Performance of Web Service and Messaging in Service Oriented Architecture

Rosni Lumbantoruan, Rizal Simamora, Dewi Simamora, and Budi Marpaung

***Abstract*—**The advanced of business has urged interoperability in order to able exchanging data between applications. Service Oriented Architecture (SOA) is one of the solutions to overcome this interoperability. SOA as a concept that enabling the integrations among applications can be applied using web service or messaging technology. Instead of messaging, web service is the most common used and also the oldest. This research compared the performance of web services and messaging from conceptual to implementation view and suggested the proper technology to apply for certain functions of the given case studies. Web services and messaging were applied to the common functions of three existing applications namely Academic Information Systems (AIS), New Students Enrollment (NSE) Application, and Online Library Information System (OLIS). Web services query response time is faster than in messaging when used to do reporting with select query while in performing command update/insert/delete, messaging performs better than web services.

***Keywords*—**interoperability, messaging, SOA, web services

I. Introduction

Interoperability is the ability of an application to communicate with others applications regardless of its platform (hardware, operating system, and language development). Service Oriented Architecture (SOA) exists to ensure interoperability and modularity in applications. Modularity of an application means introducing module for particular functions in order to ease development phase. SOA is the right choice to exchange data among collaborated applications [1].

Basically, SOA can be implemented by various different technologies such as web services, messaging, RESTFull, and socket. Among of these technologies, web services is the most commonly used and claimed have a good performance in exchanging data between collaborated applications [2], RESTFul is properly used for a stateless application and for devices such as PDAs and mobile phones [3] while socket is used for application with simple logic business and client/server communication [4].

Rosni Lumbantoruan is a lecturer of Del Institute of Technology in Information System Department, Indonesia (phone: +6281361598179; e-mail: rosni@del.ac.id).

Rizal Simamora, was the alumni of Del Polytechnic of Informatics (e-mail: if08023@students.del.ac.id).

Dewi Simamora, was the alumni of Del Polytechnic of Informatics (e-mail: if08070@students.del.ac.id).

Budi Marpaung, was the alumni of Del Polytechnic of Informatics (e-mail: if08025@students.del.ac.id).

Compared to web services, messaging is a quite new technology in SOA development. Both web services and messaging have its advantages and disadvantages. Web services is more loose coupled compared to messaging technology however the service client should firstly invoke service provided by the service provider. Messaging has fiture that enabling an application to broadcast a message to any application integrated to it [5].

In this research the performance of technologies, web services and messaging were compared in SOA implementation, both theoretically and by applying it to the case study. Features of both were compared literally while for implementation, an application was built with the common functions of three running application, they are: Academic Information Systems (AIS), New Students Enrollment (NSE) Application, and Online Library Information System (OLIS). The performance of the application was measured by its response time to compare to see the performance both web services and messaging for each given cases.

II. Research Method

The research was started by doing literature study of SOA, web services and messaging. Performances of both technologies were compared by firstly comparing the features of both literally and then finding case study that share same functions to be used as a means to integrate the existing applications with web services and messaging technology. At the end, the response time of the application that applied web services was compared to the response time given by the application that applied messaging.

In the sub chapter below are the results of the literature study consulted during the research.

A. Service Oriented Architecture

SOA (Service Oriented Architecture) is an approach in building a system or software by transforming the functionality or business process of the system to services or generally called as service oriented [6]. Communication among services is done by sending message as shown in figure 1.

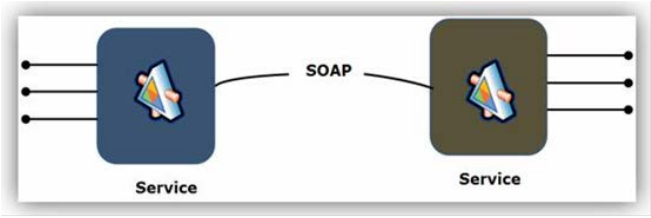

Fig. 1 Communication between services

Below are SOA concept such as services, interface, messages, synchronicity, loose coupling, registries, and quality of services [8].

1. Services

A service is representation of functionality or business process of a system. A service can be used to communicate among services with a standardized protocol. A service is not dependent to others or independent. Through a service, a system can send its value to other integrated system.

2. Interface

Interface is the intermediate between service consumers to service provider. By this it means, an access to service provider by service consumer can be done through interface.

3. Messages

A service communicates to other services by exchanging messages, in the other words operations in service can be defined as a collection of messages.

4. Synchronicity

There are two way of communication in SOA, synchronous and asynchronous communication. In synchronous, a service returns responds to service consumer after the process is finished. While in asynchronous, a service operation does not return respond to service consumer, in returns it sends acknowledgment that let the consumer notices that the operation was success.

5. Loose Coupling

Even though SOA integrates a system to others, however dependencies among systems is minimal or usually called as loose coupled. This kind of dependencies ensures that it will also give a minimal impact to other systems when a system has to change. Loose coupling increases robustness of a system.

6. Registries

A Service registry is a list of available directory services. Each service is registered into a registry to simplify searching operations. Service providers make the services in registries public. Service consumers search for service that they want to access at the registries. Searching was done by name, service functions, or business process property. Universal Description Discovery and Integration (UDDI) are the example of registries services.

7. Quality of Services

Service is closely related to quality of services attribute such as security, messages, transaction, correlation, management, policy, and other requirements. In web services, quality of services is covered in WS-Security, WS-Addressing, WS-Coordination, et cetera.

B. Web Services

Web service is one of technology that implement SOA concept. Web service is intended to increase collaboration between programs and companies by enabling a function in web services is used by any applications without have to go into detail of the program and the progress in it. Web services technology is available in platform like J2EE and.NET platform so there is no need for software developers to add another package. Based on W3C, "A Web Service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-process able format (specifically WSDL). Other systems interact with the Web Service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards."

XML, WSDL, SOAP and UDDI are components that are needed in order to implement web services [7]. Web services technology is suitable to use when the type of communication used in implementing web service is point to point communication model. In principal, web services have its own standard that able to do one to many communication. However, this technology is rarely used due to its complexity and not as simple as messaging technology. When invoking

Client should invoke a service that is going to be accessed to a service provider. In this case, client should play active. Web services technology is suitable for reporting or when the service is called on demand.

C. Messaging

Messaging is one of other technologies that is used in SOA implementation. With SOA, components or services can interact each other louse coupled, so a service can be invoked by other programs or services regardless its position and platform. In order to do so messaging framework should be standardized so all services use the same format and protocol to support data exchange [9]. Data format for transferring is binary data.

III. WEB SERVICES VERSUS MESSAGING

In this sub chapter discussed the comparison of web services to messaging.

A. Web Services

Web services are able to change business model B2C (Business to Consumer) to B2B (Business to Business), that is from communication of user (human) to web to communication of application to application.

Web services offer an ease to the user who will access the service. Service of other users can be accessed by other users without knowing the service's location and even though the service is moved. Furthermore, web services have features to able a developer to use different programming language to access the service. Web services use HTTP/HTTPS protocol running in web browser. However, web services have not supported an application component based development yet user can access the existing service by implementing the service. It enables the usage of web component services easier, in which client can customize the service based on requirements.

Based on the analysis, HTTP or HTTPS has some drawbacks; firstly it cannot be used in a long session because each request needs server to set up a new connection, as a consequent, it needs more time. Secondly, for these are stateless protocol, then each time data sent, the connection is off, and then the server needs other ways to send notification regarding client's status [9]. For the case study, Academic Information System (AIS) needs to display list of new books from Online Library Information System (OLIS), for this operation there is a need to read and report data, and web services is more suitable if used in this case. Web services supports query process very well and the drawback of web services is not a problem anymore since each time the data read, data should be directly sent to the invoker or service consumer.

B. Messaging

Messaging with distributed communication with a loosely coupled enables a software component or an application stand independently from other software components/applications. Messaging is properly used when a developer needs a minimal dependency between software component providers to information from the other software components. It will ease modification of software components or when an application is running with or without client applications at the same time.

Messaging is more efficient since data exchanging is done asynchronously, in which an application can do data transfer even though both applications are not running. The message will be saved in queue and directly sent as soon as the receiver is on. In messaging, there is pattern publish in which a sender or message provider can broadcast a message to message consumers at one time. It causes the implementation of messaging technology is more efficient and having better performance since each request from message consumer does not need a new connection from message provider since it is published in queue.

Each time a service messaging finished a request, the message will be published by event message to the listener, it makes messaging technology is better used in a case with write operation/command. Client in messaging needs only to listen each change in service provider that will be sent through a queue.

C. Comparative Parameters of Messaging and Web Services.

Web services will be compared to messaging theoretically by comparing the features of both technologies. In addition to that the response time after implementing web services and messaging to the case studies are measured and compared. Theoretical comparison will be viewed from the aspects of security while for implementation the response time of both implementations will be measured since a client invokes a request to a service till gets response from the provider. For example, data transfer of student's data from NSE application to AIS. Get Student Data was implemented using web services and messaging in NSE. AIS will play a role as a client that will invoke service getStudentData from NSE. Time spending for both web services and messaging to process the request-response until insert the data response to the database will be measured and compared.

IV. CASE STUDY ANALYSIS AND IMPLEMENTATION

Academic Information System (AIS), New Student Enrollment (NSE) Application, and Online Library Information System (OLIS) will be used as the case studies to implement web services and messaging technology. AIS is a system to process student's data, lecturer's data, and learning processes. NSE Application is an application that is used to process new student enrollment data, while OLIS is an information system used to process book's data and borrowing.

Those three applications were integrated using SOA with messaging and web services technology so each application can exchange data among them. A system can play a role as service provider (server) and service consumer (client). Below are the descriptions of case studies.

A. Functions from the Case Studies

Here are the functions from the three applications that will be used and transformed into services. In summary, AIS has function update status of student's data that will be applied using web services and messaging. Messaging was applied using JMS that will exchange data asynchronously. Services provided by AIS will be accessed by OLIS. On the other hand, services provided by OLIS will be accessed by AIS in order to announce the new items (books and CDs) in library.

NSE application has publishing announcement function and student's data transfer. This service later on will be accessed by AIS. After AIS and NSE were integrated, AIS will accessed NSE application periodically (auto-refresh) to check the new announcement in NSE that should be also published in AIS. This case is also the same function transferring student's data.

OLIS has function to publish new items in library such as books and CDs. AIS will check OLIS periodically to check the new items in library.

1) Update Student's Status from AIS to OLIS

Student's data in AIS and OLIS should be synchronized. It means that new students in AIS should be registered directly in OLIS, so they can borrow books from the library. Furthermore, any changing status of students such as from active to non-active/drop-out should be synchronized as well in both applications.

2) Publishing New Books and CDs in OLIS

Whenever the library has new books or CDs, the list of the new items should be informed through the AIS. Since AIS is the most frequently accessed than OLIS so the announcement of these new items should be directly published in AIS.

3) Publishing Announcement from NSE to AIS

When opening new enrollment students, such as entrance requirements, the result of the entrance test, registration of the new students should be announced in AIS. AIS uses services

that is provided by NSE application to published the announcement from NSE.

4) Student's Data Transfer from NSE to AIS

New student's data that are inaugurated should be transferred from NSE to AIS when the students get registered. It will ease the academic administrator to process the data.

In this research, all these functions will be changed as services. So the existing applications will different in structure with the application built during this research.

B. Implementation of Web Services and Messaging

1) Development Environment

Software and development tools in building AIS, OLIS, and NSE with functions as stated above was listed in Table I and II below.

TABLE I
SOFTWARE AND DEVELOPMENT TOOLS FOR J2EE PLATFORM

No	ITEM	DESCRIPTION
1	Platfom	Java Enterprise Edition 5.0
2	Application Server	GlassfishESBv2
3	IDE	Netbeans 6.5.1
4	Technology	JAX-WS dan JMS
5	Database	MySQL Server 5.0

While software and development tools for .NET platform are listed in Table II below.

TABLE II
SOFTWARE AND DEVELOPMENT TOOLS FOR .NET PLATFORM

No	ITEM	DESCRIPTION
1	Platforfm	ASP.NET
2	Application Server	visual studio 2010
3	IDE	visual studio 2010
4	Teknologi	Web services
5	Database	SQL Server 2005

2) Development of AIS

AIS was developed with web services and messaging technology. Web services were built in .NET platform. This application consists of many components such as controller, model, view, and web services. Controller is the intermediate between user interfaces to model. Model is a data object passing by controller and view, while view is a user interface. Web services have function to implement service that will be invoked by OLIS.

3) Development of NSE Application

NSE application was developed with web services and messaging technology. Both of the applications was built in J2EE platform. JSP play an important role as web interface, servlet as controller of user interaction with web interface and data processing. JPA was used as ORM to store data to database. Web services are used to implement service that will be invoked by AIS.

4) Development of OLIS

OLIS was developed with web services and messaging technology. Same with NSE application, both OLIS applications were built in J2EE platform. JSP play an important role as web interface, servlet as controller of user interaction with web interface and data processing. JPA was used as ORM to implement data in database while web services are used to implement service that will be invoked by AIS.

V. RESULT AND DISCUSSION

A. Theoretical Comparison

1. Availability of technology platform\

Both web services and messaging are available in J2EE and .Net platform.

2. Security

Security in web services is better than security in messaging. Web services have ws-security that applied SSL in transport layer, end to end security in application layer and firewall in network layer, while messaging does not yet have any standard for security.

3. Flexibility

Messaging technology is more flexible than web services. In web services, any changes to an application need to firstly terminate, changing the code in controller, and then re-deploy, while in messaging developer does not need to change the code but only un-deploy the service.

4. Implementation

Web services are properly used for reporting while messaging for querying data (transaction).

5. Service Invoke

Service consumer has to invoke service in web services while in messaging client needs only to listen for any changes by service provider.

6. Data Exchange

WSDL is used as protocol for data exchange in web services while in messaging used channel. So for private message, it is better to use WSDL for security reason.

7. Guaranteed Delivery Data

Web services do not guarantee data delivery, so data will lose whenever the server is off during the process. In messaging, data will be saved in queue that will guarantee data delivery.

8. Reliability Characteristics

Web service supports one-to-many communication even though in implementation it is still complex and not as simple in messaging that supports publish/subscriber with topic and point to point with queue.

9. Speed

In one-to-one communication, web services is faster in exchanging data than in messaging.

10. Interface Specification

Web services have WSDL (readable) as interface while messaging does not have any.

11. Data Format

Web services exchanging data using XML data format while messaging in binary format.

12. Ease of Use

Web service is easier in usage than messaging using JMS because JMS is complex if applied in multiple systems.

13. Firewall friendly

Firewall is used as security in web services but not in messaging. For a transaction that involves private data, it is better to use web services for security reasons.

B. Response Time Comparison

In measuring response time of the built applications; AIS, OLIS, and NSE application some testing conditions that are performed are that the database are empty and there is only one client to request the service since in this research used point-to-point communication. Testing was done for the three functions stated they are: Publishing New Books and CDs in OLIS, Publishing Announcement from NSE to AIS, and Student's Data Transfer from NSE to AIS.

Response time was measured in LAN connection with 100.0 mbps. The response time of both web services and messaging was measured in thread and using stopwatch. To ensure that the testing population was accurate then each case study was test using 100, 500, 1000, 3000, and 5000 records in which each test was repeated 5 times and the average of the time was calculated.

Below are the result of the testing for each application that implement web services and messaging technology. Table III shows the response time of publishing new books and CDs for both technologies.

TABLE III
RESPONSE TIME OF PUBLISHING NEW BOOKS AND CDS

RECORD	RESPONSE TIME IN MILLISECONDS	
	WEB SERVICES	WEB SERVICES
100	352	6.600
500	2.944	26.000
1000	3.350	49.400
3000	13.268	150.400
5000	28.143	264.000

The table concisely shows that web services perform better in publishing new books and CDs.

Table IV shows the response time of publishing announcement from NSE to AIS.

TABLE IV
RESPONSE TIME OF PUBLISHING ANNOUNCEMENT FROM NSE TO AIS

RECORD	RESPONSE TIME IN MILLISECONDS	
	WEB SERVICES	MESSAGING
100	421	6.200
500	1.643	26.200
1000	3.348	49.000
3000	13.346	152.600
5000	26.776	273.000

It can be clearly seen than for this function, web services performs better than messaging technology.

Table V below shows the response time of transferring student's data from NSE to AIS.

TABLE V
RESPONSE TIME OF PUBLISHING NEW BOOKS AND CDS

RECORD	RESPONSE TIME IN MILLISECONDS	
	WEB SERVICES	WEB SERVICES
100	343	5.800
500	1.466	25.000
1000	3.335	51.000
3000	12.613	152.000
5000	27.867	259.800

Based on the result of measuring the response time, it can be concluded that the response time of web services were always faster than messaging technology. The time was synchronized with the number of data inserted, the more the records, and the more the response time needed. So it can be concluded that one to communication of web services technology was faster in data exchanging compared to messaging technology. This might happen for messaging needs JMS configuration that can rise routing overhead that make the data transferred slower. This conclusion was also supported by concept that has been discussed earlier.

VI. CONCLUSION AND SUGGESTION

Based on the result of measuring the response time, it can be concluded that the response time of web services were always faster than messaging technology. The time was synchronized with the number of data inserted, the more the records, and the more the response time needed. So it can be concluded that one to communication of web services technology was faster in data exchanging compared to messaging technology. This might happen for messaging needs JMS configuration that can rise routing overhead that make the data transferred slower. This conclusion was also supported by concept that has been discussed earlier.

For further development, the comparison of web services and messaging can be tested with multiple number of users accesses the application so that messaging technology can apply publish/subscribe model with topic as server destination.

ACKNOWLEDGMENT

Author thanks to the members of database system and information management cluster of study area who reviews and suggestions before submitting this paper. I would also thanks my former students Rizal, Budi and Dewi who proofed as hard worker in finishing this research.

REFERENCES

[1] "Service-Orientation Design Principles", http://www.soappatterns.org, diakses 29 Oktober 2010. →13

[2] The Java EE 5 Tutorial For Sun Java SystemApplication Server 9.1. Sun Microsystem, 2007

[3] “Web Services architecture overview”, http://www.ibm.com/developerworks/webservices/library/w-ovr/#author1, diakses 20 November 2010
[4] A. Chris,“Socket Connection”, http://www.eggheadcafe.com/community/aspnet, diakses 18 November 2010
[5] http://www.infoq.com/presentations/SOA-Business-Autonomous-Components, diakses 5 November 2010.
[6] G. Dejan, “Introduction to MSMQ”, http://www.cogin.com/MSMQ/QueueExploreer, diakses 18 November 2010.
[7] “Enabling Real World SOA through the Microsoft Platform”, Microsoft Corporation, 2006.
[8] G. Dejan, “Introduction to MSMQ”, http://www.cogin.com/MSMQ/QueueExploreer, diakses 18 November 2010.
[9] B. David, ”Web Services Architecture”, http://www.w3.org/TR/ws-arch, 2004, diakses tanggal 20 November 2010.

Rosni Lumbantoruan was born in Tarutung, Indonesia 24 October 1982. She has been worked as a lecturer in Del Institute of Technology since 2007 till present at Information System department, Indonesia. She granted the bachelor degree of Engineering Informatics from Bandung Institute of Technology in 2007 and master degree of Information System Development from HAN University of Applied Sciences, The Netherlands in 2010.

She was also assigned as the HEAD of database systems and information management cluster of study area at her institute since 2010, her mainly interest were business intelligence application, information extraction, cyber-security, and information system development.