes a framework for hosting Web services on mobile devices. It consists of several components such as a “Request/response Handler”, a “SOAP Engine”, a “Publication/discovery Manager”, a “Migration Manager” and a “Context Manager”. The framework was tested under a real-world application using Windows mobile phones and was proven to support Web services in a mobile environment. However, the framework still lacks a fine-grained migration policy as well as it uses Bluetooth as a communication medium between the mobile clients and the mobile Web service host which makes the application of the framework very limited to short range connections. In [14] author proposed three different mobile web services provisioning approaches and classify them into three categories as proxy-based, Peer-to-Peer, and asymmetric infrastructure. In proxy based approach uses the high end computer system, which was attached to the fixed wired network. It used proxy based Jini technology for enabling different network
services. This approach fully depends on proxy node, this approach takes advantage of the fixed wired network but mobile web service provider connected to the proxy node using wireless technology, but intercommunication between proxy node and mobile web service provider limits the performance and increases the overhead on mobile phone. Peer-to-Peer mobile web services provisioning architecture solve the issue of the centralized repository and it is a flexible and cooperative resource sharing approach of network service. The asymmetric architecture of web service provisioning is same as the traditional Web services architecture except that the service provider is a mobile phone.

Reference [15, 21] author presented a distributed mobile service provisioning framework that partitions the execution of resource-intensive Web services between the mobile provider and a back-end server. The framework offers a distributed execution engine where tasks that require real time access to local resources are executed on the mobile devices, while the remaining processing is offloaded to a remote server. A partitioning strategy is presented to arbitrate the split of service tasks. Their partition technique relies solely on the available resources. If the available resources satisfy the Web service’s execution requirements, the execution is performed entirely on the mobile side. The framework does not offer options based on potential performance benefits and current context. In some cases, it might turn out that local execution yields better performance than remote execution, especially when a large amount of data transfer is required.

In [16] presented a concept of REST based Mobile Web Services (MobWS) provisioning and its comparison with a similar SOAP architecture in terms of HTTP payload. Author analyzed based on payload in a mobile device environment he found that direct implementations of SOAP-based Web services may introduce a significant and an unacceptable performance overhead on the resource-constrained mobile devices due to the increasingly thick SOAP messages and its expensive parsing requirements which demand for high resources (processor power, memory, battery and bandwidth). Conversely, REST architecture style supports direct interaction with Web services using the standard Uniform Resource Identifier (URI) and can greatly facilitate avoiding application performance degradation factors such as SOAP message length and parsing. The flexibility and loose coupling that RESTful Web services can afford are beneficial to both, mobile Web service hosts as well as mobile clients.

In [17] author investigated mobile RESTful Web Services and developed a prototype for providing the RESTful photo Web Service on standard devices and accessed it by different mobile clients. Author identified several issues such as latency and data format that need particular attention when applying REST concepts to the mobile environment. However they focused on consuming RESTful Web Services from mobile devices and did not address the provision of services from mobile host. “Reference” [18] tackles the adaptive and dynamic provisioning of RESTful-based Web Services and they have presented RESTful-based framework that provides Web Services to mobile clients but they did not allow providing these services from mobile devices, our solution builds beyond consuming mobile Web Services, it allows consuming and providing Web Services.

In [19] explored the way reduces the overhead of services provisioning in resource-constrained mobile device. Author presented a distributed mobile service provisioning framework that partitions the execution of resource-intensive Web services between the mobile provider and a backend server. The framework offers a distributed execution engine where tasks that require real time access to local resources are executed on the mobile devices, while the remaining processing is offloaded to a remote server. A partitioning strategy is presented to arbitrate the split of service tasks. Their partition technique relies solely on the available resources. If the available resources satisfy the Web service’s execution requirements, the execution is performed entirely on the mobile side. The framework does not offer options based on potential performance benefits and current context. In some cases, it might turn out that local execution yields better performance than remote execution, especially when a large amount of data transfer is required.

In [20, 22] author discussed offloading and migration mechanisms that facilitate provisioning of adaptive and distributed mobile Web services from mobile host. For performance evaluation uses fuzzy logic rule sets are to trigger and control offloading schemes of the web service provider. In [23] author developed Android Mobile host using RESTful philosophy and it uses open source OSGi framework for provisioning web services on android mobile host. For publishing, discovering and addressing the Mobile web services ZeroConf networks is used. HTTP/XMP protocols are used for accessing web services from mobile host. In [24] author proposed architecture for dynamic invocation and discovery of web services in mobile based on proxy mechanism and it uses a cache for dynamic invocation of the service. It helps the consumer to consuming the services in an efficient way.
IV. HYBRID FRAMEWORK

Web service provisioning is defined for the fixed wired network environment and it is not for the mobile wireless environment due to the heavy requirement of resources. Web service provisioning frameworks are developed for static high end servers not for resource constrained mobile phone. In addition these standard frameworks are too large to be deployed on resource constrained mobile devices and they require a running time environment that is not available on mobile devices. Also providing web services from mobile hosts consumes a large amount of resources and drains the battery within a short period of time. Our previously preliminary work [3, 4] has shown that RESTful based mobile web service provision frameworks outperform its corresponding SOAP-based mobile web service frameworks. Hence, RESTful-based Mobile Host is more suitable for deploying, providing and executing web services from resource constrained mobile devices. In this paper our previous RESTful-based Mobile Web Service Provisioning framework has been extended to allow the web client to access web services using HTTP and SMS. The hybrid framework has the following components as follows in figure 4.

![Diagram of proposed hybrid framework](image)

Figure 4. Proposed Hybrid Framework for REST Based Mobile Web Service Provisioning

Function and use of each block of hybrid framework is given below:

- **Request Listener**: The main function of request listener is to accept incoming HTTP and SMS request, if the request is from the HTTP client listening through server socket class and initiating a new thread for each request to support concurrency and creating input and output stream for communication. If request from SMS to mobile service provider then it creates the input and output stream for communication.
- **Request Handler**: The main function of this is to recognize the request from HTTP client or SMS client. If request from HTTP, forwarded to the HTTP manager and if the request from SMS client, forwarded to SMS manager.
- **HTTP Manager**: This block will extract the HTTP request and send it to the HTTP parser.
- **SMS Manager**: SMS manager block reads the content of SMS and send it to the SMS parser.
HTTP parser: The main function of this block is to parse the HTTP request and acquire the required information for invoking a Web Service such as the name of the service, service URL and some parameters. This required information extracted with the help of database mapping module. We have created our parser for HTTP request. Parser module extracts the information as per the client request.

SMS Parser: This is the most important module for SMS parsing. This is again our contribution to parse the SMS client request and extract the required information with the help of database mapping.

Database Mapping: This module is used for extracts the information from SQLite database and send back a response to the appropriate request manager. The request manager sends this response to request handler.

Response Handler: It is responsible for interpreting the result for HTTP client or SMS client then sending it back to the appropriate client. If the HTTP response then no problem, but if the SMS response then it requires more efforts.

Hosted Web Service: It's most important module in mobile web provisioning. It deploys the new Web services on mobile host and supports invoking of requested services. Once the Mobile Host was developed, it was extensively tested for performance issues like the memory load, server-processing load, Battery power consumption etc. The evaluation of the Mobile Host was conducted using services like Hospital search service, blood bank service developed for testing and some readymade services are also used, the location information service and weight converter service and etc. The test setup comprised a Mobile Host developed and deployed on the Android Samsung GT-S7562.

V. IMPLEMENTATION AND RESULT ANALYSIS

The proposed framework is an alternative solution deployment paradigm as a hybrid solution prototype, which gives an effective state-of-the-art web service provisioning on mobile device. We have developed a RESTful based web services for Android mobile host, that exposes multiple functionality as web service methods; each operation is represented by a unique URI in the form of http://192.168.43.1:8080/home.html. Initially two web services are deployed on Android mobile host and SQLite is used as database. Hospital and blood bank search web services are deployed on Android mobile host for testing.

The main objective of implementation is to analyses the performance of hybrid framework for resource constraint mobile device and check the feasibility of hybrid framework in mobile wireless environments. Due to HTTP and SMS support, it reduces the processing and communication overhead of resource constrained mobile device. The web service client request could be an HTTP request or SMS request. Requests listener, Request handler and response handler, HTTP parser and SMS parser, handle the request and generate the response for the web service consumer.

HTTP requests parse by HTTP parser and request are analyzed directly by a servlet which selects the appropriate class and methods to respond to requests. Once the execution is done, the response is sent to the response handler, where the response handler sends response to requested client. If a service request is received from SMS, the massage forwarded to SMS parser and broadcast receiver which select the appropriate class and method to respond the request in the form of SMS. Responses for web service requests are treated the same way except that they are put in a device-compatible form, if HTTP requests the uses same protocol for response or if a SMS request uses SMS as response before sending them to the requester.

Hybrid framework implementation and testing experimental setup required following components:

- Mobile Host (Service Provider): Samsung Galaxy S Duos S7562 Android Smartphone, 4GB internal memory, 1 GHz Cortex-A5 processor.
- Web service clients: Mobile phone, Tablet, Laptop, PC, Notebook
- Development Environment: Eclipse indigo SR-1 with Android SDK 14 integrated.
- Database used: SQLite
- Communication Environment: GPRS, EDGE-2 and SMS
- Web Services: Hospital Web Service, Blood Bank Web Service
- Tools : Apache JMeter, Battery Monitor

Hybrid framework has been experimented on Samsung Galaxy S Duos S7562 Android Smartphone and test bed consists of Android mobile devices as a mobile host and other devices are the Web service clients. Wen service clients are PC, Laptop and mobile phones. The Android mobile host with help of potable Wi-Fi is assigned a local IP address to the web service clients. We developed and deployed two web services on android mobile i.e. Hospital and Blood bank search web service. SQLite database is used to store the
hospitals and blood banks data. The web service testing client directly sends an HTTP request using a URL or Send SMS to Android mobile host. Hospitals and blood banks information stored in a SQLite database that is integrated into the Android Web service mobile host and it expose requested information to HTTP client or SMS client.

Figure 5 shows the sequence diagram, it illustrates the exact communication sequence between the web service clients and Android Web Service Provider.

Figure 5. Illustrates detail communication between clients and mobile host

After the starting the web service provider (Mobile Host), it is continuously in listening mode. When the client sends a HTTP request to Android mobile host, MainServer module executes and forward the request to the URL parser through Response class and parser send reply back to the response class. Now response class extracts the expected data from the database. Database fetches data from SQLite database and send back to response class. Response class populates the data on the client. If the request is in the form of SMS then SMS request forwarded to broadcast receiver. Broadcast receiver extract and forward SMS contains to the SMS parser, parser parse the SMS and forwarded to database mapping module to fetch the data from SQLite database and send response to back to the broadcast receiver. Broadcast receiver sends response as a SMS to client.

Smartphones are very powerful platform to allow the user to create various new services, not limiting user to use Smartphone only for voice call and SMS servicing. Now Smartphone is used as a web service provider. Provisioning of continuous web service requires battery power. Since Smartphones are battery driven, they
have limitations in energy consumption. Processing of the web service request and communicate response to the client, it requires energy. If light weight frameworks are developed, it reduces resource consumptions of mobile phone and RESTful based web services reduces processing and communication time. Lightweight framework, RESTful based web services and heterogeneous accessing nature of framework, reduces battery consumptions. The energy consumption is directly proportional to operational time, which is one of the most important issues while deploying web service on smart mobile phones. Therefore energy consumption in Smartphones is very important when deploying and providing web service from mobile phone. Our proposed Hybrid framework reduces battery power consumption, due SMS based web service request and REST based web service provisioning. Moore's law states every 18 months double the power computing devices. Smartphones are limited in terms of energy and power. It is crucial to understand these two terms that sometimes are used interchangeably.

Power is defined as the rate at which work is performed [30].

\[
\text{Power} = \text{work/Time [Watts]} \\
\]

Energy is defined as the time integral of Power.

\[
\text{Energy} = \text{Power * Time [Joules]} \\
\]

Power and energy plays a crucial role for evaluation of performance of web services from Smart mobile phones. Improvement of battery capacity is quite moderate as compared to Smartphone power computing capabilities and features provided by phone. Without disturbing the basic functionality of the Smartphone and it used as a web service provider, due to this other features drain the battery in a short period of time. To overcome this problem, energy consumption needs to be reduced; the solution for this is to use the Hybrid framework with REST based web services.

The mobile system consumes a power unit \( p_c = 0.9 \text{ watt/second} \) for computing and \( p_t = 1.3 \text{ watt/second} \) for sending or receiving data over a wireless Wi-Fi link based on the above scenarios. The two devices communicate over a portable hotspot Wi-Fi with a Round Trip Time (RTT) = 15.7ms and a data bandwidth \( B = 8.4 \text{ MB/Sec} \). We call each web service from different client 10 times and take the average readings of execution time, overall response time. We also calculate the energy consumption as a function of the web service execution time and data transfer between the mobile host and web service client as the following:

\[
E = T \times p_c + \frac{D}{BW} \times p_t \\
\]

Where \( E \) is the total energy consumption of a mobile host, \( T \) is the total CPU time of the mobile host, \( p_c \) is the power consumption unit of computation of web service response, \( p_t \) is the power consumption unit of data transfer, \( D \) is size of payload and \( BW \) is the link bandwidth. The android platform not providing much facility to measure energy consumption [25], but internal battery monitoring is based on a time interval. Some of recent research was on Smartphone battery consumption [27], but everyone using own setup for measuring battery consumption. For measuring the power consumption of the web services based on the Android open source project that is Power Profile developed in [26]. The power consumption estimate in framework is based on solely on computation.

VI. CONCLUSIONS

In this paper we proposed a Hybrid framework for hosting and provisioning RESTful Web services from mobile devices. After deploying web services on mobile host, it showed that it is feasible to mobile device worked as host in a dynamic environment without disturbing its basic functionality. Our framework is based HTTP and SMS, due its hybrid in nature framework provides an alternative way to access web services without GPRS service. The proposed framework is more suitable in the low range area and ruler places. This framework provides a service to a number of clients on time, so ultimately it increases the throughput of mobile host. The framework used REST based web service provisioning, REST generates lightweight payload, so it reduces the processing and communication time and SMS processing and communication requires very less power. From this it is proved that battery consumption is reduced. It is easier to build, deploy, publish and invoke web services on the Android Host in a wireless environment. Our framework has been tested for HTTP and SMS requests clearly outperformed and showed several advantages. These advantages are less request/response message size, less processing and communication power, less resource consuming, less battery consuming, more scalable and reduces the overhead on mobile host and SMS based web service is an attractive alternative for the implementation of web services on resource-constrained mobile devices.
REFERENCES


