

Web-Based Service Optimization with JSON-RPC Platform in Java and PHP

WachyuHari Haji

*Faculty of Computer Science, Mercu Buana University
Jl. Meruya Selatan No. 1 Jakarta Barat, Indonesia
¹wahyuhari@gmail.com*

Abstract-Web based application has been used by many organizations to support their business activities within the organization. Sometimes, that application requires the availability of information from other parties outside the organization. Other than that, there is a need to exchange information between applications in a particular organization with applications in other organizations. The applications may be on the different systems and platforms. There is a technology that can be used, commonly known as a web service. JSON-RPC is an alternative to build a web service. This technology involves two applications that are connected via a data connection. During the connection, an application can execute a remote procedure from web service provider to do implementation web service using JSON-RPC technology, with PHP platform for client side application and java platform for server side application.

Keywords: Web service, JSON-RPC, PHP, java

I. INTRODUCTION

At present the need for web-based applications become a major requirement for many organizations to support all business activities in the organization. Sometimes an application requires the availability of information from other parties outside the organization, then the information is used as data for further processing. In addition, there is a need to exchange information between applications in a particular organization with applications in other organizations, which are likely to be on different systems with different platforms. That is why we need a technology that enables cross-platform communication via the web. One technology that can be used is a web-based service technology or commonly known as a web service. This technology allows for cross-platform communication. Web-based services are used as a service provider in the form of information to other applications, so that between different applications can interact and communicate through the functions provided by an application service provider. This technology facilitates multiple applications to interact with other applications within an organization and outside organization using standards that are not tied to any platform and are not tied to the issue of what programming language used by each application, so that applications can access a sub-routine in Other applications, such as the

application is currently accessing the sub-routine in the local system.

RPC is a simple protocol that can be used to build a web-based services. RPC allows an application to call to the sub-routine provided by other applications residing on different systems. RPC uses a specific data exchange formats like XML or JSON as a communication liaison between systems. RPC protocol with JSON format as a data exchange format called JSON-RPC. This research will implement a web-based services with JSON-RPC for PHP platform using the Java platform client application and server application service provider.

II. WEB SERVICE

Web-based service is a technology that aims to facilitate communication between applications, which use a standard platform that is not bound and not bound by the programming language used by each application. W3C definition (2011) "A software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL Web Services Description Language). Other systems interact with the service in a manner prescribed by its description using SOAP messages, conveyed using HTTP typically with an XML serialization in conjunction with other Web-related standards."

a. JSON

JSON is a data exchange format which has the characteristics of a lightweight, easily understood by humans, and have translated easily by computer. JSON is based on part of the JavaScript programming language. JSON format itself is a text that is independent of the programming language of any kind, this is because the style of language used is a style that is commonly used by programmers for a programming language family C includes C, C++, C#, Java, JavaScript, Perl, Python, etc. These properties make JSON is ideal for use as a data exchange format. JSON using the following form:
Object

A pair of name/value that is not sorted. The object of writing begins with the character "{" (open brace) and ends with the character "}" (curly braces). Each name is always followed by the character ":" (colon), and each name / value pair is separated by the character "," (comma).

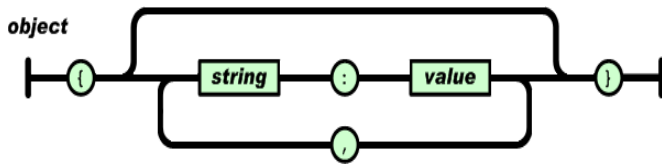


Figure 1 JSON Object

b. Array

Array is a collection of values sorted. Array always begins with the characters "[" (open square brackets) and ends with the characters "]" (close brackets). Each value is separated by the character "," (comma).

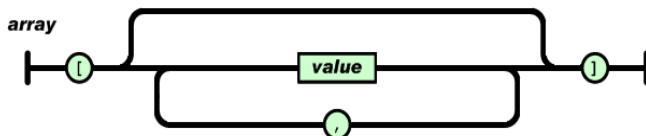


Figure 2 JSON Array

c. Remote Procedure Call

Remote Procedure Call (RPC) is a fairly simple protocol that is widely used today to build web-based services. According to Dave Marshall (1999) RPC defined "RPC is a powerful technique for constructing distributed, client-server based applications. It is based on extending the notion of conventional, or local procedure calling, so that the called procedure need not exist in the same address space as the calling procedure. The two processes may be on the same system, or they may be different on the systems with a network connecting them."

RPC allows us to access a sub-routine that is in another application while the application is located on the system and on different machines. To do this an application server must provide a set of sub-routines which can be accessed by client applications. The approach can be implemented as a server opening a socket, then the server will wait for a client that calls the sub-routine provided by the server.

RPC protocol uses a specific data format to exchange data, a data format that many use is to use XML technology, the XML protocol is usually named with XML-RPC. However, over the technology, found a new data exchange format is JSON, which has advantages over XML, JSON, among others, the nature of light, easy to read by human beings, easily written by humans, as well as small size, so that the resulting byte will be smaller than XML. It is certainly useful for later data processing, which will be

with the

faster than XML.

RPC protocol which uses JSON as a data exchange format called JSON-RPC

d. JSON-RPC

The basic concept of JSON-RPC itself is very simple, which is a common mechanism consisting of two systems are connected via a data connection. During the connection still exists, a system can perform to call a sub-routine found in other systems.

To make a long distance call to a sub-routine, then the object of an object inquiry or notification will be sent by the client to the server, then the server will give the child object to the processing of the request object, unless it is in the form sent by the client object notification, the server will not give the object a reply in response to the client.

A call to a sub-routine in the server by the client as well as when to call to a local sub-routine. When an RPC is done, the input parameter is mapped to a sub-routine in the service provider server, and the client to call the sub-routine is waiting for a reply object that is returned from the sub-routine in the call.

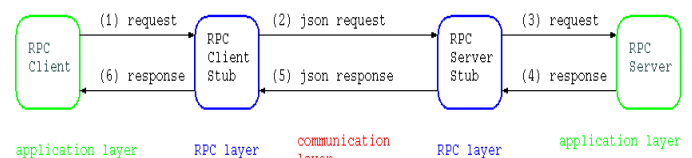


Figure 3 Flow processes in JSON-RPC

The picture describes a general mechanism that occurs in the JSON-RPC. In the picture there are two machines communicating with each other via a network, the client machine and server machine.

e. PHP

PHP (PHP Hypertext Preprocessor) is a web-based programming language that can process data dynamically. PHP will be fully executed by the server but is included in regular HTML pages. Applications are constructed by PHP in general will give results on the web browser, but the overall process is executed on the server. In principle, the server will work if there is a request from the client. In this case the client uses PHP code to send a request to the server. When using PHP as the scripting language of the application server, the server will do the following:

- Read request from client
Search the site/page on the server
- Perform the instructions given by the PHP to make modifications to the page.
- Send back the page to the client via the internet or intranet.

PHP has several advantages that can run on different platforms (Windows, Linux, Unix, etc.), but it's very easy to learn PHP.

JavaScript is a scripting language based on the concept of prototype-based programming. This language is particularly well known for its use in websites (as client-side JavaScript), and also used to provide access to object-sembbed script in another application.

f. JAVASCRIPT

III. GLOBAL WEB DESIGN BASED SERVICES

Making web-based services with JSON-RPC involves two applications that are connected via a data connection. During the connection still exists, an application can make calls and the execution of a sub-routine is contained in another application. To make call to a sub-routine in other systems, the object of an object inquiry or notification will be sent by the client application to the application on the server, then the server will provide the result of processing the object of retaliation against a client application based on the request object, unless the being sent by the client is the object of the notification, then the server will not send any reply to the client. Figure 4 below illustrates the flowchart for the process of communication that occurs between client and server on the system web-based services with JSON-RPC technology globally.

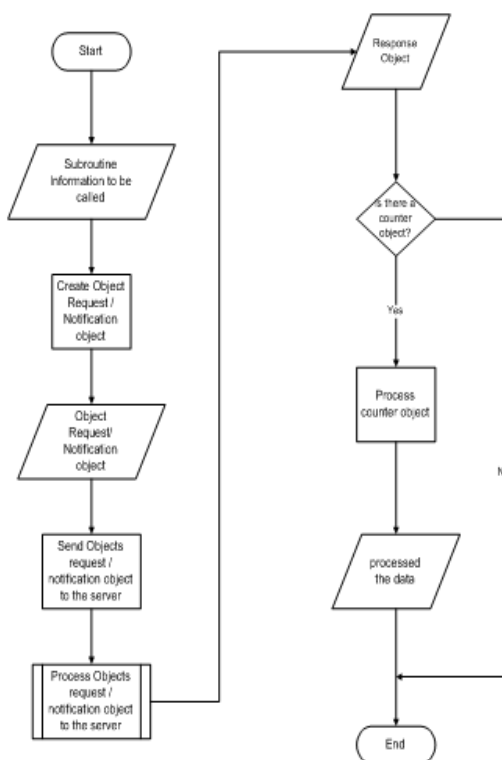


Figure 4 Flowchart Web-Based Service Systems With JSON-RPC

The implementation for this component is a Javascript file that will be placed in the client application. At this Javascript file there is a class that is JSONRpcClient class, which has a sub-b-routine specification as follow:

SUB-CLASS ROUTINE JSONRPCCLIENT / JSON-RPC JAVASCRIPT LAYER

No	Name of Sub-Routine	Uses
1.	JSONRpcClient (url, vuser, vpas s)	Sub-routine is useful for early initialization constructor and venue of the parameter is the URL where the server be
2.	JSONRpcClient .prototype.doLogin = function()	Sub-routine is useful for logging onto the application server web-based service providers
3	JSONRpcClient .prototype.getDataFilm = function(id, namaBioskop)	Sub-routine is useful for showing filmsthat are beingaired today at several theaters
4	JSONRpcClient .prototype.getDataBioskop = function(id)	Sub-routine is useful for displaying data existing cinema
5	function GetXmlHttpRequest()	Sub-routine is useful for objects Getting XmlHttpRequest, which will be used for access to web-based service provider's server via AJAX technology
6	JSONRpcClient .prototype.getHasilPolling = function(id)	Sub-routine is useful for displaying data poll
7	JSONRpcClient .prototype.setPolling = function(pilihan ,id)	Sub-routine is useful to set the selection of the most popular movie poll
8	function getRandomString()	Sub-routine is useful to get a random string to be used as an ID on the request object

a. JSON-RPC SERVLET

Implementation of JSON-RPC Servlet component is in the form of a class named ServletJSONRPC which is a subclass of the servlet class (javax.servlet.http.HttpServlet) in the java library. This class serves as a liaison between the client application to the application server. All sub-routines that exist in this class the same as its parent class (javax.servlet.http.HttpServlet), with a new sub-routine reimplementation of the sub-routine "doGet" and "doPost". Details can be seen in the following table:

TABLE 1

TABLE2.
SUB-CLASS ROUTINEServletJSONRPC(JSON-RPC Servlet)

No	Name of Sub-Routine	Uses
1.	doGet(HttpServletRequest Request req, HttpServletResponse resp)	Sub-routine will be called when an object request sent by the client through the delivery method "GET"
2	doPost(HttpServletRequest Request req, HttpServletResponse resp)	Sub-routine will be called when an object request sent by the client through the delivery method "POST"

b. OBJECT HANDLER

Implementation of Object handler component is a Java class named `twoRequestHandler` and `NotificationHandler`

TABLE 3.
SUB-ROUTINE
CLASS REQUEST HANDLER (OBJECT HANDLER FOR OBJECT REQUEST)

No	Name of Sub-Routine	Uses
1.	public String[] handledRequests()	Get names of all the sub-routine that handles all the request object
2	public JSONRPC2Response process(JSONRPC2Request req)	Object to process incoming requests

TABLE4.
SUB-ROUTINE
NOTIFICATION HANDLER CLASS (OBJECT HANDLER FOR OBJECT NOTIFICATION)

No	Method	Uses
1.	public String[] handledNotifications()	Get names of all the sub-routine that handles all notification objects
2	Public process(JSONRPC2Notification notification)	Processing notification of the entering Objects

IV. TEST RESULTS

The results of direct access to the application server through a browser can be seen in the following table:

The test results of the steps- steps described above are as follows:

- Test result showed that the communication goes as expected, and the data is displayed in accordance with data obtained from the server.
- Test result showed that the object is given by the server response was appropriate, and also supports the use of object return an error

The results of direct access to the application server through a browser can be seen in the following table:

TABLE5.
TEST RESULT THROUGH BROWSER APPLICATION SERVER

Object request	Object replies	Specification
{ "id": "hDCV1dNX", "method": "getDatabioskop", "params": [], "jsonrpc": "2.0" }	{ "id": "hDCV1dNX", "result": ["Citos", "Grand Indonesia", "Pejaten Village"], "jsonrpc": "2.0" }	
{ "method": "poolingFilm", "params": ["indonesia"], "jsonrpc": "2.0" }	-	There was no reply object because the object of the notification
{ "id": "hDCV1dNX", "method": "getArtis", "params": [], "jsonrpc": "2.0" }	{ "id": "hDCV1dNX", "error": { "message": "Method not found", "code": -32601 }, "jsonrpc": "2.0" }	Because the sub-routine is not registered

V. CONCLUSIONS

Based on implementation of case studies have been conducted, it can be concluded as follows:

JSON-RPC can be used as one alternative solution to create cross-platform web-based service, which allows communication between two applications with different platforms, the process of communication between client and server applications with JSON-RPC is faster and lighter than the other web-based service technology that uses XML data exchange format, this is because it uses the JSON data interchange format used to make the transmitted data size is smaller.

Application development process becomes faster and more efficiently because of the JSON-RPC technology weapons, the development of an application can be done separately for the application server and client applications.

REFERENCES

- [1] JSON-RPC.ORG (2005). JSON-RPC Specification. From <http://json-rpc.org/>, accessed on 2 December 2011
- [2] Marshall, Dave(2000). Remote Procedure Calls (RPC). From <http://www.cs.cf.ac.uk/Dave/C/node33.html>, accessed on 5 December 2011
- [3] DigitalMutual AidSociety(MDGR), Introduction toComputer OperatingSystems, 2008.
- [4] Wikipedia the Free Encyclopedia (2011), Web Service. From http://en.wikipedia.org/wiki/web_service , accessed on 2 December 2011
- [5] w3schools.com(2011). W3Cweb servicetutorial. From<http://www.w3schools.com/webservices/default.asp>,accessedon2 December2011
- [6] AdbulKadir, BasicDynamicWebPerogramanUsingPHP, AndiYogyakarta, Yogyakarta, 2002
- [7] http://www.itelkom.ac.id/library/index.php?view=article&catid=15%3A%20pembelajaran&id=304%3Ajson&option=com_content&Itemid=15 (accessed on 24 June 2009)