INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATION & MANAGEMENT

IJRCM

A Monthly Double-Blind Peer Reviewed (Refereed/Juried) Open Access International e-Journal - Included in the International Serial Directories

Indexed & Listed at:

Ulrich's Periodicals Directory ©, ProQuest, U.S.A., EBSCO Publishing, U.S.A., Cabell's Directories of Publishing Opportunities, U.S.A.

as well as in Open J-Gage, India [link of the same is duly available at Inflibnet of University Grants Commission (U.G.C.)]

Registered & Listed at: Index Copernicus Publishers Panel, Poland

Circulated all over the world & Google has verified that scholars of more than 1500 Cities in 141 countries/territories are visiting our journal on regular basis.

Ground Floor, Building No. 1041-C-1, Devi Bhawan Bazar, JAGADHRI – 135 003, Yamunanagar, Haryana, INDIA

www.ijrcm.org.in

CONTENTS

CALL FOR MANUSCRIPTS

We invite unpublished novel, original, empirical and high quality research work pertaining to recent developments & practices in the area of Computer, Business, Finance, Marketing, Human Resource Management, General Management, Banking, Insurance, Corporate Governance and emerging paradigms in allied subjects like Accounting Education; Accounting Information Systems; Accounting Theory & Practice; Auditing; Behavioral Accounting; Behavioral Economics; Corporate Finance; Cost Accounting; Econometrics; Economic Development; Economic History; Financial Institutions & Markets; Financial Services; Fiscal Policy; Government & Non Profit Accounting; Industrial Organization; International Economics & Trade; International Finance; Macro Economics; Micro Economics; Monetary Policy; Portfolio & Security Analysis; Public Policy Economics; Real Estate; Regional Economics; Tax Accounting; Advertising & Promotion Management; Business Education; Management Information Systems (MIS); Business Law, Public Responsibility & Ethics; Communication; Direct Marketing; E-Commerce; Global Business; Health Care Administration; Labor Relations & Human Resource Management; Marketing Research; Marketing Theory & Applications; Non-Profit Organizations; Office Administration/Management; Operations Research/Statistics; Organizational Behavior & Theory; Organizational Development; Production/Operations; Public Administration; Purchasing/Materials Management; Retailing; Sales/Selling; Services; Small Business Entrepreneurship; Strategic Management Policy; Technology/Innovation; Tourism, Hospitality & Leisure; Transportation/Physical Distribution; Algorithms; Artificial Intelligence; Compilers & Translation; Computer Aided Design (CAD); Computer Aided Manufacturing; Computer Graphics; Computer Organization & Architecture; Database Structures & Systems; Digital Logic; Discrete Structures; Internet; Management Information Systems; Modeling & Simulation; Multimedia; Neural Systems/Neural Networks; Numerical Analysis/Scientific Computing; Object Oriented Programming; Operating Systems; Programming Languages; Robotics; Symbolic & Formal Logic and Web Design. The above mentioned tracks are only indicative, and not exhaustive.

Anybody can submit the soft copy of his/her manuscript **anytime** in M.S. Word format after preparing the same as per our submission guidelines duly available on our website under the heading guidelines for submission, at the email address: infoijrcm@gmail.com.

GUIDELINES FOR SUBMISSION OF MANUSCRIPT

1. **COVERING LETTER FOR SUBMISSION:**

DATED: _____

THE EDITOR
IJRCM

Subject: **SUBMISSION OF MANUSCRIPT IN THE AREA OF .**

(e.g. Finance/Marketing/HRM/General Management/Economics/Psychology/Law/Computer/IT/Engineering/Mathematics/other, please specify)

DEAR SIR/MADAM

Please find my submission of manuscript entitled '_____' for possible publication in your journals.

I hereby affirm that the contents of this manuscript are original. Furthermore, it has neither been published elsewhere in any language fully or partly, nor is it under review for publication elsewhere.

I affirm that all the author (s) have seen and agreed to the submitted version of the manuscript and their inclusion of name (s) as co-author (s).

Also, if my/our manuscript is accepted, I/We agree to comply with the formalities as given on the website of the journal & you are free to publish our contribution in any of your journals.

NAME OF CORRESPONDING AUTHOR:
Designation:
Affiliation with full address, contact numbers & Pin Code:
Residential address with Pin Code:
Mobile Number (s):
Landline Number (s):
E-mail Address:
Alternate E-mail Address:

NOTES:

a) The whole manuscript is required to be in ***ONE MS WORD FILE*** only (pdf. version is liable to be rejected without any consideration), which will start from the covering letter, inside the manuscript.
b) The sender is required to mention the following in the **SUBJECT COLUMN** of the mail:
New Manuscript for Review in the area of (**Finance/Marketing/HRM/General Management/Economics/Psychology/Law/Computer/IT/ Engineering/Mathematics/other, please specify)**
c) There is no need to give any text in the body of mail, except the cases where the author wishes to give any specific message w.r.t. to the manuscript.
d) The total size of the file containing the manuscript is required to be below **500 KB**.
e) Abstract alone will not be considered for review, and the author is required to submit the complete manuscript in the first instance.
f) The journal gives acknowledgement w.r.t. the receipt of every email and in case of non-receipt of acknowledgment from the journal, w.r.t. the submission of manuscript, within two days of submission, the corresponding author is required to demand for the same by sending separate mail to the journal.

2. **MANUSCRIPT TITLE:** The title of the paper should be in a 12 point Calibri Font. It should be bold typed, centered and fully capitalised.

3. **AUTHOR NAME (S) & AFFILIATIONS:** The author (s) **full name**, **designation**, **affiliation** (s), **address**, **mobile/landline numbers**, and **email/alternate email address** should be in italic & 11-point Calibri Font. It must be centered underneath the title.

4. **ABSTRACT:** Abstract should be in fully italicized text, not exceeding 250 words. The abstract must be informative and explain the background, aims, methods, results & conclusion in a single para. Abbreviations must be mentioned in full.

5. **KEYWORDS:** Abstract must be followed by a list of keywords, subject to the maximum of five. These should be arranged in alphabetic order separated by commas and full stops at the end.

6. **MANUSCRIPT:** Manuscript must be in ***BRITISH ENGLISH*** prepared on a standard A4 size ***PORTRAIT SETTING PAPER***. It must be prepared on a single space and single column with 1" margin set for top, bottom, left and right. It should be typed in 8 point Calibri Font with page numbers at the bottom and centre of every page. It should be free from grammatical, spelling and punctuation errors and must be thoroughly edited.

7. **HEADINGS:** All the headings should be in a 10 point Calibri Font. These must be bold-faced, aligned left and fully capitalised. Leave a blank line before each heading.

8. **SUB-HEADINGS:** All the sub-headings should be in a 8 point Calibri Font. These must be bold-faced, aligned left and fully capitalised.

9. **MAIN TEXT:** The main text should follow the following sequence:

INTRODUCTION

REVIEW OF LITERATURE

NEED/IMPORTANCE OF THE STUDY

STATEMENT OF THE PROBLEM

OBJECTIVES

HYPOTHESES

RESEARCH METHODOLOGY

RESULTS & DISCUSSION

FINDINGS

RECOMMENDATIONS/SUGGESTIONS

CONCLUSIONS

SCOPE FOR FURTHER RESEARCH

ACKNOWLEDGMENTS

REFERENCES

APPENDIX/ANNEXURE

It should be in a 8 point Calibri Font, single spaced and justified. The manuscript should preferably not exceed ***5000 WORDS***.

10. **FIGURES & TABLES:** These should be simple, crystal clear, centered, separately numbered &self explained, and **titles must be above the table/figure**. **Sources of data should be mentioned below the table/figure**. It should be ensured that the tables/figures are referred to from the main text.

11. **EQUATIONS:**These should be consecutively numbered in parentheses, horizontally centered with equation number placed at the right.

12. **REFERENCES:** The list of all references should be alphabetically arranged. The author (s) should mention only the actually utilised references in the preparation of manuscript and they are supposed to follow **Harvard Style of Referencing**. The author (s) are supposed to follow the references as per the following:

- All works cited in the text (including sources for tables and figures) should be listed alphabetically.
- Use (**ed.**) for one editor, and (**ed.s)** for multiple editors.
- When listing two or more works by one author, use --- (20xx), such as after Kohl (1997), use --- (2001), etc, in chronologically ascending order.
- Indicate (opening and closing) page numbers for articles in journals and for chapters in books.
- The title of books and journals should be in italics. Double quotation marks are used for titles of journal articles, book chapters, dissertations, reports, working papers, unpublished material, etc.
- For titles in a language other than English, provide an English translation in parentheses.
- The location of endnotes within the text should be indicated by superscript numbers.

PLEASE USE THE FOLLOWING FOR STYLE AND PUNCTUATION IN REFERENCES:

BOOKS

- Bowersox, Donald J., Closs, David J., (1996), "Logistical Management." Tata McGraw, Hill, New Delhi.
- Hunker, H.L. and A.J. Wright (1963), "Factors of Industrial Location in Ohio" Ohio State University, Nigeria.

CONTRIBUTIONS TO BOOKS

- Sharma T., Kwatra, G. (2008) Effectiveness of Social Advertising: A Study of Selected Campaigns, Corporate Social Responsibility, Edited by David Crowther & Nicholas Capaldi, Ashgate Research Companion to Corporate Social Responsibility, Chapter 15, pp 287-303.

JOURNAL AND OTHER ARTICLES

- Schemenner, R.W., Huber, J.C. and Cook, R.L. (1987), "Geographic Differences and the Location of New Manufacturing Facilities," Journal of Urban Economics, Vol. 21, No. 1, pp. 83-104.

CONFERENCE PAPERS

- Garg, Sambhav (2011): "Business Ethics" Paper presented at the Annual International Conference for the All India Management Association, New Delhi, India, 19–22 June.

UNPUBLISHED DISSERTATIONS AND THESES

- Kumar S. (2011): "Customer Value: A Comparative Study of Rural and Urban Customers," Thesis, KurukshetraUniversity, Kurukshetra.

ONLINE RESOURCES

- Always indicate the date that the source was accessed, as online resources are frequently updated or removed.

WEBSITES

- Garg, Bhavet (2011): Towards a New Natural Gas Policy, Political Weekly, Viewed on January 01, 2012 http://epw.in/user/viewabstract.jsp

COMPARATIVE STUDY ON THE FEATURES OF DIFFERENT WEB SERVICES PROTOCOLS

DHARA N. DARJI
ASST. PROFESSOR
DEPARTMENT OF COMPUTER SCIENCE
GANPAT UNIVERSITY
KHERVA

NITA B. THAKKAR
ASST. PROFESSOR
DEPARTMENT OF COMPUTER SCIENCE
GANPAT UNIVERSITY
KHERVA

ABSTRACT

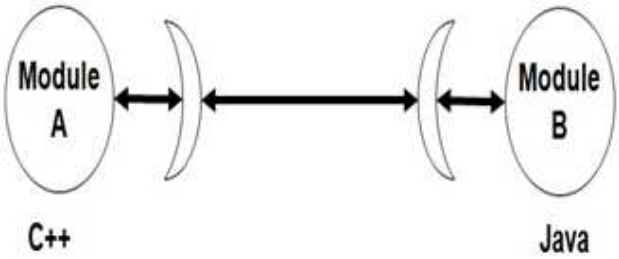
This article focuses on the basic idea of web services and its basic protocols. Basic architecture of SOAP & REST as well as the comparison of SOAP & REST.

KEYWORDS
DISCO, REST, SOAP, UDDI, WSDL.

WHAT IS WEB SERIVCES

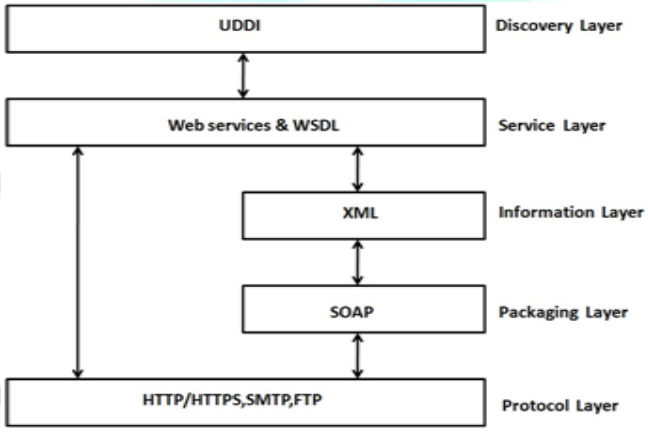
Web service is a software system designed to support interoperable machine-to-machine interaction over a network. Distributed components are interfaced via non-object-specific protocols

FIGURE 1: COMMUNICATION BETWEEN DIFFERENT APPLICATION [8]



In other words, Web Services is a software system that allows machines (including servers) to communicate with each other regardless of each individual machine's operating systems and programming languages. The Symons's Extensible Markup Language (XML) Page provides a very nice formula that neatly defines the major components of Web Services [1]:
Web services = XML + SOAP + WSDL + UDDI

FIGURE 2: WEB SERVICES STACK [8]



XML (EXTENSIBLE MARK-UP Language)
XML is an acronym for eXtensible Mark-up Language and was developed and finalized by the W3C in 1998. XML is a well formed, tree structured, plain text document that is human readable and machine consumable.
XML forms the basis for all modern Web services, which use XML-based technologies to describe their interfaces and to encode their messages. WSDL, SOAP, and UDDI all use XML-based messaging that any machine can interpret [1].

WSDL (Web Services Description Language)
WSDL is an XML-based format for describing Web services. Clients wishing to access a Web service can read and interpret its WSDL file to learn about the location of the service and its available operations. Through the WSDL, a Web services client learns where a service can be accessed, what operations the service performs, the communication protocols the service supports, and the correct format for sending messages to the service.
A WSDL file is an XML document that describes a Web service using six main elements:
Port type – groups and describes the operations performed by the service through the defined interface.

Port – specifies an address for a binding, i.e., defines a communication port.
Message – describes the names and format of the messages supported by the service.
Types – defines the data types (as defined in an XML Schema) used by the service for sending messages between the client and server.
Binding – defines the communication protocols supported by the operations provided by the service.
Service – specifies the address (URL) for accessing the service.
The WSDL document that describes a Web service acts as a contract between Web service client and server. By adhering to this contact the service provider and consumer are able to exchange data in a standard way, regardless of the underlying platforms and applications on which they are operating [1].

UDDI (Universal Description Discovery and Integration)

UDDI is a standard sponsored by OASIS (Organization for the Advancement of Structured Information Standards). Often described as the yellow pages of Web services, UDDI is a specification for creating an XML-based registry that lists information about businesses and the Web services they offer. UDDI provides businesses a uniform way of listing their services and discovering services offered by other organizations. Though implementations vary, UDDI often describes services using WSDL and communicates via SOAP messaging. Registering a Web service in a UDDI registry is an optional step, and UDDI registries can be public or private (i.e. isolated behind a corporate firewall). To search for a Web service, a developer can query a UDDI registry to obtain the WSDL for the service he/she wishes to utilize. Developers can also design their Web services clients to receive automatic updates about any changes to a service from the UDDI registry [1].

SOAP (Simple Object Access Protocol)

SOAP is one of XML-based web service's messaging protocols, which is used for information exchange in distributed computing environment. SOAP defines mechanisms for message transferring between distributed system and remote procedure call (RPC) processes. Web services use SOAP to send messages between a service and its client(s). Because HTTP is supported by all Web servers and browsers, SOAP messages can be sent between applications regardless of their platform or programming language. This quality gives Web services their characteristic interoperability.
Data is sent between the client(s) and the Web service using request and response SOAP messages, the format for which is specified in the WSDL definition. Because the client and server adhere to the WSDL contract when creating SOAP messages, the messages are guaranteed to be compatible [1].

Anatomy of web services

Basic Architecture of web services

The basic architecture includes Web services technologies capable of [6]:

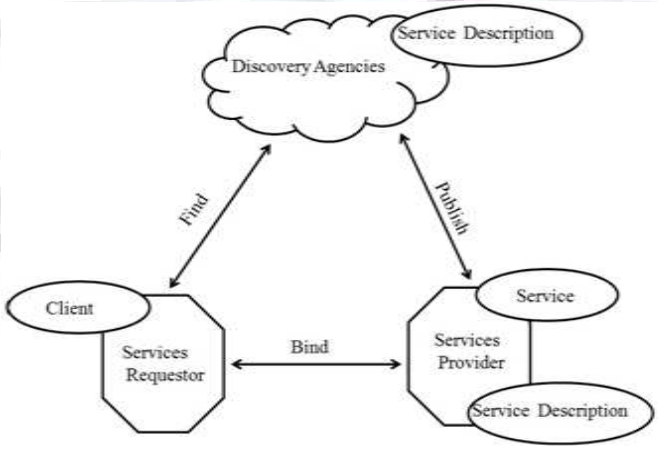
- Exchanging messages
- Describing Web services
- Publishing and discovering Web service descriptions

The basic Web services architecture defines an interaction between software agents as an exchange of messages between service requesters and service providers. Requesters are software agents that request the execution of a service. Providers are software agents that provide a service. Agents can be both service requesters and providers. Providers are responsible for publishing a description of the service(s) they provide. Requesters must be able to find the description(s) of the services.
The basic Web service architecture models the interactions between three roles: the service provider, service discovery agency, and service requestor. The interactions involve to publish, find, and bind operations. These roles and operations act upon the web service artifacts: the web service software module and its description. In a typical scenario a service provider hosts a network accessible software module (an implementation of a web service). The service provider defines a service description for the web service and publishes it to a requestor or service discovery agency. The service requestor uses a find operation to retrieve the service description locally or from the discovery agency (i.e. a registry or respository) and uses the service description to bind with the service provider and invoke or interact with the web service implementation. Service provider and service requestor roles are logical constructs and a service may exhibit characteristics of both.
Requesters and providers interact using one or more message exchange patterns (MEPs) that define the sequence of one or more messages exchanged between them. A service description is hosted by a discovery service, to which a provider publishes the description, and from which the requester discovers the description. The description includes data type and structure information, identifies the MEP, and contains the address of the service provider.
The extended architecture describes Web services support for MEPs that group basic messages into higher-level interactions, details how support for features such as security, transactions, orchestration, privacy and others may be represented in messages (SOAP modules), and describes how additional features can be added to support business level interactions. The extended architecture builds on the basic architecture using the extensibility mechanisms inherent in the basic technologies.
Software agents in the basic architecture can take on one or all of the following roles:

- Service requester -- requests the execution of a Web service
- Service provider -- processes a Web service request
- Discovery agency -- agency through which a Web service description is published and made discoverable

A software agent in the Web services architecture can act in one or multiple roles, acting as requester or provider only, both requester and provider, or as requester, provider, and discovery agency. A service is invoked after the description is found, since the service description is required to establish a binding.

FIGURE 3: WEB SERVICES ACTORS, OBJECTS & OPERATION [6]



Features of Web services

Web services provide several technological and business benefits, a few of which include:

- Application and data integration
- Versatility
- Code re-use

- Cost savings

The inherent interoperability that comes with using vendor, platform, and language independent XML technologies and the ubiquitous HTTP as a transport mean that any application can communicate with any other application using Web services. The client only requires the WSDL definition to effectively exchange data with the service – and neither part needs to know how the other is implemented or in what format its underlying data is stored. These benefits allow organizations to integrate disparate applications and data formats with relative ease. (Web services provide interoperability between various software applications running on disparate platforms/operating systems)

(Web services allow software and services from different companies and locations to be combined easily to provide an integrated service.)

Web services use open standards and protocols

Web services are also versatile by design. They can be accessed by humans via a Web-based client interface, or they can be accessed by other applications and other Web services. A client can even combine data from multiple Web services to, for instance, present a user with an application to update sales, shipping, and ERP systems from one unified interface – even if the systems themselves are incompatible. Because the systems exchange information via

Web services, a change to the sales database, for example, will not affect the service itself.

Code re-use is another positive side-effect of Web services' interoperability and flexibility. One service might be utilized by several clients, all of which employ the operations provided to fulfill different business objectives. Instead of having to create a custom service for each unique requirement, portions of a service are simply re-used as necessary. All these benefits add up to significant cost savings. Easy interoperability means the need to create highly customized applications for integrating data, which can be expensive, is removed. Existing investments in systems development and infrastructure can be utilized easily and combined to add additional value. Since Web services are based on open standards their cost is low and the associated learning curve is smaller than that of many proprietary solutions. Finally, Web services take advantage of ubiquitous protocols and the Web infrastructure that already exists in every organization, so they require little if any additional technology investment. Web services allow the reuse of services and components within an infrastructure **.**Web services are loosely coupled thereby facilitating a distributed approach to application integration **.**

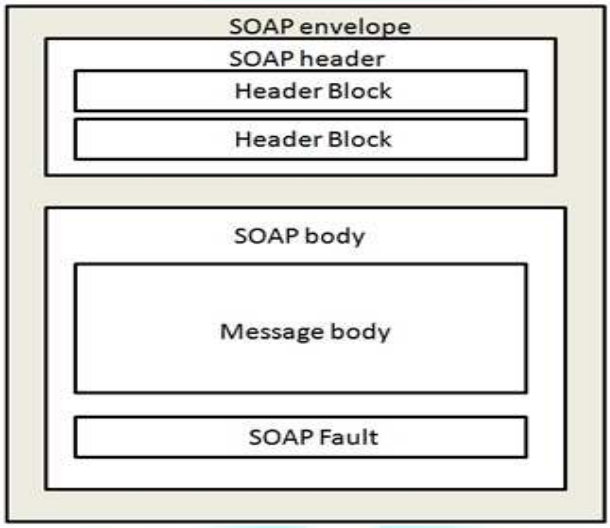
What is soap

Architecture of SOAP message

SOAP messages are XML documents that contain some or all of the following elements [2]

- Envelope – specifies that the XML document is a SOAP message; encloses the message itself.
- Header (optional) – contains information relevant to the message, e.g., the date the message was sent, authentication data, etc.
- Body – includes the message payload.
- Fault (optional) – carries information about a client or server error within a SOAP message.

FIGURE 4: DIAGRAM OF A SOAP MESSAGE [3]



Advantages of the SOAP

- Supported by the popular companies such as IBM & Microsoft and also supported by open source communities
- Simple & easy to understand because the messages of SOAP is in XML
- SOAP is scalable for transporting messages because its uses HTTP protocol
- SOAP is Platform independent & portable[7]

Disadvantages of the SOAP

- SOAP is slow because of parsing of envelop all time.
- SOAP coding is complicated. Performing SOAP request is quite detailed. The
- SOAP programmer has to construct a message, put the arguments in the message, and then send it. After that, he has to wait for reply, parse the reply XML message, and find out the data.
- The roles of interacting parties are fixed, when relying on HTTP as a transfer protocol and not using Web services addressing. Only one party (the client) can use the services of the other. So developers must use polling instead of notification in these common cases.
- SOAP toolkit is always needed for the developers to form requests and parse the results.
- There is no standard way to dynamically discover the services (for example, parameters, methods).
- It is depending on WSDL which is an external standard [7].

What is REST?

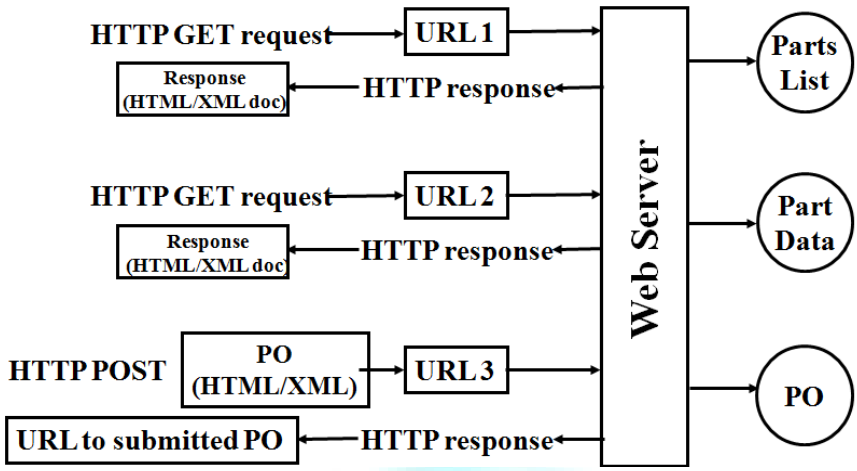
REST defines a set of architectural principles by which you can design Web services that focus on a system's resources, including how resource states are addressed and transferred over HTTP by a wide range of clients written in different languages.

The acronym REST stands for *Representational State Transfer*, this basically means that each unique URL is a representation of some object. To get the contents of that object using an HTTP GET, to delete it, use a POST, PUT, or DELETE

- Resources are identified by uniform resource identifiers (URIs)
- Resources are manipulated through their representations
- Messages are self-descriptive and stateless
- Multiple representations are accepted or sent

REST way of designing the web services

FIGURE 5: WEB SERVICES DESIGN USING REST



The web service makes available a URL to a parts list **resource**. A client uses this URL to get the parts list [4]
This type of design,

- Create a resource for every service.
- Identify each resource using a URL.

Advantages of the REST

- Main advantage for the REST is that, it is easy technology, which can be used to create web services. Using REST gives more benefit to the users. In order to implement web services developers need to know HTTP, HTML and XML and they can start building Web services right away, without needing any toolkit.
- Another benefit of the REST interface is that requests and responses can be short.
- REST developers can easily create and modify an URI to access different Web resources.
- With the support for caching, REST provides improved response times and server loadings.
- By reducing the need to maintain communication state, REST improves server scalability. This means that initial and subsequent requests can be handled in different servers.
- Since a single browser can access any resource and application, REST demands less client-side software to be written than other approaches.
- With the use of hyperlinks in content, a separate resource discovery mechanism is not needed [7].

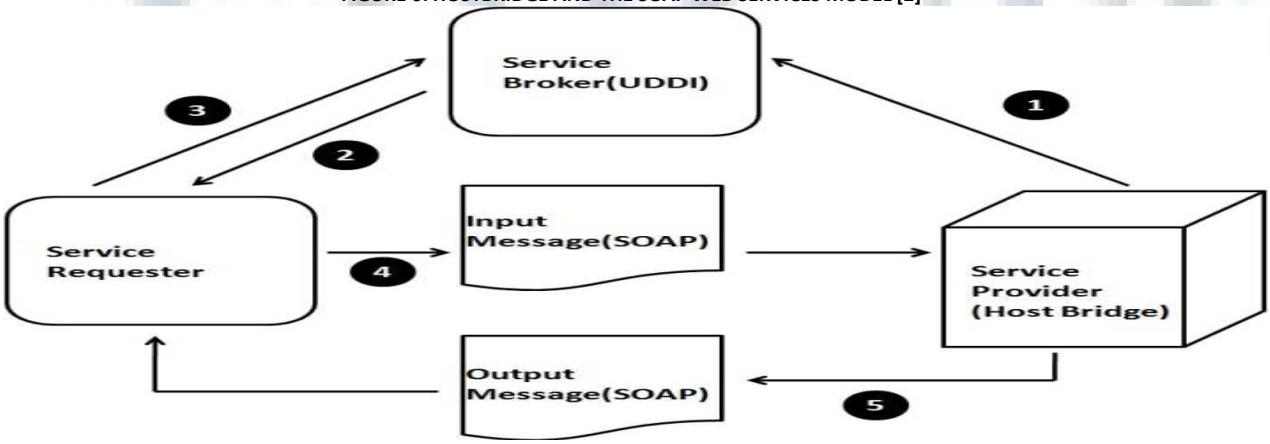
Disadvantages of the REST

- REST is a good solution for most Web service, but not the best, because in REST data is sent as parameters in URIs. This is not very good. The large amounts of data such as purchase orders can be too big and too complicated for an URI.
- Probably the most important debate is the security area. REST calls go over HTTP that creates not reliable messaging.[7]

Comparison of SOAP & REST

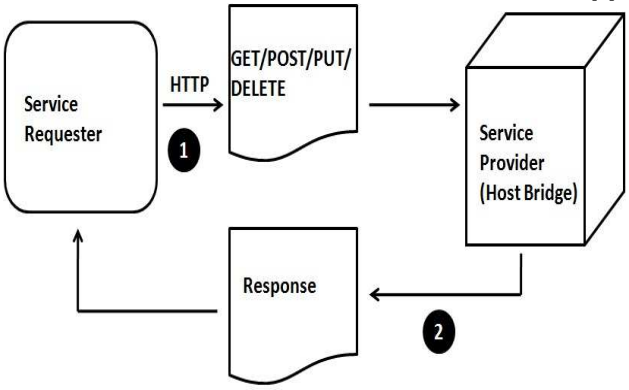
SOAP	REST
Focus on design of integrated (distributed) applications Designed to handle distributed computing environments[2]	**Focus on scalability and performance of large scale distributed hypermedia systems** Assumes a point-to-point Communication model--not usable for distributed computing environment where message may go through one or more Intermediaries[2]
Activity/Service oriented Conceptually more difficult, more "heavy-weight" than REST Harder to develop, requires tools[2]	**Resource oriented** Simpler to develop than SOAP Small learning curve, less reliance on tools Concise, no need for addition al messaging layer[2]
Lack of standard naming mechanism The prevailing standard for web services, and hence has better support from other standards (WSDL, WS-*) and tooling from vendors	**URI: Consistent naming mechanism for resources** Lack of standards support for security, policy, reliable messaging, etc., so services that have more sophisticated requirements are harder to develop
Who's using REST? All of Yahoo's web services use REST, including Flickr, del.icio.us API uses it, pubsub, bloglines, technorati, and both eBay, and Amazon have web services for both REST and SOAP.	**Who's using SOAP?** Google seams to be consistent in implementing their web services to use SOAP, with the exception of Blogger, which uses XML-RPC. You will find SOAP web services in lots of enterprise software as well.

FIGURE 6: HOSTBRIDGE AND THE SOAP WEB SERVICES MODEL [2]



1. REST has no WSDL interface definition
2. REST is over HTTP, but SOAP can be over any transport protocols such HTTP, FTP, STMP, JMS etc.
3. SOAP is using soap envelope, but REST is just XML.

FIGURE 7: HOSTBRIDGE AND THE RESTFUL WEB SERVICES MODEL [2]



REFERENCES

1. An article on Web services: Benefits, challenges, and a unique, visual development solution" by Erin Cavanaugh Available at: http://www.altova.com/whitepapers/webservices.pdf
2. A paper on "SOAP and REST Choosing formal and informal Web services for CICS integration " by hostbridge technology in 2009 Available at: http://www.hostbridge.com/images/uploads/WP_HostBridge_SOAP_and_REST_090303.pdf
3. A thesis on "SOAP Web Services" by Brian Suda submitted to University of Edinburgh in 2003 Available at: http://suda.co.uk/publications /MSc/brian.suda.thesis.pdf
4. An article on "REST vs. SOAP Making the Right Architectural Decision" by Cesare Pautasso from University of Lugano (USI), Switzerland Available at: http://www.jopera.org/files/soa-amsterdam-restws-pautasso-talk.pdf
5. A book on "programming web services with SOAP" published by O'Reilly
6. A paper on "Introduction to Web services architecture" by K. Gottschalk,S. Graham H.,Kreger, J. Snell",2002 Available at: http://www.pa.icar.cnr.it/ cossentino/ICT/doc/D30-Introductionto WebService Architecture.pdf
7. A book on "Web service implementation with SOAP and REST Group" by Kristina Jucyte,Karolis Kevelaitis,Sung Won Park,2006 Available at http://rudar.ruc.dk/bitstream/1800/2108/1/Web%20services%20%20SOAP%20%26%20REST.pdf
8. An article on “web services” by mahmoud rabie at EGJUG, 2006

REQUEST FOR FEEDBACK

Dear Readers

At the very outset, International Journal of Research in Computer Application and Management (IJRCM) acknowledges & appreciates your efforts in showing interest in our present issue under your kind perusal.

I would like to request you to supply your critical comments and suggestions about the material published in this issue as well as on the journal as a whole, on our E-mail **infoijrcm@gmail.com** for further improvements in the interest of research.

If you have any queries please feel free to contact us on our E-mail **infoijrcm@gmail.com**.

I am sure that your feedback and deliberations would make future issues better – a result of our joint effort.

Looking forward an appropriate consideration.

With sincere regards

Thanking you profoundly

Academically yours

Sd/-

Co-ordinator

ABOUT THE JOURNAL

In this age of Commerce, Economics, Computer, I.T. & Management and cut throat competition, a group of intellectuals felt the need to have some platform, where young and budding managers and academicians could express their views and discuss the problems among their peers. This journal was conceived with this noble intention in view. This journal has been introduced to give an opportunity for expressing refined and innovative ideas in this field. It is our humble endeavour to provide a springboard to the upcoming specialists and give a chance to know about the latest in the sphere of research and knowledge. We have taken a small step and we hope that with the active co-operation of like-minded scholars, we shall be able to serve the society with our humble efforts.

Our Other Journals