www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 2 Issue 5 May, 2013 Page No. 1449-1452

PERFORMANCE ANALYSIS OF WEB SERVICES BY USING J METER PERFORMANCE TOOL

Nitin Kumar Verma*, Dr. Neeta Singh, Rishi Gautam
School of ICT,
Gautam Buddha University Yamuna Expressway Greater Noida, U.P, India
nitin_kumar_verma@yahoo.co.in
School of ICT
Gautam Buddha University Yamuna Expressway Greater Noida, U.P, India
neetasingh@gbu.ac.in
School of ICT
Gautam Buddha University Yamuna Expressway Greater Noida, U.P, India
rgautam107@gmail.com

Abstract--In this paper we have discussed web services performance and compares the performance differences of web services development approaches in java, Soap API and business logic EJB, is use POJO for access request and pass this object request with data server or server, response back to client which is time consuming in middle layer. Hibernate gives the solution by mapping data directly with data server or server. After developed that application, we have analyzed web services performance by using Apache J Meter web services performance analysis tool that configure IP request address of web services in the form of SOAP xml format and get a response from web services in term of throughput, standard deviation, error, average, median. WSDL contains the information like request for service, IP address and port number of called web services. Performance analyses result can be used to make appropriate choice for developing and optimizing mechanism to web services environment.

Keywords:- Simple Object Access Protocol(SOAP), Web Service Description Language(WSDL), Extensible Markup Language(XML), Enterprises Java Beans(EJB), Service Oriented Architecture(SOA), Plain Old Java Object(POJO).

I. INTRODUCTION

In today's world of competition on performance, the web service performance play a role to increase an access of resources over an Internet. Lot of research is to done for accessing web services in efficient manner. Web services moving from legacy technology like CORBA, RMI and advanced technology like EJB, Hibernate . Latest technology need to combine with web services which gives better interoperability, Fast access over a internet.

Web services are client and server applications that make network communication over the World Wide Web's(WWW), intranet, HyperText Transfer Protocol (HTTP). As described by the World Wide Web Consortium (W3C) guidelines, web services[4] provide a standard means of interoperating between Software applications running on a remote or different platforms and frameworks. Web services can be characterized by its great interoperability and reusability, as well as their machine process able descriptions, we can give thanks to the use of XML to parse data over a soap protocol. Web services[3] can be combined in a loosely coupled way to achieve complex operations. Programs providing simple services can interact with each other to deliver sophisticated added-value services. Hibernate is for ORM (object relational mapping) that is, make objects persistent in data server. It may be used also as a AOP, Dependency Injector, a Web Application and ORM among other things. So, if only need ORM, just use Hibernate. Time will come, Software [7] systems are rapidly increasing in complexity that will cause by their growing size, multi-functionality, multimodal user interaction, higher quality requirements, degree of distribution, dynamic execution environments, mobility, etc. In particular the distribution across heterogeneous networks over different processes, nodes, control units or components makes reactive systems difficult to develop and maintain. Multi-functionality increases this challenge – the purpose of the systems magnifies and in combination with distribution leads to complex functional dependencies, feature interactions and inefficient architectures. A[5] promising paradigm that offers solutions to several of the mentioned problems is service-oriented development. Most generally, a service can be understood as a coherent, purpose oriented unit of behavior. Thus, SOA [2] provides a method to modularize and structure the functionality of a system, dividing its behavior from a purpose-oriented point of view. Different module at different platform use remote services over communication network easily available to us.

Web services architecture[14] describes how to create and access web services in figure shows below

Publish: Only services that can be found and used by users can really play its role, so service providers[11] must describe and publish services to the registered server firstly. In the publish operation, service providers need to register the server authentication in this ways, users can publish and modify service descriptions.

Find:The finding functionality makes services found by users after service publishing, with the help of find operator service requestors can obtain relevant information needed by bind service through sending a query specification interfaces based on the registered server.

Bind:In the binding operation, the service requester obtained the service binding information from the registration server based on the analysis, from which they know the detailed requirements for the service, like, service access path, the service call parameters,
return results, transport protocols, safety requirements, etc. Then, the service requester can achieve the service of remote calls by means of deploying their corresponding configuration based on these information systems.

In figure:2, explains the architecture of webservices in which web services are communicated with service client,those alredy published by service provider and all information contains by UDDI registry tells availability of web services.

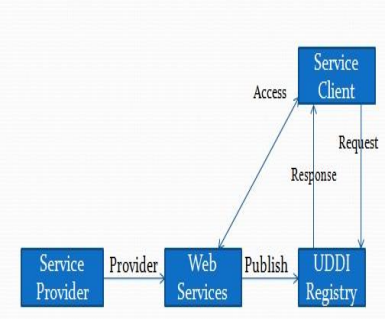

Figure 1: Web Services Architecture

II. FUNCTIONAL COMPARISION

A. Enterprises Java Beans(EJB)

EJB is one of way to achieve distributed computing for n-tier architecture, EJB is a class and it works as a mediator which take request from many source like mobile, console, html. EJB provide additional features transactional security, multithreaded, persistant.

Types of EJB

The EJB is categorized in two part ofSession Bean
Stateless and Stateful

Session Bean: Session bean is a purpose of apply to business logic functionality definition for web services.

Stateful Session Bean: A stateful bean is a business object having state that is keep track of calling client .They are dealing with throughout a session and that access the bean instance is strictly limited to only one client at a time. If concurrent access to a single bean is attempted any way the container serialize those object .The stateful session bean state may be persistent (passivate) automatically by the container to free up memory after the client has not access bean for some time.

For example checking out in a web store might be handle by stateful session bean that could use it state to keep track of where the customer is in the checkout process. Possibly holding logs in the item, the customer is purchasing from a system architecture point of view it could be less ideal to the client manage those logs.

Stateless Session Bean:

Stateless session bean is a business object that does not have a state associated with item however excess to the single bean instance if still limited to only one client at a time, concurrent access to the bean is prohibit. If concurrent access to a single bean is attempted the container simply routes, each request to different instance. this make stateless session bean automatically thread save. An instance variable can be used during a single method called from a client to a bean but the content of those instance variable are not guarantee to be preserve across different client method called instance of a stateless session bean typically pooled. If a second client access a specific bean right after a method called on it made by a first client has finished .It might get a same instance. The lack of poor head to maintain a conversion with a calling client make them less resource intensive than stateful bean.

The EJB consists of POJO for accessing business logic and web services request to data server. SOA[1] is now building loosely coupled applications with SOA, we able to focus on building web services. SOA is an architecture principle whereas a web services is a platform to build loosely coupled applications. Here we have built a implementation of web services with enterprises java beans **For example** The Big Bazaar built a web services and provide a WSDL describing how to call the service from the registry, generates the end point interface and proxy classes, and then invokes the web services Besides interoperability and application integration, the primary benefit that a web service provides is reusability of discrete business functionality. Figure 2 is shown in which web services composite with other component with the help of distributed technology (middleware)[9] and data server.

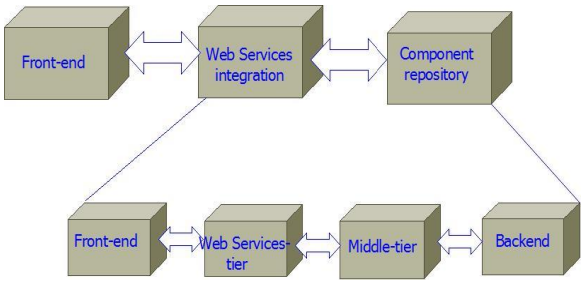

Figure 2: Web Services Communication

B. Hibernate

Hibernate is a java based middleware[12] designed to complete the Object Relational (O/R)mapping model, it is a traditional java objects and handle all the work in persisting those objects based on appropriate O/R mechanism and patterns. Hibernate will persist objects to the database tables of the underlying Database Server; in most cases the mapping exercise is straightforward. Hibernate configuration file contains data base connection specific information and hibernate mapping file contains class specific(Table Specific) for each persistent class a hibernate mapping is created.

Transaction [3]is an interface whose implementation is provided by framework and it is used when we want to work with transaction objects is created from the session.

Query is also a interface whose implementation is provided by framework and it is responsible to execute user based query. In figure Application user interact with configuration, it read all necessary xml files to stored database specific information of data server. After getting configuration file information, Application user request for session factory and session factory is granted by configuration. Session factory provides session to application user for transaction

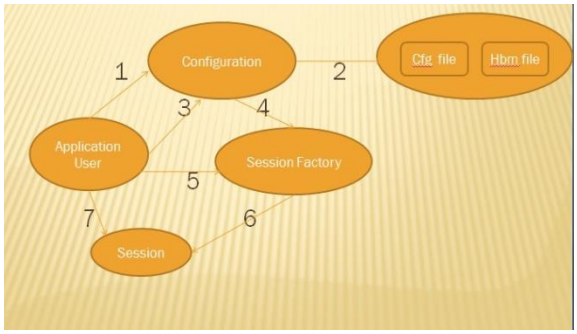

Figure3: Hibernate Configuration Steps

In figure 3, we have proposed a model to enhance the performance of web services by [8] using the technology hibernate which provides interoperability to access a web services. In this model client Want to communicate with web services with the help of SOAP[10], and it stored data in Data server with the help of distributed technology.

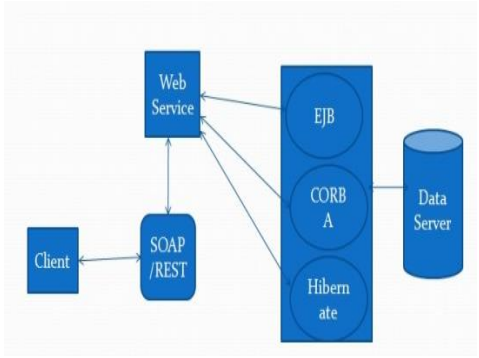

Figure 4: Enterprises Model

III. WEB SERVICES PERFORMANCE ANALYSIS TOOL

Testing Web Services Using J Meter

Web Services are tested for throughput, efficiency, and response simulating real-world conditions. A well designed load-testing strategy can simulate real-world load and performance scenarios with minimal hassle and cost. User loads and network conditions of varying nature can be effort less created and replicated. Testing can be undertaken till the output charts show a performance [13] range considered acceptable for an application of its nature. Load-testing results can hence be taken as a strong indicator of application performance in actual business environments.

Building a Web Service Test Plan

In the J meter web services[6] to create a Test Plan to test a Web Service. We have create 100 users that send requests to server for a resource. So, the total number of requests is (100 users) x (1 requests) x (repeat 2 times) = 100 HTTP requests. To construct the Test Plan, we will use the following elements Thread Group, Web Service(SOAP) Request, and Graph Results.

Here, we added or defined the thread group user for a request of resources from webservices, after definin g the thread users we set ramp up which delays the request accessing by second user, then number of times to execute the test plan (Loop) need to be set.

First control tells J Meter to execute the number of threads to execute the test.**Second control** tells J Meter how long to "ramp-up" to the full number of thread chosen.Here100 threads are used, and the ramp-up period is 100 seconds, then J Meter will take 1seconds to get all 1 00threads up and running. Each thread will start 100 (100/100) mili seconds after the previous thread was begun.**Third control** tells J meter how many times to execute the test plan.

Adding Web Services Requests: In figure5, thread user samplers are used to tell J Meter to send requests to a server. For [4]load/stress testing of Web Services we can use Web Service (Soap) Request sampler.

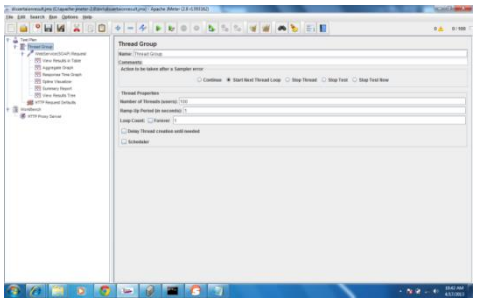

Figure 5: Number of user request for web services

By using the Listener to Store or view Test results: we are using a listener for storing all of the results of our thread user HTTP requests in a file and presenting a visual model of the data.

Response time: Response time is the most important parameter to reflect the quality of a Web Service.Response time is the aggregate time it takes after the client sends a request till it gets a response as a output.

Throughput: Throughput is measured in bytes and represents the amount of data that the Thread users receive from the server at any given second. We compared this graph to the response-time graph to see how the throughput affects transaction performance.

Load size: The number of concurrent Thread users trying to access the Web Service at any particular functionality in an interval of time.

CPU utilization: The amount of CPU time,instruction used by the Web Service while processing the request.

Memory utilization: The amount of memory used by the Web Service while processing the request.

Wait Time (Average Latency): The time it takes from when a request is sent until the first byte is received.

IV. WEB SERVICES PERFORMANCE ANALYSIS RESULT

We have taken 100 sample thread user for request(which is shown in figure) web service during ramp up time interval ,we found the throughput has been improved by using the interoperability concept and hibernate cache improve the throughput of web services response for user request.

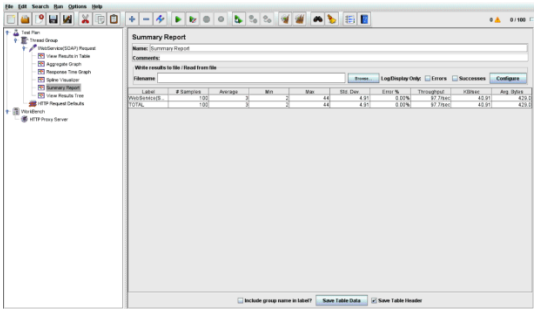

Figure:6 Performance result of web services

V. CONCLUSION

In this paper, we use a hibernate technology for access a web services instead of EJB, use of hibernate gives better interoperability with data server and web services. In future, we plan to improve the performance of our resource identification approach. We have validated our approach with the help of WSDL, which contains whole information of web services communication to server. There is lot of work need to be done for improving web services performance.

REFERENCES

[1] I. Naveed, "Web service composition patterns from execution logs "Department of Software Engineering International Islamic University, Jo DOI 10.1109/ICSEA 2011.

[2] P. James , "BPEL and SOA are changing web service development", IEEE Computer Society, pg 659–664, 2009 .

[3] A. Rabhi ," Transactions on service computing", University of Groningen, Nijenborgh, VOL. 4,NO. 2,2004.

[4] B. Fenglin,"Service oriented information system", School of Software Shanghai JiaoTong University Shanghai, China. 2012 .

[5] B. Youcef ,"An Approach for Web Services an SOA Development", Department of Compute Science Sultan Qaboos University Muscat, International Conference on Innovations Information Technology (IIT),Oman,2012 .

[6] P. La ,"Cohesive design of web application" University,Argentina IEEE, 2009.

[7] L.Philippe,Cristina,"A domain configurable development environment for Service-Oriented Applications",cedex, France ,IEEE, Vol 7,2008.

[8] S.Chang-ai,K.Elie,A.Marco,"Transaction management in service oriented system :requirements and a proposal" ,IEEE ,Vo l 4, No. 2.2011.

[9] H.Hassan," A design pattern for dependable web services using design diversity techniques", Automated Software Engineering Research Group, ECE Faculty, ShahidBeheshti University GC, Tehran, Iran, pp.58-78,2009.

[10] B.Upadhaya,"Migration of SOAP-based services to RESTful Services",Dept Electrical and Computer Engineering, 2011.

[11] L. Wen-Yau,"Apply rough set theory into the web services composition",Department of Information Management,IEEE,2008.

[12] K.Manjan,"Web Services and Java Middleware functional and performance analysis for SOA", Information Systems Laboratory, Trzaska 25,Ljubljana, Slovenia,IEEE,2007.

[13] Z.Zhaoli, Y.Zongkai,L.Qingtang,"Performance analysis of composite Web service", Engineering &Research Center for Information Technology, Vol. 1, No.1, pp.– 30, 2008.

[14] L.Konstantin, "Web application architecture for J2ee",IEEE, Vo 5, No. 3. 2005.