Marima Andrew Mambondiumwe

Professor Matt Jadud

CSC 412 Networking

10 February 2017

Implementing Remote Procedure Calls

The article describes a RPC project that the authors designed, the purpose of the project was to make distributed computation easy, i.e. to remove unnecessary difficulties and to provide secure communication within RPC. Work for the RPC’s began as early as in 1976. For this current project that the authors wrote about, an RPC facility was constructed for the Cedar project. Cedar is a large project that is concerned with the development of a programming environment that is both powerful and convenient for the building of experimental programs and systems (Birrell 40). XML-RPC is a Remote Procedure Calling protocol that works over the Internet and uses HTTP as the transport and XML as the encoding. It allows software to run on disparate operating systems. An XML-RPC message is an HTTP-POST request. The body of the request is in XML. A procedure executes on the server and the value it returns is also formatted in XML. XML-RPC is designed to be as simple as possible, while allowing complex data structures to be transmitted, processed and returned” Whereas SOAP (Simple Object Access Protocol), is a protocol for exchanging structured information based on XML for the messages format and on protocols such as RPC and HTTP for messaging negotiations and transfer, it is used as communication protocol for making web service (SOAP model) requests and to obtain answers. SOAP consists of three parts, which are, an envelope that defines a framework for describing what is in a message and how to process it, a set of encoding rules for expressing instances of application-defined datatypes, and a convention for representing remote procedure calls and responses. A RESTful web-service is a web-service implemented using HTTP and respecting the principles of REST. So, REST is a set of principles or rules for telling how standards like HTTP and URI can be used (Ploscar).

Per a short description on GitHub about the zerorpc open source code, zerorpc is a light-weight, reliable and language-agnostic library for distributed communication between server-side processes. It provides support for streamed responses it is similar to python generators and makes zerorpc more than a typical RPC engine. Built-in heartbeats and timeouts detect and recover from failed requests. Introspective capabilities, first-class exceptions and the command-line utility make debugging easy.

An example of a client -server relationship that can be implemented using SOAP is the Berea College Network printing. Or testing can be done using a simple “hello world” to see whether the message will be able to be relayed across the network. Basically, client-server implementation can be any of the following relationships, synchronous server - synchronous client, asynchronous server - synchronous client, synchronous server - asynchronous client, asynchronous server - asynchronous client. There are two ways to synchronize the two ends of the communication. The synchronous signaling methods use two different signals. A pulse on one signal indicates when another bit of information is ready on the other signal. The asynchronous signaling methods use only one signal. Synchronous server and client design is easier to implement and maintain. However, it has problems with scaling. Contrary, asynchronous server and client are harder to implement and maintain. For the server side, it is easier to expose a set of methods using soap than it is to expose them using regular HTTP, the opposite is true for the client side.

There are several differences that are mentioned in several articles as well as from others sources that I found whilst doing my research, SOAP has more complicated designs compared to XML-RPC, another difference between SOAP and XML-RPC is that Soap is not well supported in python whereas XML-RPC has great support in python through the standard library. SOAP supports document-level transfer, whereas XML-RPC is more about values transfer, although it can transfer structures such as structs and lists. For XML-RCP, the order is more relevant than the procedure. For SOAP, it is the other way around. SOAP is more powerful in terms of capabilities compared to RPC. RPC, on the other hand, has more python support than that of SOAP. An advantage of REST over SOAP is that REST web services can send plain text, JSON, and XML whilst SOAP is just restricted to XML format. REST was released after Soap to fix the problems with SOAP, however, REST has problems of its own. Both techniques have issues to consider when deciding which protocol to use (Ploscar).

It was interesting to note that all of Yahoo’s web services use REST whilst companies like eBay and Amazon use both REST and SOAP. Google use SOAP. Yahoo choosing to use REST web services because they have some of the following examples. They are light weight and easy to build.

An important thing to note about RPC is the fact that during the design process, the Remote Procedure calls should be as similar as possible to the semantics of the single machine or local procedure calls. This is necessary to ensure that the RPC is easy to use particularly for the programmers who are only used to the single machine network programming. Nelson’s thesis is mentioned several times in the reading and it basically is, a doctoral dissertation on the examination of the design possibilities for an RPC system. It includes extensive analysis of several RPC protocols and implementations, and included an examination of the contributing factors to the differing performance characteristics (Birrell 56).

The project that is described in the reading has been fully implemented and was in use by Cedar programmers, However, it was informative to compare how the early RPC procedures were done compared to the modern way of doing it that involves the use of SOAPS, XML-RPC and RESTful. This article enabled me to be able to research and know both the similarities as well as the differences between these various forms of RPCs.

Works Cited

Ploscar, Adina. "*XML-RPC vs. SOAP vs. REST web services in Java – uniform using WSWrapper*." INTERNATIONAL JOURNAL OF COMPUTERS 2012th ser. 6.4 (2012): 215-23. Web. 09 Feb. 2017. <http://www.naun.org/main/NAUN/computers/16-579.pdf>.

0rpc. "*0rpc/zerorpc-python*." GitHub. Open source code, 31 Jan. 2017. Web. 10 Feb. 2017.

Birrell, Andrew D., and Bruce Jay Nelson. "*Implementing remote procedure calls*." ACM Transactions on Computer Systems 2.1 (1984): 39-59. Web. 09 Feb. 2017.