Distributed database communication based on UDDI standard

Martin Tsenov

Abstract: *This paper explained decision for network and web services architecture between distributed network systems. The proposed method is based on Web services standards - SOAP and UDDI. Software solution for development of an application, which provides data access and exchange to various distributed networks is presented and explained.*

Key words: *database, protocols, networks, web services*

INTRODUCTION

The use of data exchange on the World Wide Web is expanding rapidly as the need for application-to-application communication and interoperability grows. These services provide a standard means of network communication between different software applications involved in presenting dynamic context-driven information to the user. In order to promote interoperability and extensibility among these applications, as well as to allow them to be combined in order to perform more complex operations, data exchange architecture is needed. In this paper the author describes a set of requirements for data exchange architecture for Web services. The proposed method for data exchange between distributed databases is based on the SOAP protocol [1] and UDDI standard [2]. The main aim is to provide data access and exchange to various clients via SOAP protocol and UDDI (Universal Description, Discovery and Integration) registry. For software development is used open source server-side script language PHP. The example revolves around the open source standards, which allows for clients to access and exchange data between distributed databases. It also allows various other system users to interact with the application. The below is the sequence of activities of the example:

- Define the method for data exchange.
- Implementation of the proposed method.
- Develop a Software implementation.

PHYSICAL IMPLEMENTATION

A. Theoretical part

- UDDI: UDDI (Universal Description, Discovery and Integration) [2] - is the name of a group of web-based registries that expose information about services or other entity and its technical interfaces (or API's). These registries are run by multiple Operator Sites, and can be used by anyone who wants to make information available about one or more services or entities, as well as anyone that wants to find that information. There is no charge for using the basic services of these operator sites. By accessing any of the public UDDI Operator Sites, anyone can search for information about web services that are made available. The benefit of having access to this information is to provide a mechanism that allows others to discover what technical programming interfaces are provided for interacting with a business for such purposes as electronic commerce, etc.

UDDI specifications consist of an XML schema for SOAP messages, and a description of the UDDI API specification. Together, these form a base information model and interaction framework that provides the ability to publish information about a broad array of Web services. The core information model used by the UDDI registries is defined in an XML schema. XML [3] was chosen because it offers a platform-neutral view of data and allows hierarchical relationships to be described in a natural way. The emerging XML schema standard was chosen because of its support for rich data types as well as its ability to easily describe and validate information based on information models represented in schemas.

The UDDI XML schema defines four core types of information that provide the kinds of information that a technical person would need to know in order to use a partners Web services. These are: business information; service information, binding information; and information about specifications for services

- SOAP: SOAP (Simple Object Access Protocol) is a simple, lightweight protocol for structured and strong-type information exchange in a decentralized, distributed environment. The protocol is based on XML (eXtensible Markup Language) and consists of three parts:

1. An envelope which describes the contents of the message and how to use it
2. A set of rules for serializing data exchanged between applications
3. A procedure to represent remote procedure calls, that is, the way in which queries and the resulting responses to the procedure are represented.

Similar to object distribution models (IIOP, DCOM...), SOAP can call methods, services, components and objects on remote servers. However, unlike these protocols, which use binary formats for the calls, SOAP uses text format (Unicode), with the help of XML to structure the nature of the exchanges.

SOAP [4] can generally operate with numerous protocols (FTP, SMTP, POP...), but it is particularly well suited to the HTTP protocol. It defines a reduced set of parameters which are specified in the HTTP header, making it easier to pass through proxies and firewalls.

B. Structure of the data exchange architecture.

Using XML and SOAP, the integration and interoperability problem has been simplified in layers. XML provides a cross-platform approach to data encoding and formatting. SOAP, which is built on XML, defines a simple way to package information for exchange across system boundaries. SOAP bindings for HTTP are built on this packaging protocol and define a way to make remote procedure calls between systems in a manner that is independent of the programming language or operating system choices made by individual companies. Prior approaches involved complex distributed object standards [5] or technology bridging software. Neither of these approaches has proven to be cost effective in the long run. Using XML and SOAP, this cross-language, cross-platform approach simplifies the problem of making systems at two companies compatible with each other.

Table1. Data exchange architecture layers

Universal Description, Discovery Integration (UDDI)
Simple Object Access Protocol (SOAP)
Extensible Markup Language (XML)
Common Internet Protocols (HTTP, TCP/IP)

C. Implementation of data exchange architecture.

Software implementation of the network and web services resource architecture is based on an open source technologies PHP scripts language [6], Apache Web server, MySql databases [7]. For communication based on proposed method are developed a software module with PHP called soapserver.php. It is used for SOAP request and SOAP response, which are based on the proposed architecture. The other part of the software implementation (uddidata.php) is used to prepare the data from databases in XML format. It is needed because of the SOAP standard and proposed method specification. The software implementation defined the following steps:

- The calling application makes a procedure call on the XML client indicating the URI of the server, the procedure to be called on the UDDI server, and the parameters to be sent to that procedure.
- The XML client takes the method and parameters and builds an XML container for them; the XML container is the sent over HTTP as a SOAP request.
- XML server that receives the SOAP requests parses that XML container and determines the method to be called and the parameters to this method.
- The method is executed on the server and returns a result.
- The result is packaged as XML and the server returns the XML result container as the response of the POST request.
- The client parses the XML response container and returns the result to the calling application.
- The application processes the result.

Figure 1 presented structure of software implementation based on the data exchange architecture.

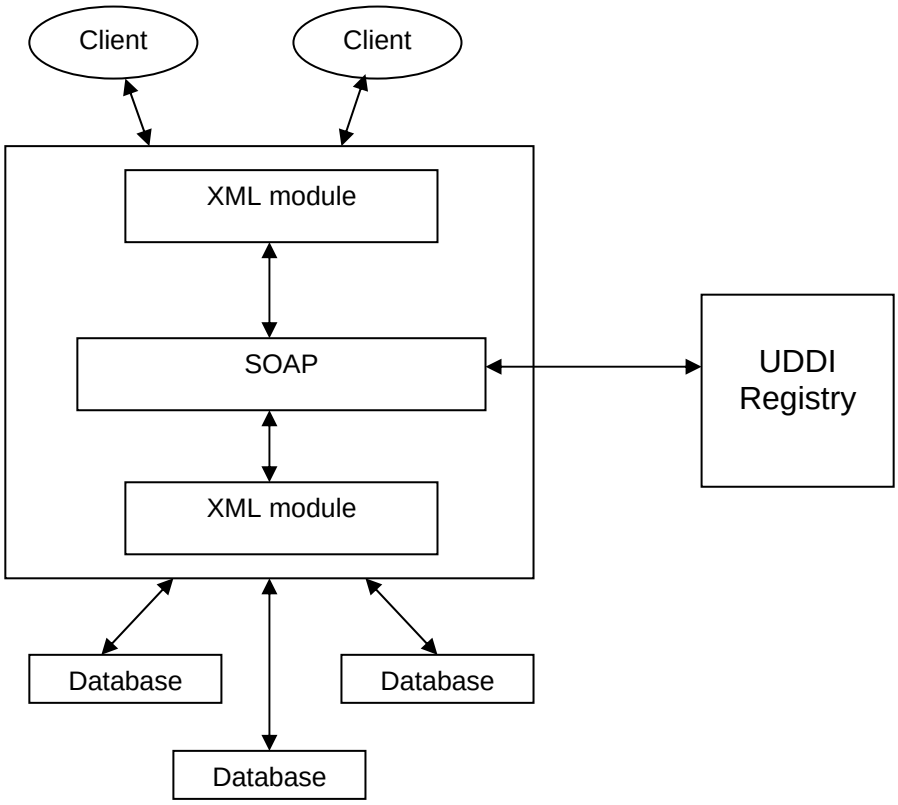

Fig1. Structure of software implementation.

Sample source code of the software implementation based on the data exchange architecture is presented above.

```
//XML-RPC declaration:
$payload[]="HTTP/1.0 200 OK\r\n";
$payload[]="Status: 200\r\n";
$payload[]="Server: SOAPx4 Server v0.5\r\n";
$payload[]="Connection: Close\r\n";
$payload[]="Content-Type: \ text/xml;charset=UTF-8\r\n";
```

```
$payload[]="Content-Length: ".strlen($xml_query)."\r\n\r\n";
//SOAP request
function send ($soap_data,$path,$server)
{   global $outgoing_payload;
    $incoming_payload='';
    $action='urn:soapBI'; $port='80';
$fp=fsockopen($server,$port,$errno,$errstr,3);
$outgoing_payload =  "POST ".$path."HTTP/1.0\r\n".
                          "User-Agent: SOAPx4 v0.5\r\n".
                          "Host: ".$server."\r\n".
                           "Content-Type:text/xml\r\nContent-Length:
                           ".strlen($soap_data)."\r\n"."SOAPAction:
                           \"$action\""."\r\n\r\n". $soap_data;
    // send
   // SOAP response
       while($data = fread($fp, 32768))
       { $incoming_payload .= $data; }
       fclose($fp); $incoming_payload = $incoming_payload;
       return $incoming_payload;
```

The software implementation can be reach at http://hs19.iccs.bas.bg.

CONCLUSIONS

This paper explained decision for network and web services architecture between distributed network systems. The proposed method is based on Web services standards – SOAP and UDDI. Software solution for development of an application is presented and explained. The main aim of the proposed model is to provide data access to various clients via Web Services. The example revolves around the open source standards, which allows for clients to access and exchange data between distributed databases. It also allows various other system users to interact with the application.

REFERENCES

[1] http://www.w3.org/TR/soap/

[2] http://www.uddi.org/

[3] Stoilov T., Stoilova K., XML-EDIFACT Translation in EDIPA System. Proceeding volume from the 9-th IFAC Symposium on Transportation Systems, Edited by E.Schneider and U.Becker. Braunschweig, Germany, Pergamon Press, 2001, p.203-208.

[4] Ivanova E., SOAP Based Multiple Search,International conference CompSysTech 2003, p.III.26-1 III.26-6

[5] Trichkov Kr. Distributed search in heterogeneous databases (repositories) system in global network using Z39.50 standard. Proceedings of the International Conference of Computer Systems and Technologies (e-learning), CompSysTech, 2003, Sofia, Bulgaria, 19-20 June, p.IIIA.21-1 - IIIA-21-6

[6] C. Scollo, J. Castagnetto, D. Veliath, H. Rawat, S. Schumann, Professional PHP Programming, Wrox, December 1999

[7] Paul Dubois, "MySql", New Riders, 2000

ABOUT THE AUTHOR

Assist. Prof. Martin Tsenov, Department Hierarchical Systems, Institute of Computer and Communication Systems - Bulgarian Academy of Sciences, Sofia, Phone +359 2 979 2774, E-mail: mcenov@hsi.iccs.bas.bg