

Design & Development of a REST based Web Service Platform for Applications Integration on Cloud

Ritesh Sinha¹, Manisha Khatkar², Subhash Chand Gupta³

¹ Computer Science, Amity University,
Noida, Uttar Pradesh, India
riteshsinha9891@gmail.com

² Computer Science, Amity University,
Noida, Uttar Pradesh, India
manisha407@gmail.com

³ Computer Science, Amity University,
Noida, Uttar Pradesh, India
scgupta@amity.edu

Abstract

Web services are used to communicate over internet by many different organizations to achieve their business goals. Web services are used in many areas like software organizations, medical care and banking services by using cloud. Earlier web services used SOAP but now REST have replaced which is used to convey information using HTTP in addition with XML standards. The web services are used by completely different applications to communicate data using cloud. GET, POST, PUT and DELETE methods can be used for these purposes.

Keywords: Web Service, SOAP, WSDL, HTTP, XML, REST, GET, PUT, DELETE, POST.

1. Introduction

The paper explores a relatively new approach REST based method to communicate and transfer data from one application to another using cloud platform such as Force.com platform of Salesforce. The SOAP, XML and JSON standards are used to interact with each other over cloud. These services are very cheap and dynamic. The Web services can be implemented in many different architectures and forms as suitable. As REST is light in weight as compared to SOAP and is based on url, many companies have used REST such as Amazon, Yahoo, eBay and Flickr.

The purpose of this paper is to Design & Development of a REST based Web service platform for applications integration on Cloud. This paper is intended to integrate applications on cloud environment to another application on different platform/web server by using RESTful web services. The coding for this purpose is done in Apex language, the propriety language of Salesforce.

2. About Apex Language

It is a strongly type written, developers execute flow and transaction control and management statements as it is object oriented programming language on the Force.com platform. The syntax used by Apex is much like Java and .Net, Apex allows developers to create business logic and, as well building Visualforce pages for designing applications and softwares. Visual force page is very similar to HTML as in traditional Net Beans or Dot Net. Triggers are used to initiate Apex code.

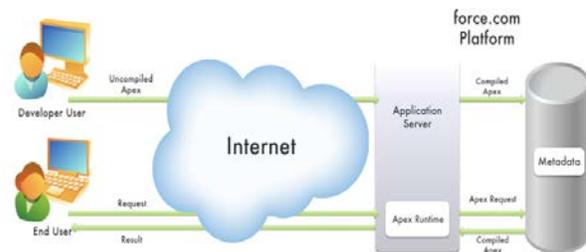


Fig. 1 How Does Apex Work

3. Background of Web Services

Cloud Services are now being adopted by many Business organisations and IT enterprises because cloud offers many services to these organisations like SaaS(Software as a Service), PaaS(Platform as a Service), and IaaS(Infrastructure as Service) where these organisations

need not to worry about setting up the whole infrastructure in order to start the company, whatever is needed is offered by these service providers what we call as on demand services where the client pays for each and every services he uses. When there is no need of the the services, the client can stop paying for that specific service, This type of model is called as licensing model, where client pay for each and every license he buys for specific time period. Hence there is no need to purchase the software or infrastructure as needed in traditional architectures.

In Cloud Computing, RESTful services are replacing SOAP based services in communications over cloud and internet. The word form REST stands for representational state transfer, this primarily means every distinctive uniform resource locator could be a illustration of some object. You'll get the contents of that object by a hypertext transfer protocol GET, you then may use a POST, PUT or DELETE to change the object (in real use most of the services use a POST for this). Therefore REST could be a new way to produce and integrate web services, whose main benefits are: being light-weight, clear and human readable results, simple to create services (no toolkits required). Though REST continues to be generating discussion regarding its implementations, and many different proposals are suggested, it provides enough mechanisms to permit knowledge-representations sharing among heterogeneous intelligent services.

4. Problem Outline

In this paper, the requirement is to integrate the Salesforce, a cloud platform with another application using Web services. The goal of this project is to integrate any application with cloud using web service as a middleware to send data and providing IaaS(Integration as a Service) on demand by the client.

There is no need to setup a VPN connection by the users in cloud computing, The client feels that he is master of the services he is using because service providers integrate these services on the clients side or what we call on-premise systems, between cloud applications and integration between two applications using these web services of cloud.

The Salesforce provides CRM Software-as-a-Service virtually on Cloud.

5. Research Objective

The objective of this project is Design and Development of a REST based Web service platform to integrate Salesforce Objects to Client Application, means they are able to do synchronous communication. In this project, a

completely unique way to integrate intelligent Web-services are going to be designed and developed, and therefore the resulting system are going to be deployed within the domain of recommendation.

Some part of the project work has been carried out within the context of Client Application, and has been deployed to integrate a set of applications to create a virtual space to support innovation processes.

This web services platform under SOA promise interoperability benefits which has resulted in a new impetus to SaaS integration on the Cloud. In the Research Objective, the required and desired objectives are identified with some challenges due to the dependence on partner services. The develop platform must meet both the objectives and ready to take all the challenges.

6. Methods and Models Used

The Integration Coding for the proposed modules is done at the Salesforce side using the Apex language.

- It contains a Class that uses the URLs of the Client application which needs to integrated using Web Service.

- This Class actually collects the project object information from Client side and puts the data from Salesforce object to Clients Application.

- Similarly a Class can be developed to update or edit data from Clients Application to Salesforce using a Refresh button.

- A Trigger is used to update or insert in the object of Salesforce which updates Clients Application information immediately.

The Model chosen for coding is MVC. It uses three tier architecture namely Model, View and Controller.

MVC pattern contains below three modules :-

Model

sObjects are the model as each entity is related to some sObjects.

Salesforce uses these to represent the whole system and schema.

View

Visualforce is used to depict the front end to users. It presents the data to users as HTML in traditional .Net or Net beans.

Controller

These are the classes that are used to represent the actions when users interact with Visualforce page and trigger an action.

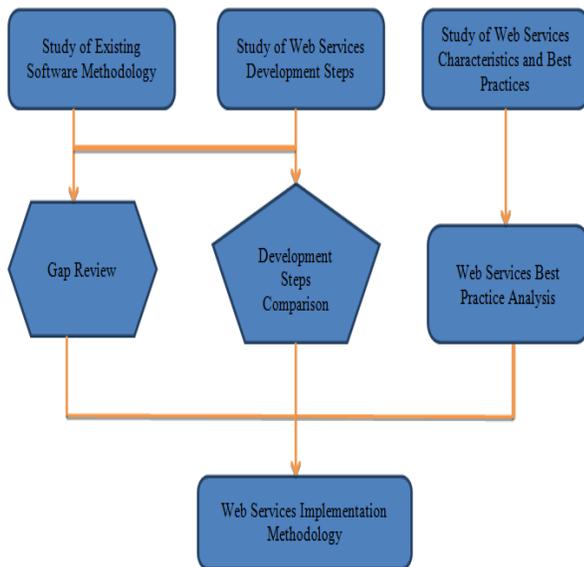


Fig. 2 Research Methodology

7. Scope of Work

This Salesforce- Application integration provides instant extendibility with synchronous communication and will be based on the service oriented architecture (SOA), which uses the RESTful services and Java Script Object Notation(JSON) Standard for data transportation or data exchange over cloud using web service, so that data can be transferred in a form that is human readable. The uniform interface in such implementation is obtained through HTTP verbs or methods: POST, GET, PUT and DELETE, which defines a CRUD (Create, Retrieve, Update, Delete) interface for any REST resource. Resources are identified by means of URI's which provide with syntax to build unique identifiers.

The Lifecycle of Web Service implementation typically, would undergo, from the identification of the requirement of the Web Service to the ultimate deployment and usage by the end-users. The phases known to be relevant within the Web Service Implementation Lifecycle are: requirements phase, analysis phase, designing phase, coding phase, testing and deployment phase. In each of the phases, Web Service specific activities are carried out and applied.

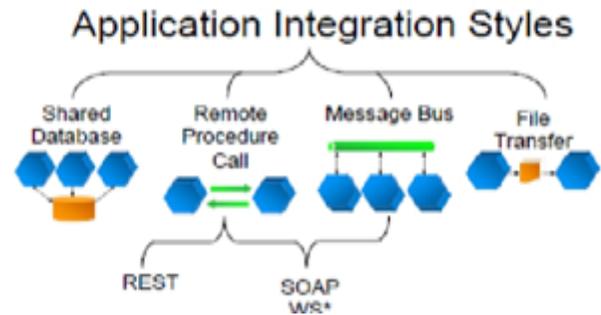


Fig. 3 Application Integration Styles

Using the REST Framework, the Resources are manipulated by means of representations transfers through a uniform interface addressed by the resource identifier. Each representation has to implement a CRUD interface to achieve the uniform interface required by the REST architecture. HTTP/1.1 protocol defines GET, PUT, POST and DELETE HTTP methods and corresponding CRUD actions in the following table.

Table 1: Showing HTTP methods

Method	CRUD	Description
POST	Create	Creates a new resource
GET	Retrieve	Retrieve a representation
PUT	Update	Update a resource
DELETE	Delete	Delete a resource

The HTTP status codes as per the HTTP protocol define in the following table.

Table 2: Showing HTTP Status Codes

Status	Codes	Description
Success	200 OK	Successful read, update or delete
	201 Created	Successful create
Error or Exception Codes	400 Bad Request	Unrecognized or invalid request URL

	<u>401 unauthorized</u>	<u>Credentials not provided or are incorrect</u>
	<u>422 un-processable entity</u>	<u>Unsuccessful create, update or delete</u>
	<u>404 Not Found</u>	<u>Record does not exist or you do not have permission to access it</u>
	<u>500 Internal Server Error</u>	<u>Unexpected Error</u>

Conclusion

In this project, an Apex class was developed for Synchronous Web service which is not permitted using Apex triggers.

The Apex language is used to do an Integration Coding for the proposed modules at the Salesforce side in this project. It contains a Class that uses the URLs of the Client Application which needs to be integrated using Web Service.

- This Class actually collects the project object information from Client Application and puts the data from Salesforce object to Client Application.

- Similarly a Class can be developed to update or edit or send data from Client Application to Salesforce using a Send button or Refresh button.

- A Trigger is used to update or insert in the object of Salesforce which updates Client Application information immediately.

Hence “Synchronous Communication” has been achieved in this project work.

Testing of code developed using Salesforce.com Platform can be done. These test code can cover both positive as well as negative test cases. Approximately 75 percent of the code must be covered by a test code to be able to qualify as tested by that unit test code. These are termed as code coverage in Salesforce.com

Acknowledgments

I consider it a privilege to express my gratitude and respect to all those who guided and inspired us in the completion of this paper.

I would like to thank Mr. Subhash Chand Gupta who helped me as mentor and guide.

I am also very thankful to Dr. Abhay Bansal, HOD of Computer Science and Engineering of Amity University. The support and encouragement provided by them has been a key factor in the successful completion of this paper.

The support provided by Amity University in terms of infrastructure is invaluable. It was a pleasure working at the laboratory.

References

- [1] PirnauMironela, “The Importance of Web Services using the RPC and REST Architecture”, IEEE, International Conference on Computer Technology and Development, 2009.
- [2] Khaldoon Al-Zoubi, Gabriel Wainer, “Using REST Web-Services Architecture for Distributed Simulation”, IEEE, SCS Workshop on Principles of Advanced and Distributed Simulation, 2009.
- [3] Steve Fisher, “The Architecture of the Apex Platform, salesforce.com’s Platform for Building On-Demand Applications”, IEEE, 29th International Conference on Software Engineering, 2007.
- [4] SiewPoh Lee, Lai Peng Chan, EngWah Lee, “Web Services Implementation Methodology for SOA Application”, IEEE, 2006.
- [5] K. Mukherjee and G.Sahoo, “A Secure Cloud Computing”, IEEE, International Conference on Recent Trends in Information, Telecommunication and Computing, 2010.
- [6] R.Nagappan, R. Skoczylas, R. Sriganesh, “Developing Java Web Services”, Wiley 2003.
- [7] KishorWagh, “A Comparative study of SOAP Vs REST Web Services Provisioning Techniques for Mobile Host”, Journal of Information Engineering and Applications, ISSN 2225-0506 (Online), Vol 2, No. 5, 2012.
- [8] CesarePautasso, Olaf Zimmermann, Frank Leymann, “RESTful Web Services vs. ‘Big’ Web Services: Making the Right Architectural Decision”, IEEE, International Conference on World Wide Web, 2008.
- [9] Leonard Richardson, Sam Ruby and David Heinemeier, “RESTful Web Services. First edition”, O’Reilly Media, 2007.
- [10] HatemHamad, MotazSaad, and Ramzi Abed, “Performance Evaluation of RESTful Web Services for Mobile Devices”, International Arab Journal of e-Technology, Vol.1, No. 3, January 2010.
- [11] Michael zurMuehlen, Jeffrey V. Nickerson, and Keith D. Swenson, “Developing Web Services Choreography Standards - The Case of REST vs. SOAP”, 2004, Elsevier B.V.
- [12] Eric Newcomer, “Understanding Web Services: XML, WSDL, SOAP and UDDI”, Addison-Wesley Professional (May 23, 2002).
- [13] LudmilaCheraksova, Yun Fu, Wenting Tang, and Amin Vahdat, “Measuring and Characterizing End-to-End Internet Service Performance”, ACM Transactions on Internet Technology, Vol. 3, 4.

- [14] Dr. Roy Thomas Fielding - doctoral thesis, "Architectural Styles and the Design of Network-Based Software Architecture", Department of Computer and Information Science,
- [15] David Booth, Hugo Haas, Francis McCabe, et al., "Web Services Architecture," <http://www.w3.org/TR/ws-arch/>, last accessed April 20, 2013.
- [16] EranChinthaka, "Enable REST with Web Services, Part 1: REST and Web Services in WSDL 2.0", April 20, 2013.
- [17] Alex Rodriquez, "RESTful Web Services: The Basics", April 21,2013.
- [18] Yutu Liu, Anne H. H. Ngu, and LiangzhaoZeng, "QoS Computation and Policing in Dynamic Web Service Selection", 2004, May 17-22, New York, USA.
- [19] Varun Goyal, "Web Services in Mobile Devices", Computer Science Department, Rochester Institute of Technology, 2013.
- [20] Dr. Atul Gonsai and Mr. Rushi Raval, "Mobile Cloud Computing: A Tool for Future", IJCSET, 2013.
- [21] Dejan Kovachev, Yiwei Cao and Ralf Klamma, "Mobile Cloud Computing: A Comparison of Application Models", Information Systems & Database Technologies RWTH Aachen University, 2013.

Ritesh Sinha currently persuing M.Tech(CSE) from Amity University, Noida, Uttar Pradesh.

Manisha Khatkar currently persuing M.Tech(CSE) from Amity University, Noida, Uttar Pradesh.

Subhash Chand Gupta currently working as Assistant Professor in Amity University, Noida, Uttar Pradesh.