Scientific
Research
Publishing

Designing Conceptual Framework for Aligning Service Oriented Architecture with Business Process

Zulqarnain Abdul Jabbar[1*], Manoj Kumar[1], Asia Samreen[2]

[1]Department of Computer Science, Muhammad Ali Jinnah University, Karachi, Pakistan
[2]Department of Computer Science, Bahria University, Karachi, Pakistan
Email: [*]z_ul_qarnain@hotmail.com

Abstract

A problem in most of the organizations is to find the best way to mature their business processes. Process maturity can make the organizations more profitable and competent for other competitors. The problem can be solved by aligning service oriented architecture (using web services) with business processes within the organization. The purpose of this paper is to design a conceptual framework for aligning business processes with the Service Oriented Architecture and then prove the framework's validity using simulation. An organization is working in education sector and wants to integrate its multiple web applications to allow data flow from one system to another. The framework will help the organization to align business process of service oriented architecture and to easily mold their business process according to business needs. The framework's validity will be proved by simulating the entire process using Yet Another Workflow Language (YAWL).

1. Introduction

A business process is a collection of activities that takes one or more kinds of input and creates an output that is of value to the customer [1]. Business Process Reengineering (BPR) is a process-centered change management tool that is used to control the dynamic behavior of business processes and that allows us to examine business

---

[*]Corresponding author.

processes and redesign them, and thus can improve the service effectiveness and cost efficiency.

Section 1 gives the introduction of Service Oriented Architecture with basic and enhanced framework of SOA. In Section 2, we will discuss various issues related to business process reengineering, service oriented architecture and enterprise service bus. Section 2 will summarize the research done in the domain of BPR, SOA's approaches/structure, failure/success indicators for SOA, SOA impacts and its implementation in different industries. In Section 3, we will discuss the methodology used in our research. Section 4 will simulate the implementation of Service Oriented Architecture. Conclusion and future work will be presented in the next section.

1.1. Service Oriented Architecture

Service Oriented Architecture (SOA) is a services based software integration solution for most of the systems. It is a loosely-coupled architecture designed to meet the business needs of the organization.

Key features of Service oriented architecture are:
- Divides our code in reusable modules and encapsulates design decisions in modules.
- Modules should be combined with each other following loose coupling for different purposes.
- Easily respond to changes and make maintainability easier.
- Improves productivity, agility, and speed for both business and IT.
- Aligns IT services to business and delivers faster services.

Reusability, flexibility and maintainability are the main powers of service oriented architecture.

SOA can provide following benefits to business.

1) Efficiency: Transform business processes from replicated processes into highly leveraged shared services that cost less to maintain.

2) Responsiveness: Rapid adaptation and delivery of key business services are to meet market demands for increased service levels to customers, employees, and partners.

3) Adaptability: More effectively rollout changes throughput the business with minimal complexity and effort, saving time and money.

Basic interactions of web services in SOA are given in **Figure 1**.

1.1.1. Basic Framework of SOA
Service providers publish their services in a service directory called Universal Description, Discovery and Integration UDDI. Services' description and their usage are defined using Web Service Description Language in XML format. Service requester search for the desired service in service directory and send a request to the service provider to bind or use the service. Message passing between service provider and service requester is managed by the SOAP protocol in XML format.

1.1.2. Enhanced Framework of SOA
There could be multiple requesters for a service at a time and it is difficult for the provider to serve all requestors without any delay. In this situation, ESB Enterprise Service Bus serves as a middleware between service provider and requesters and is responsible to manage connection and interaction between service provider and service requesters. It is like a Hub in networking.
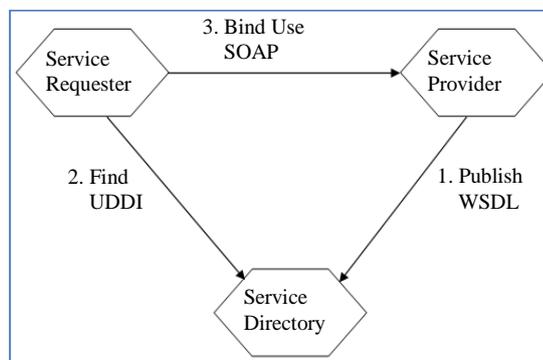

Figure 1. Framework of SOA.

Some web services standards discussed in different research are; WSDL, SOAP, UDDI, ESB, and BPEL.

WSDL: Web service description language is an XML file used to define what service is and how to interact with it [2]?

UDDI: Universal Description, Discovery and Integration. It is used to register web services.

ESB: ESB serves as a middleware in SOA. Service provider publishes services on ESB and service users can invoke the services on the ESB according to their business requirements [3]. ESB provides a set of infrastructure capabilities, implemented by middleware technology, that enable the integration of services in an SOA.

SOAP: Simple Object Access Protocol provides a framework that manages the interaction between requester and responder through message passing. An example of SOAP format is in **Figure 2**.

BPEL: Business Process Execution Language is an XML based standard for defining business processes. It supports processes which exchange information by using web service interfaces [4].

2. Review of Various Issues

We studied various issues regarding SOA. These issues are very important because to design a framework based on web-services it requires to tackle critical complications carefully.

2.1. Business Process Reengineering

Business process is the most abstract entity of process technology. Two important aspects of dynamic behavior of a business process are, 1) variability (e.g. customer arrival or customer service time), 2) Interdependence (number of decision points that affect the overall performance of the system) [5].

In order to control this dynamic behavior of business process, a methodology of Business Process Reengineering BPR was introduced to accommodate changes to business processes and to align the business processes with the strategic aims of the organizations. Improvements in efficiency, effectiveness, productivity, speed of services, and cost savings are the primary objective of Business Process Reengineering [1]. Business Process Reengineering implementation has been done in many organizations especially in the western world in order to improve the organizational performance [6].

2.2. Enterprise Service Bus

Service Oriented Architecture SOA is a software integration solution for most of the systems. It is a loosely-coupled architecture designed to meet the business needs of the organization. SOA is defined as an approach used to fulfill the requirements of loosely coupled, standard-based distributed computing which is protocol-independent as well. Enterprise Service Bus (ESB) serves as an integration and communication backbone in SOA approach. ESB utilizes web services standards to support communication patterns over multiple transport protocols and is responsible for the control, flow, and translations of all messages between services using different messaging protocols. SOA is a flexible architecture in terms that larger applications are modularized into services to unify business processes [7].

2.3. Construction Principles for SOA

Architecture is basically a workflow framework including standards, processes and procedures for implementation and SOA is one of the widely used architecture in enterprises that can effectively handle the rapid changes in architecture. Load testing, performance testing, and security testing are mostly done for SOA based systems.

```
http://mygateway.com/services/createOrder
<order>
     <id>1234</id>
     <customer_id>AB35</customer_id>
     <line>
          <item>
               <part_no>127.871</part_no>
               <quantity>2</quantity>
```

Figure 2. Example of SOAP.

Service level agreement SLA is an agreement between consumer and service provider that is used to verify the QoS of services [2]. SOA allows business transactions among loosely connected service in heterogeneous environment. Flexibility and agility are the strengths of the architectural model provided by SOA to implement business process. Design principles of SOA are service contract, loose coupling, abstraction, reusability, autonomy, statelessness, discoverability and composition [8].

2.4. Factors to Be Considered to Implement SOA

Some researchers discuss the impacts of the SOA implementation in organizations and the factors causing the success or failure of SOA implementation. Many factors are considered as essential success factors for SOA implementation. Some of the factors suggested by different authors are; reusability, complexity, governance, lack of skills in developers, lack of expertise and possible business models, organizational strategy/structure/culture, granularity of services, integration of applications, network governance, agile development, stakeholder's participation, business leadership, security, web services standards, system potential and strategy for migration, risk assessment and complexity increment. Architects and SOA implementers must have to consider the essential factors for SOA implementation as per business domain [9].

2.5. Information Security

Information security challenges should be considered during SOA implementation. Some challenges such as loosely coupled services, service contract, abstraction, reusability, autonomy, statelessness, discoverability, and composition have been discussed in literature. Some of the controls for providing solutions for these challenge are; SOA information security governance, SOA information security management, SOA information security model, and Policy information security framework [8].

2.6. Limited Recourses of Small Devices

Mobile (a low band-width and low computational device) data streaming applications can face problems in SOA environment due to the difference of attributes of service provider and consumer. For example, service provider provides video stream at 150 MB/sec but the consumer speed is just 5 MB/sec. A generic framework prototype is presented for managing and disseminating streaming data within an SOA environment [10].

2.7. Rapid Change in Consumer Demands

As the enterprises approaches towards business globalization, the rapidly increasing consumer demands become a big challenge. Solution for this problem is the agility of business and IT. Integration of traditional and new IT systems is a major issuer so the integration of logistic information system is a crucial issue for modern business. Logistic information system has several subsystems like purchasing, warehouse etc. and needs to contact with third parties like suppliers, customers etc. A general design based on SOA and ESB can be described for the integration of distributed logistic information systems [11].

2.8. Areas of SOA Application

Problems associated with the development of e-government in Southern African Development Community are analyzed and an SOA framework is proposed as a solution that can integrate back-end legacy systems and deploy services [12]. E-banking can provide many benefits to customers, banks, enterprise and also offers green banking. SOA adoption can provide many benefits to e-banking like efficiency, agility, flexibility etc. [13]. Non-integrated nature of healthcare systems causes loss of tens of thousands of patients per year due to the medical errors. So there was a need to increase the success rate of SOA adaptation in healthcare systems. SOA implementation can extend the body of knowledge in healthcare organizations [14]. An architectural baseline is proposed by Kim *et al.* to design and implement the SOA based Operation Support System model for telecommunication providers'. The model can be extended to other domains like diverse service provisioning [15].

3. Methodology/Modeling

In this research, we are going to suggest a workflow to an organization to help them out to integrate multiple

software systems based on Service Oriented Architecture and will present the simulation of the suggested workflow.

3.1. Background Knowledge of Organization

A well-known and well managed organization is working in education sector and providing free education to needy children all over the country. The organization is non-profitable and running its setup based on the donations collecting from donors.

3.1.1. Basic School Information
Classification of the educational units in this organization is described in **Table 1** below.

3.1.2. IT Landscape
School Management System (SMS) is the center point of their IT landscape, enabling them to input relevant school information and disseminate information to all stakeholders from decision makers at the HO/RO level to those who make and implement decisions at the AO and Principal level. Current system is represented by **Figure 3(a)** while required system should be interactive that is shown in **Figure 3(b)**.

As till Dec 2014, the organization has already grown to 900 units with 160,000 students. Their growth pattern is displayed in **Figure 4**.

3.2. Scope of the Suggested Workflow

We shall cover the data synchronization task between two systems SMS and HRMIS for this organization in our suggested workflow.

Focus Areas of SMS
The main areas of the SMS and related sections are illustrated in **Figure 5**.

3.3. Workflow

3.3.1. Purpose
Integration between HRMIS and SMS will be a two way process passing selected fields from SMS to HRMIS and vice versa at predefined intervals.

3.3.2. How Will It Be Done?
Web services will be designed for pushing information to HRMIS and for pulling information from HRMIS.

Table 1. Classification of the educational units.

Classification	Definition	Naming convention
Region	Region is a geographical entity organized roughly along provincial boundaries	South, South West, North, North West
Area	An Area is an administrative unit created (and occasionally revised) to reflect an optimal composition of school units and campuses.	Based on main City in the Area, as determined by the respective Regional Manager
Cluster	Cluster is logical grouping of Campuses in close proximity (typically less than 5 km) and is meant for administrative and expansion planning	Based on main Town in the cluster.
Campus	Campus is a concrete structure designed to house a single or multiple school units.	Usually named after the Build Donor
E-Compound	E-compound refers to a group of campuses located in close proximity to another within the same boundary wall, although they operate as independent units	S/4/05 which means that this E-Compound was the fourth E-Compound made in the year 2005 in South
School	School is a function of campus (that houses the school), shift (morning, afternoon) and type (primary, secondary)	<Campus Name><Type><Shift> e.g. Khaliq Noorani Primary Morning
Unit	Unit is the most basic form of classification based on Capacity of the school	1P, 2P, 1S, 2S etc.

(a)

(b)
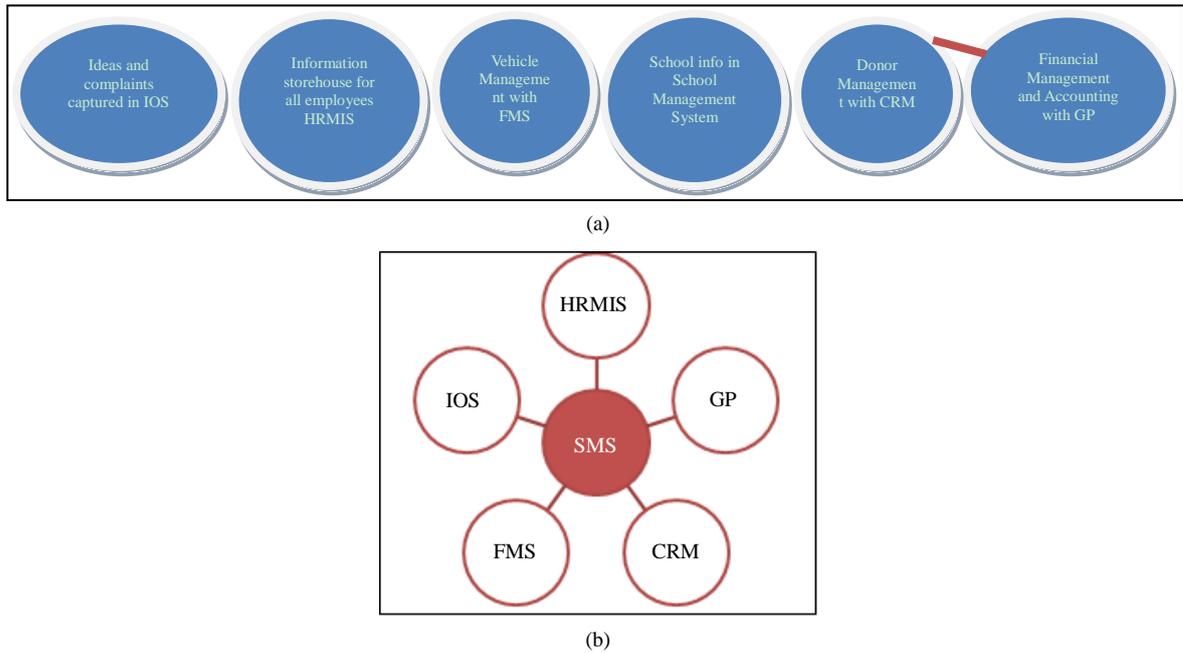
Figure 3. (a) AS-IS-IT landscape; (b) To-Be-IT landscape.
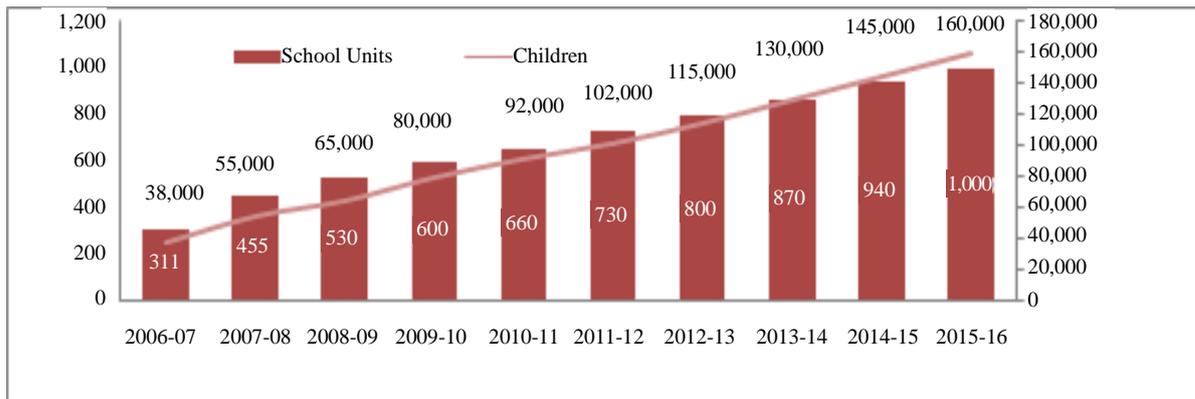
Figure 4. Growth pattern.
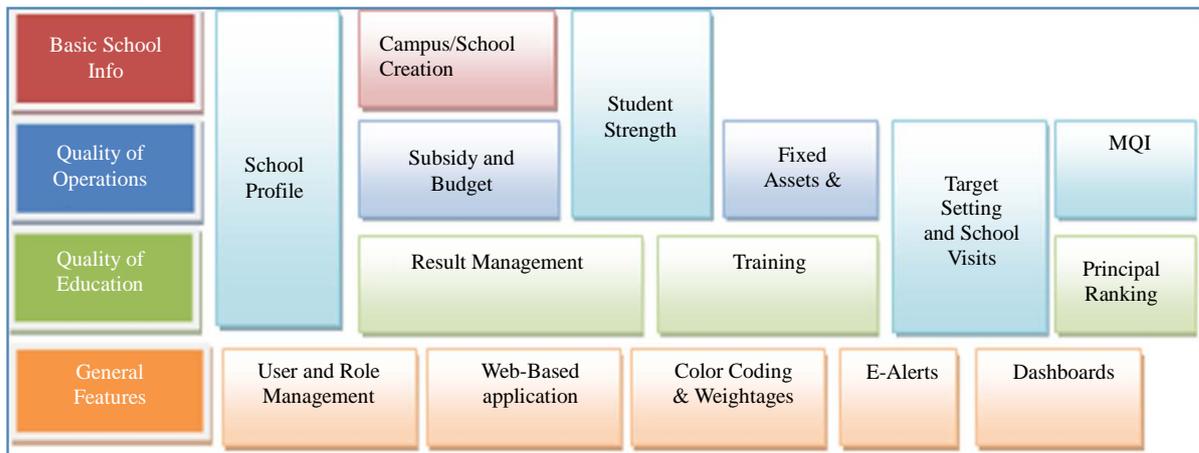
Figure 5. Main areas of SMS.

This is the most secure manner of integration as no application will need to interact with the database directly. Web service can also utilize the methods and functions already built in the system. This will also be a more flexible approach when more systems (such as CRM, GP) will require two way communications with SMS going forward. Both web services, will be registered on UDDI so that the web service can operate at all times after required authentication.

Some important features of the web service are as follows:

1) Web Service will send and receive only those fields as defined by business users of SMS and HRMIS. Ideally, as the needs of business users evolve, an appointed system administrator should have the option to add more fields so that more information can be passed/retrieved vis-à-vis the web service.

2) For SMS specifically, the web service will always update existing school information for example staff strength and staff credentials and not create new school. The web service will be able to update selected fields, not delete any data point.

3) The web service will not be event based, rather it would be linked to predefined intervals. At specified time, web services will be called to read and write data. Hence all changes done to school data within the SMS over the period will get updated in HRMIS in bulk.

It is recommended that this practice should be done thrice a year *i.e.* end of April (when new schools are created), end of August (when school information like signage are likely to be finalized), end of November (when midterm results are finalized along with some minor changes in school information).

SMS will also need HR data to be updated at defined intervals. Most likely, this information should get updated twice a year one in end of May (when teachers, faculty have been assigned to schools) and once in November (in case there is any change in faculty information).

3.3.3. Required Fields Mapping (Table 2)

Table 2. Required fields mapping.

Fields for HRMIS from SMS	Fields for SMS from HRMIS
School_Name	Principal's Contact No.
School_Full_Name	Area Manager's Contact No.
Start_Date	Regional Manager Contact No.
End_Date	Principal Name
School_Address	No. of Principal
City_ID	No. of Admin Assistant
Number_Of_Students	No. of Driver
Telephone_No	No. of Chowkidar
Emp_ID(principle)	Total No. of Staff
Country_ID	Break up of Admin staff and teaching staff
State_Province_ID	No. of Teachers
campusID	Break down of teacher by subject
CostCenterCode	Teachers' Quality: by Qualification and experience
noOfClasses (No. of Units)	Teachers' Availability: Shortages
establishmentYear	Principal Qualification
Shift	-
typeOfSchool	-
Email	-
phoneCode	-
RegionName	-
Branch_Code	-
Region_ID	-
Branch_Name	-
Area ID	-
Cluster	-
APR (Annual Percentage Result)	-
E-compound	-
District	-
Tehsil/Town	-
Location	-

3.3.4. Proposed Solution for Integration

Now we describe our solution to the problem to integrate interactively all the web-service in use for any organization. **Figure 6** shows how the web services will interact with their respective application database and over all interaction with other web services.

Here is how this will work:

1) Each application (SMS, HRMIS, CRM etc.) have its own Web Service which is responsible to handle all requests related to that application. So, SMS Web Service does not know how the HRMIS web service will update or fetch data from HRMIS database.

2) We have a "Scheduled Integration Service" which is serving a center point for communications for all the applications. Important points for this application are:

a) This will be a windows service or scheduled task which we will setup on a server or machine. We will need to discuss how to define the scheduling logic for this application.

b) This application will use the web services to retrieve and update data for one application to another application.

c) Application will maintain a log of all the operations/errors.

4. Simulation

Here we'll present a simulated workflow for the SOA implementation for that organization. Simulation is performed using the software YAWL (Yet Another Workflow Language).

4.1. Introduction to YAWL

Based on a rigorous analysis of existing workflow management systems and workflow languages, a new workflow language called YAWL (Yet Another Workflow Language) was developed by Wil van der Aalst (Eindhoven University of Technology, the Netherlands) and Arthur ter Hofstede (Queensland University of Technology, Australia) in 2002. This language was based on the one hand on Petri nets, a well-established concurrency theory with a graphical representation, and on the other hand on the well-known Workflow Patterns. The workflow Patterns form a generally accepted benchmark for the suitability of a process specification language. Petri nets can capture quite a few of the identified control-flow patterns, but they lack support for the multiple instance patterns, the cancellation patterns and the generalized OR-join. YAWL therefore extends Petri nets with dedicated constructs to deal with these patterns [16].
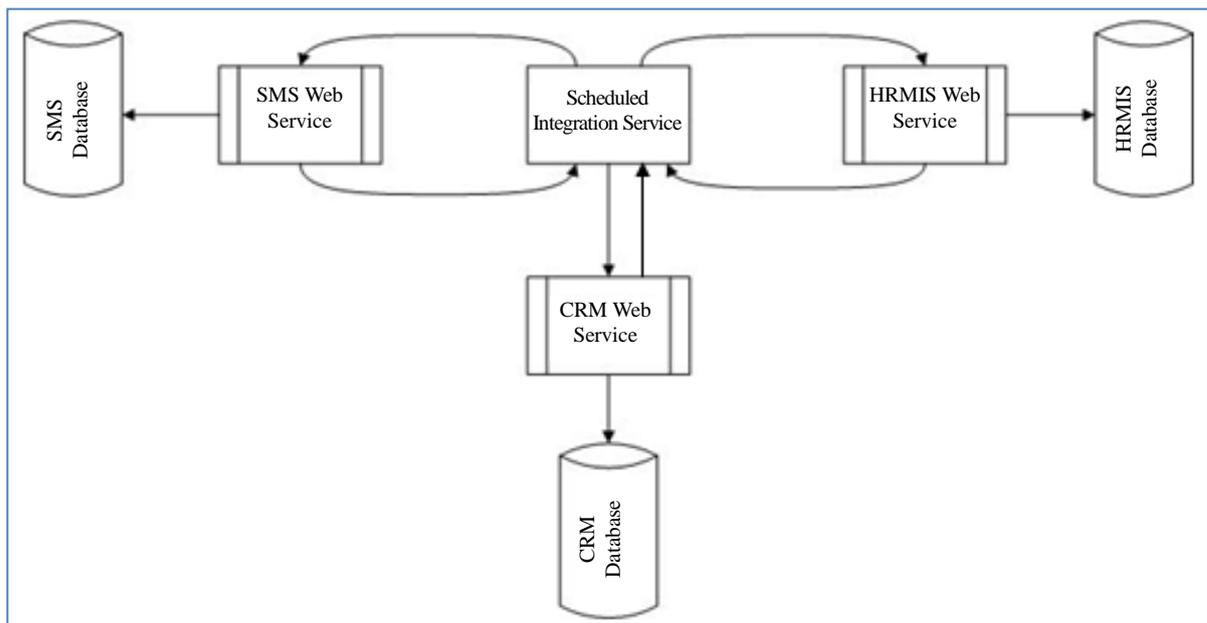

Figure 6. How web services interact.

4.2. Workflow for the Data Synchronization Using Web Services

In this section, we will describe the main workflow of our suggested framework.

4.2.1. Workflow in YAWL

This workflow presents the entire data sync process between SMS and HRMIS system. Similar work flow can be created for the data sync process between SMS and GP, SMS and FMS, SMS and CRM (**Figure 7**).

Data Sync Process for HRMIS System

Schedule Integration Service that is running on the SMS system will invoke and send a request to Request To SMS to Pull Data task with the output variable RequestType = SMS. Request to SMS to Pull Data task takes the input of RequestType = SMS and calls the Authentication 1 task with 2 output variables which are Userid = smsuser and Password = sms123. This task will authenticate the requester and invokes the SMS Web Service with output variable authentic = true. SMS Web Service gives a flag of servicestart = true to Service Pulls the Data from SMS DB task. This task will getdata in string or json string format from sms database and senddata in string or json format to the Send Data to Requester 1 task. This task will pass the senddata string to Push Received Data in HRMIS task that will finally push data in HRMIS database.

4.2.2. Simulation Result

This simulation doesn't only gives the workflow model but also gives a XML based specification. This specification clearly defines the input/output parameters and invoke condition for all services participated in the workflow. This specification will help in the final implementation of the services based model. A code snippet of the XML based specification for the service Invoke_HRMIS_Web_Service is given below:

```
<specification uri="SOA_Simulation">
<metaData>
<creator>Zul_qarnain</creator>
<description>No description provided</description>
<version>0.1</version>
<persistent>false</persistent>
<identifier>UID_63f35c80-ec0f-4773-af4e-6f2d3b84d5f7</identifier>
</metaData>
<xs:schemaxmlns:xs="http://www.w3.org/2001/XMLSchema"/>
<decomposition id="Net_SOA" isRootNet="true" xsi:type="NetFactsType">
<localVariable>
<index>0</index>
<name>RequestType</name>
<type>string</type>
```
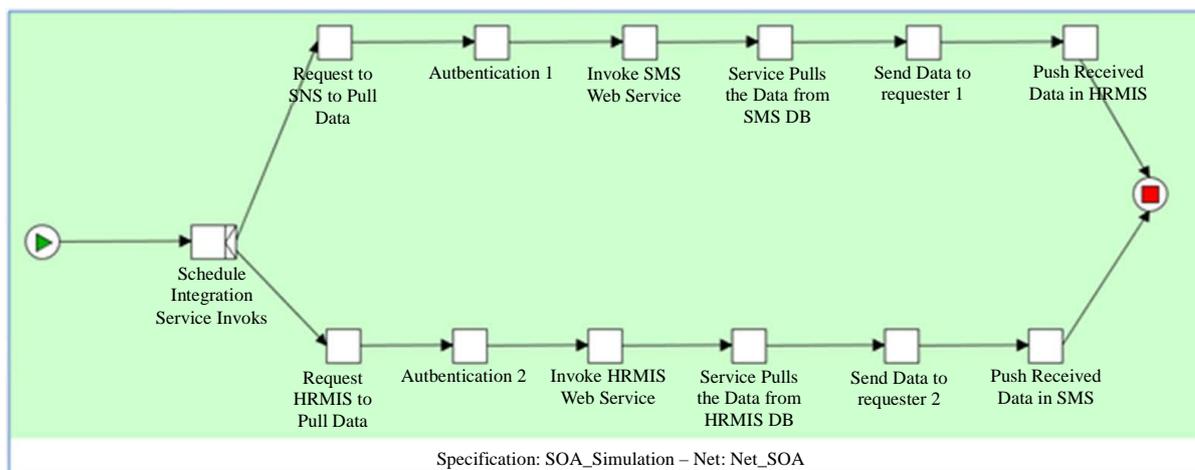

Figure 7. Example of workflow in YAWL.

```xml
<namespace>http://www.w3.org/2001/XMLSchema</namespace>
<initialValue/>
</localVariable>
<localVariable>
<index>1</index><name>Userid</name>
<type>string</type>
<namespace>http://www.w3.org/2001/XMLSchema</namespace>
<initialValue/>
</localVariable>
<localVariable>
<index>2</index><name>Password</name>
<type>string</type>
<namespace>http://www.w3.org/2001/XMLSchema</namespace>
<initialValue/>
</localVariable>
<processControlElements>
<inputCondition id="InputCondition">
<flowsInto>
<nextElementRef id="Schedule_Integration_Service_Invoks"/>
</flowsInto>
</inputCondition>
<task id="Invoke_HRMIS_Web_Service">
<name>Invoke HRMIS Web Service</name>
<flowsInto>
<nextElementRef id="Service_Pulls_the_Data_from_HRMIS_DB"/>
</flowsInto>
<join code="xor"/>
<split code="and"/>
<startingMappings>
<mapping>
<expression query="&lt;authentic&gt;{/Net_SOA/authentic/text()}&lt;/authentic&gt;"/>
<mapsTo>authentic</mapsTo>
</mapping>
</startingMappings>
<completedMappings>
<mapping>
<expression
query="&lt;servicestart&gt;{/Invoke_HRMIS_Web_Service/servicestart/text()}&lt;/servicestart&gt;"/>
<mapsTo>servicestart</mapsTo>
</mapping>
</completedMappings>
<resourcing>
<offer initiator="user"/>
<allocate initiator="user"/>
<start initiator="user"/>
</resourcing>
<decomposesTo id="Invoke_HRMIS_Web_Service"/>
</task>
<outputCondition id="OutputCondition"/>
</processControlElements>
</decomposition>
</specification>
<layout>
```

```
<locale language="en" country="US"/>
<specification id="SOA_Simulation">
<size w="58" h="28"/>
<net id="Net_SOA" bgColor="-3342388">
<container id="Invoke_HRMIS_Web_Service">
<vertex>
<attributes>
<bounds x="552" y="376" w="32" h="32"/>
</attributes>
</vertex>
<label>
<attributes>
<bounds x="520" y="408" w="96" h="40"/>
</attributes>
</label>
</container>
<flow source="Authentication" target="Invoke_HRMIS_Web_Service">
<ports in="13" out="12"/>
<attributes>
<lineStyle>11</lineStyle>
</attributes>
</flow>
<flow source="Invoke_HRMIS_Web_Service" target="Service_Pulls_the_Data_from_HRMIS_DB">
<ports in="13" out="12"/>
<attributes>
<lineStyle>11</lineStyle>
</attributes>
</flow>
</net>
</specification>
</layout></specificationSet>
```

5. Conclusions/Future Work

The dynamic behavior of the business processes can be well controlled by aligning the business processes with the web services using the service oriented architecture. SOA doesn't only allow business process reengineering in an efficient manner, but its loosely coupled nature also helps organizations to evaluate their process and to accommodate changes in process without affecting the entire architecture. Most of the organizations are getting benefit of the SOA by mapping their business processes to web services and by using a middle ware in the form of Enterprise Service Bus to control traffic (requests) to web services to serve the required purpose. In this research, we are focusing on an organization of education sector and suggesting a workflow to them for the mapping of their business processes to web services. In this workflow, we suggested a web services-based solution to them. We performed the simulation of the suggested workflow using YAWL (Yet Another Workflow Language). We gave the complete specification of the suggested workflow in XML format as well. This XML based workflow describes the data flow in this workflow. No one presents this type of workflow for the requirement of the information synchronization in multiple systems. We are the first, proposing this solution and this is an easiest and cheapest solution for the automated information synchronization among multiple systems.

In next phase, we will present a workflow for the dynamic service replacement in SOA.

Acknowledgements

In the last, we are also thankful to our family, friends, fellows, and colleagues who supported us in their best.

References

[1] Hammer, M. and Champy, J. (1993) Reengineering the Corporation: A Manifesto for Business Revolution. Harper Collins Books, NY.

[2] Parveen, T. and Tilley, S. (2008) A Research Agenda for Testing SOA-Based Systems. *Proceedings of System Conference* 2008—*IEEE international Systems Conference*, Montreal, 7-10 April 2008, 1-6. http://dx.doi.org/10.1109/SYSTEMS.2008.4519032

[3] Zhu, C.S., Chai, M.M., Lu, Y.F. and Guo, Y.D. (2013) Service Oriented Architecture Design of Energy Consumption Information System about Petroleum Enterprise. *IEEE International Conference on Computational and Information Sciences*, Shiyang, 21-23 June 2013, 1174-1177. http://dx.doi.org/10.1109/ICCIS.2013.311

[4] Arslan, F. (2012) Towards Service Oriented Architecture (SOA) for Massive Multiplayer Online Games (MMOG). *IEEE* 14*th International Conference on Modelling and Simulation*, Cambridge, 28-30 March 2012, 538-543. http://dx.doi.org/10.1109/UKSim.2012.82

[5] Greasley, A. (2003) Using Business Process Simulation within a Business Process Reengineering Approach. *Business Process Management Journal*, **9**, 408-420. http://dx.doi.org/10.1108/14637150310484481

[6] Weerakkody, V. and Currie, W. (2003) Integrating Business Process Reengineering with Information Systems Development: Issues and Implications. *BPM* 2003, 302-320.

[7] Papazoglou, M.P. and ven den Heuvel, W.-J. (2007) Service Oriented Architecture: Approaches, Technologies and Research Issues. *The VLDB Journal*, **16**, 389-415. http://dx.doi.org/10.1007/s00778-007-0044-3

[8] Chetty, J. and Coetzee, M. (2010) Towards an Information Security Framework for Service Oriented Architecture. *Information Security for South Africa*, Sandton, 2-4 August 2010, 1-8. http://dx.doi.org/10.1109/ISSA.2010.5588272

[9] Waris, M., Khan, S.A. and Fakhar, M.Z. (2013) Factors Effecting Service Oriented Architecture Implementation. *Science and Information Conference*, 1-7.

[10] Ahmed, N., Linderman, M. and Bryant, J. (2010) Towards Mobile Data Streaming in Service Oriented Architecture. 29*th IEEE International Symposium on Reliable Distributed Systems*, New Delhi, 31 October-3 November 2010, 323-327. http://dx.doi.org/10.1109/SRDS.2010.45

[11] Zhang, L.Y., Li, J.Q. and Yu, M. (2006) An Integration Research on Service Oriented Architecture (SOA) for Logistic Information System. *IEEE International Conference on Service Operations and Logistics*, *and Informatics*, Shanghai, 21-23 June 2006, 1059-1063. http://dx.doi.org/10.1109/SOLI.2006.328898

[12] Ramtohul, A. and Soyjaudah, K.M.S. (2013) Service-Orientation Method to Realise Government e-Services in SADC. *VINE*: *The Journal of Information and Knowledge Management Systems*, **43**, 237-258. http://dx.doi.org/10.1108/03055721311329972

[13] Basias, N., Themistocleous, M. and Morabito, V. (2013) SOA Adoption in e-Banking. *Journal of Enterprise Information Management*, **26**, 719-739. http://dx.doi.org/10.1108/JEIM-07-2013-0042

[14] Koumadities, K. and Themistocleous, M. (2013) SOA Implementation Critical Success Factors in Healthcare. *Journal of Enterprise Information Management*, **26**, 343-362. http://dx.doi.org/10.1108/JEIM-06-2012-0036

[15] Kim, J.W. and Lim, K.J. (2007) An Approach to Service-Oriented Architecture Using Web Services and BPM in the Telecom-OSS Domain. *Internet Research*, **17**, 99-107. http://dx.doi.org/10.1108/10662240710730524

[16] The YAWL Foundation (2004-2014) YAWL-User Manual.