Data Exchange Model of Patient Records in Kingdom of Saudi Arabia Using Cloud Computing

Hosam F. El-Sofany, Khalid F. Al-Otaibi, and Majed S. Alsanea

***Abstract*—The main objective of this research study is the development of conceptual framework for the exchanging of patient records located in different hospitals all over the Kingdom of Saudi Arabia. The proposed framework is aimed to improve the way of retrieving the patient medical records from different health information system. The proposed architecture, designed to highlight the method by which data should be searched and retrieved efficiently from the different health information systems. Our system design is based on Cloud Computing Service Oriented Architecture. These medical systems storing the medical information records including: demographics medical history, medication and allergies, immunization status, laboratory test results, radiology images, vital signs, personal stats like age and weight. All of these medical records are identified by National ID number. These systems will be utilized by the web services XML based approach. The patients will use their ID number to send request for their medical informatics and the web service will analyze the patient records and send the result back to the patient. The main contribution of this study is to provide a data exchange model of patient records, this model used to decrease the cost and time of patients, and help patients to get its medical records information from any location by using the Web.**

***Index Terms*—Data Exchange System (DES), Service Oriented Architecture (SOA), Cloud Computing, XML, Electronic Medical Record (EMR).**

I. Introduction

In the recent years the need of electronic patient data exchange has been increased tremendously. The patients are very interested to know their old medical history. As a result of this needed from the patients, health organizations are going towards implementing patient data exchange systems. According to the World Health Organization (WHO), the main Information and Communication Technologies (lCT) challenge is to enable "more informed decision-making and more cost-effective use of resources" [1]. To fulfill these objectives, ICT shall in particular track and provide patient information, facilitate the communication between the patients and the physicians, and between the latter, and deliver a quality of services despite the distance and the time. However, establishing an automatic and transparent data exchange between health information systems remains a challenging issue for clinical practice.

In addition to standard Electronics Data Interchange (ED!) solutions, using Health Level Seven (HL 7) for example, health institutions and organizations are increasingly adopting Web services and its eXtensible Markup Language (XML) based technologies to communicate and exchange data [2], while their underlying information systems are usually storing data in relational databases which are still the most common data storage technology. Indeed, there exist thousands of heterogeneous legacy data management systems that have been set up to optimize the efficiency of the practice of different specialties, which is an obstacle towards open data interchange and integration. But the information society is at present promoting data exchange between the different specialties, both to improve the quality of care and, in the meantime, to reduce their cost [1].

The main objective of this study is to develop a model for a proposed patient data exchange system, which will rely on web services and SOA, cloud computing based architecture to retrieve patient records from different health care organizations located in different parts of Saudi Arabia.

The paper is organized as follows: in section two we present condensed survey and previous works related to exchanging of medical patient records. In section three we introduce the data exchange model of patient records in KSA. In section four we present the quantitative evaluation and analysis of the proposed model. In section five we show the system limitation. The paper finally concluded in section six with conclusion and future works.

II. Previous Works

In this section we present some research studies and previous works that have been conducted in the same domain of our research area.

C. Oberdan Rolim, F. Luiz Koch, C. Becker Westphall, J. Werner, A. Fracalossi, G. Schmitt Salvador [3], conducted a research on current medical processes which are responsible for gathering patient's data and they came to the conclusion that the current processes are slow and more error prone. So they proposed a cloud computing solution for patient's data collection in health care institutions. They also figured out that it requires lot of labor work to collect, input and analyze data. So they proposed a solution to automate this process by using sensors attached to existing medical equipment that are inter-connected to exchange service. Their proposed solution is based on the concepts of utility computing and wireless sensor networks. The information becomes available in the cloud from where it can be processed by expert systems and/or distributed to medical staff. The study focused on the problem of patients' vital data collection, distribution, and processing. They suggested that current solutions based on manual note taking are slow, time consuming, and labor resource intensive.

Manuscript received September 3, 2012; revised October 10, 2012.

Hosam F. El-Sofany, Khalid F. Al-Otaibi, and Majed S. Alsanea are with Arab East Colleges for Graduate Studies, Department of Computer Science, Riyadh, Kingdom of Saudi Arabia (e-mail: hosam_elsofany@hotmail.com).

DOI: 10.7763/IJIMT.2012.V3.301

The article proposed a system to automate the process of collecting patient's vital data via a network of sensors connected to legacy medical devices, and; deliver this information to the medical center’s cloud for storage, processing, and distribution. The Exchange Service acts like a broker between local and remote services. It is responsible to receive collect data from sensors and to dispatch it to appropriate storage service hosted on cloud. It also receives requests from content service to retrieve data from the Cloud Service, whose functionality is two folded:

1) It is responsible to provide services to store collected data.
2) It provides a platform for development, testing and deployment of applications needed by medical staff. Mobile and stationary devices interacts with applications using Content Service. This service acts like a door where medical staff devices can access all available information.

T. Takase, K. Tajima [4], conducted a research study on efficient web Services message exchange by SOAP bundling framework. They proposed a SOAP message bundling framework. Their framework enables bundling multiple messages into one message. With this framework, application developers do not have to consider the service granularity. Instead, the framework bundles some fine-grained messages into a single coarse-grained message. To support this framework, they provided for service providers a WSDL conversion tool and a skeleton wrapper generator. These tools let service providers receive bundled messages without modifying existing service implementations. They also provided a stub wrapper generator that allows service requesters to use bundled services easily.

Currently, Web services are spreading widely throughout the world. Web services are an enabling technology for interoperability within distributed, loosely coupled, and heterogeneous computing environments. However, previous studies have shown that the performances of Web services are relatively poor because of the encoding processing and network latency. Therefore, technologies to improve the performance of web services are needed. In this paper, they described a message bundling framework appropriate for Web services architectures. The most important point of Web services is interoperability. Web services are based on certain specifications, such as XML as a message format, SOAP as a message layer protocol, and WSDL as a description of the interfaces. Once a service application is developed as a Web service, the interface description can be published for external partner companies and the service can easily be provided to new partner companies.

The authors presented many problems in the previous researches which were conducted in the same subject. Some researches pointed issues regarding the document-oriented style of communication in web services. Some of the issues were raised for the client side caching in the web services. In this framework, the author provided a WSDL converter and a skeleton wrapper generator for service providers. The WSDL converter adds new bundled service interfaces to an existing WSDL document. The added bundled service interface bundles all of the existing service interfaces in the WSDL document. Specifically, a bundled operation is added. Here, operation is a WSDL term, but generally this has the same meaning as function or method. The input parameters for the bundled operation include parameters for the arbitrary number of operations which exist in the service. The return value of the bundled operation can also hold arbitrary numbers of return values for all existing operations. A skeleton wrapper generated by our skeleton wrapper generator implements the added operation for the WSDL converter.

In the most naïve implementation, the skeleton wrapper sequentially calls the existing operations internally. If the platform this skeleton is deployed on offers advantages for multi-thread operations, as in a multi-core processor environment, the implementation should exploit this. However, even in the most naïve implementation, their bundling approach improves the performance.

The skeleton wrapper provides an implementation for the bundled operation by using the existing operations. The service provider’s steps are:

1) Convert the existing WSDL document using the WSDL converter and re-publish the converted WSDL.
2) Generate the skeleton wrapper with the skeleton wrapper generator and deploy the generated skeleton wrapper into the service platform.

L. Hu, P. Li, Y. Wang [5], presented a research on design and implementation of a SOA-based data exchange middleware. They proposed a service-oriented architecture for data exchanging between applications using heterogonous databases. Heterogeneous data makes it difficult to be shared across organizations. In these organizations, data is often produced with diverse applications relying on a mix of software and information systems. Each application uses its individual system to increase efficiency, but sharing data across the enterprise is a near impossibility. They presented a service oriented architecture of using web services which has been adopted for integrating diverse repositories of data.

Data is increasingly becoming available on the Internet, allowing a large number of users to share and access the rich databases that currently being maintained in several organizations. However, data is immensely heterogeneous, being available in various formats and stored in diverse media (flat files, relational database). In this paper they have followed a Service-oriented architecture (SOA) using web services for integrating diverse repositories of data. The data repositories are available on the web as services with some well-defined interface. It is based on XML technology and a client can access any data repository having data in any format located on any platform if it only knows how to communicate with service provider. The goal is to provide unified access to data from heterogeneous data providers. Any client (user) can submit its query to a server. Service oriented architectures involve three different kinds of actors: service providers, service requesters and discovery agencies. The service provider exposes some software functionality as a service to its clients. In order to allow clients to access the service, the provider also has to publish a description of the service. Since service provider and service requester usually do not know each other in advance, the service descriptions are published via specialized discovery agencies.

L. Hu, P. Li, Y. Wang, [5], proposed SOA-based model of data exchange middleware (DEM) that offers applications and services participating in a SOA to exchange any kind of data, including relational, XML, or application-specific data objects, from various kinds of data sources in a consistent, standards based way. It is this middleware that enables services to access heterogeneous data sources in the most efficient, flexible way possible.

S. Qian, Z. Fu [6], conducted a research on design and implementation of the data share and data exchange system based on SOA. They analyzed the function of Data share and data exchange systems. The system architecture is based on SOA. They also highlighted the main problems which occur in the implementation of SOA based architecture. In this study the authors shown that, *data sharing platform* is infrastructure for management of information resources, supporting resource sharing exchange, which is unified administrated by related agencies. According to the uniform standards and specifications, the public directory library, where all kinds of resources sharing data catalogues are stored, is constructed. According to needs of data sharing platform, we make the related resources information generated corresponding resource cataloguing, directory, and carries on the unified management, to provide accurate resource directory release, discovery and positioning services, and resource directory access control. Data provide department establishes resource directory structure and registration system according to data sharing platform, registers in data sharing platform directory content, and is responsible for the after maintenance work. The one who needs the resources calls data sharing platform provides directory services, searching information resource directory, positioning directory content of information resources, associated get information from the relevant system, so as to achieve the information sharing. Information is deployed on data sharing nodes according to uniform standard, there are two kinds of information: data resources and the service resources. Generally speaking, the data sharing nodes need to deploy a single server, its resources are voluntary released and maintenance by management department. Data sharing and exchange system provides different roles in different users access method, users can be divided into three levels according to different needs, which are system administrator, system maintainer, ordinary users. Ordinary users can also support data access and data exchange two roles. Administrator is responsible for management of various users, system maintainer is responsible for the management of system and the various data shared resource nodes directory, ordinary users are limited to access resources. Resource list format: resource list is the directory structure mode that the user access to, a system maintainer provide Shared resource directory for frontend client. At present catalogue of share files basically include text document (doc), Excel files and database files.

H. Xu, Y. Tian, G. Dong, Y. Wang [7], conducted a research on schema of data exchange for heterogeneous data. They came to conclusion that there are many difficulties in data exchanging and sharing among health and medical information systems. Their research introduced some key techniques of the solution, which include the adoption of P2P communication mode, the framework for heterogeneous data integrating and semi-structured data mapping. These techniques employ distributed data storage and XML technology, which makes the system low in cost, easy to implement and excellent in scalability.

To solve this problem H. Xu, Y. Tian, G. Dong, Y. Wang [7], introduced a need to screen the specific techniques and their implementation modes applied by different information systems. They also emphasized a need to study the key exchange technique between health files and hospital medical records in order to realize classified medical treatment and referral constitution and finally to get a new medical set-up of minor illness in the community, serious illness in the hospital, recovery back to the community. The implementation of the system management of community health files will effectively solve the problem of difficult and expensive medical treatment for the public to increase the medical health security. It will promote the development of medical infrastructure, new rural cooperation medical system and community medical treatment. The solution plan of community health files management needs to consider the problems such as schema compatibility, data heterogeneity, the convenience of penetrating fire wall and the rest etc. The authors of the research gave a detailed description of the key techniques involved, including the access modes of different medical and health systems (HIS, LIS, CIS, PACS, RIS etc.); the transformation and integration of heterogeneous data stored in different systems.

H. Xu, Y. Tian, G. Dong, Y. Wang [7], introduced data integration design adopts the layered architecture. This architecture includes application layer, mediator layer and data layer. This architecture has a better portability which enables the system to be run on a variety of platform. They proposed a system architecture which is based on three layers: *The application layer*, *mediator layer* and *data layer*. *Application layer* is the highest layer of the architecture which consists of the web client component. This layer is designed specifically to enable direct user computing through graphical user interface in web environment. The web client component communicates with the user interface of the mediator layer by forwarding users' requests and receiving the returned results from the user interface in XML schema. In doing so, cross systems interoperability can be achieved with XML flexible data model. This layer is the core layer of the architecture. Its primary function is to realize the connection between data layer and application layer. This layer consists of four components, namely, the user interface, the manager, and the searcher and metadata dictionary. The user interface is responsible for displaying the overall schemas of various information sources to users, generating global transactions associated with the user's requests. The user's information and a global transaction would be packed into a user's package before forwarding to the manager. The user interface subsequently forwards the results obtained from the manager to the upper layer web clients. Data layer consists of heterogeneous data sources and wraps. Each wrap has direct connection with data source. Their major task is to serve the requests from the searcher. It would change the result returned from data source into an XML-based data schema and then wrap the result to send it back to the searcher.

Q. Guo, G. Chang, D. Sun, X. Wang [8], conducted a research on shared service architecture for emergency management system development. They realized that it needs good data integration and application integration for the existing applications, and it is very effective for a given period of time. This paper discusses how to establish the cross application system and rapid integration through the shared service architecture based on web service. The architecture has important reference significance for e-Government integration of cross application systems.

In recent years, along with the vigorous development of e-Government systems, many vertical application systems have become the core of e-Government daily operation systems. To deal with major emergent events effectively, a new requirement is raised to the e-Government system of local government. The authors mainly discussed how to use the unified data platform technology which is based on Web Service in the upgrading of emergency management systems to establish a cross application system. Along with the rapid development of Internet-based applications, service oriented architecture (SOA) has become a mainstream style of architecture for the development of various distributed software system. The authors proposed an architecture to reduce the difficulty of the system integration by establishing a shared data and business service layer so as to provide fast and flexible platform to the unified decision-making. They discussed some related problems with establishing unified data platform for emergency management systems by using this model.

According to the analysis of the emergency management system structure, characteristics and application requirements, based on typical shared service architecture in SOA, the authors designed an overall architecture of the emergency management system. In this architecture, all the applications of emergency management Centre are defined as independent services, which have some well-defined inviolable interfaces. The emergency management Centre can call the services in a certain defined order to form business flow, in order to have the ability to handle the business process flexibility. This architecture mainly consists of three levels, the heterogeneous resources layer, the unified integration layer, and the unified access layer. The heterogeneous resource layer includes various specialized emergency management system, e-Government systems, special-purpose systems, the database systems under development, and so on. It is the general name of the heterogeneous resources waiting for integration on the background. The unified integration layer layer includes mainly four aspects, the central enterprise service bus, the central data service platform, the shared business service, and the shared application service. The main functions are as follows. The central data services platform is responsible for the integration of data in the heterogeneous resources layer. In a narrow sense, these data can be understood as those in the heterogeneous database systems. In general, these data may include from the data in the databases to the data coming from application system interfaces. These integration data are packed into the services, and registered to the enterprise services for the shared business services or the shared application services. The shared application services platform packs all the existing application systems into services, and these services are registered to the central enterprise service for the shared business services. The central enterprise service bus is a bus for all the application services, data services, business services, and portal services in the whole centre. The invocations of portal or web applications to the services and services to services all depend on it. The shared business services reflect the functions of business flow. They are the integration of shared application services and data services and can be used by portal or other applications as an advanced pattern. The unified access layer can adapt to the flexible and diverse user access methods. Through the related service channel, it provides unified visit entrance (Portal) for users, integrates diverse user interfaces of business functions, and establishes a cross application, cross equipment, integrated, and interactive user interface. Based on in-depth analysis on the characteristics and requirements of the emergency management centre, based on SOA and the shared services architecture, the literature proposed the architecture for the emergency management system [8].

A. Yahia, S. Kotidis, Y.Res [9], conducted on efficient web services architecture for XML data exchange. Business applications often exchange large amounts of enterprise data stored in legacy systems. The advent of XML as a standard specification format has improved applications interoperability. However, optimizing the performance of XML data exchange, in particular, when data volumes are large, is still in its infancy. Quite often, the target system has to undo some of the work the source did to assemble documents in order to map XML elements into its own data structures. This publishes & map process is both resource and time consuming. In this paper, we develop middle-tier Web services architecture to optimize the exchange of large XML data volumes. The key idea is to allow systems to negotiate the data exchange process using an extension to WSDL. The source (target) can specify document fragments that it is willing to produce (consume). Given these fragmentations, the middle-ware instruments the data exchange process between the two systems to minimize the number of necessary operations and optimize the distributed processing between the source and the target systems. We show that our new exchange paradigm outperforms publish &map and enables more flexible scenarios without necessitating substantial modifications to the underlying systems.

Large organizations use a plethora of systems to support their daily operations. Depending on the application, a system may act as a data broker by disseminating information that is consumed by the receiving applications. For instance, in a telecom provider like AT&T, a sales and ordering system provides an interface to extract data on customer orders accounts in order to collect revenue. In such real-world applications, the data that is exchanged can reach very large volumes. As an example, usage data from the telephony network easily exceeds 60GB per day. In this paper, we focus on the optimization of data exchange between two applications when the amount of data is large. In order to collaborate, applications implement pair-wise agreements that define the format of the data to exchange. To this end, XML is most commonly used. Web services are typical examples that use XML as the grammar for describing services on the network as a collection of systems capable of exchanging data and messages. The specification of a Web Service Description Language (WSDL) document hides the details involved in data communication by focusing on the format in which that data is being produced and consumed and the services that are provided at each endpoint. However, optimizing the performance of exchanging large data volumes has not attracted a lot of attention. Quite often, the target system has to undo some of the work the source did to assemble documents in order to map XML elements into its own data structures. This process is both resource and time consuming. In a typical data exchange scenario, that we will refer to as publish & map, XML documents are built at a source application and shipped to be consumed at a target one. The process of publishing an XML document from stored data translates often too costly combine operations (through joins in case of relational stores) that piece document fragments together. Quite often, some of these fragments are stored similarly at the source and at the target systems, in which cases combining such fragments at the source is unnecessary because the target system will split them again into its internal structures. Furthermore, in publish & map, XML documents are built at the source and consumed at the target, imposing a strict processing distribution that does not explore the capabilities of the underlying systems. This data is used to drive a provisioning process that implements changes to the physical network in order to support the line features requested by customers. Finally, this data, along with usage information generated from the operation centers, is consumed by a biller to setup customer. This literature review has demonstrated that this new data exchange approach is very promising. In the future, we would like to explore solutions to derive the best fragmentation for a system based on its internal indices and data structures [9].

C. Fan, W. Juang and M. Chen [10], conducted a research on efficient fair content exchange in cloud computing. They concluded that in cloud computing, the service providers can offer a virtualization environment and also provide applications as a service for user's demands. Users in a cloud are able to access services, such as e-book purchasing provided by the service providers (like Yahoo or Google). If a cloud user attempts to exchange her/his digital contents with another user located in the other cloud, no watermarking protocol or DRM mechanism can efficiently provide the watermark exchange function in this environment. In addition, users must guarantee the exchange transaction fairly. Since the Internet is not secure, there may be an attacker to impersonate the user to perform the digital content exchange. In order to solve the above problems, authors proposed an efficient and fair digital content exchange scheme in cloud computing. Their proposed scheme can provide an efficient way for users to do mutual authentication, ownership transfer, and fair transaction.

In recently years, cloud computing has become an interesting and popular research topic due to the mature of related technologies. In cloud computing, it provides a virtual computing environment (called cloud) and also supplies the resources pool for users demands. In clouds, consumers (or users) can purchase the service in the form of infrastructure-as a-service (IAAS), platform-as-a-service (PAAS), and software as- a-service (SAAS). IAAS concerns about the integration of databases and IT systems for consumers like IBM, HP, and so on. The IAAS layer also contains server systems, switches, routers, and other systems that handle specific problems from processing or data storing for users demands. PAAS is a higher abstraction layer. It focuses on the services to design, test, host, and maintain applications integrated in the platform. The PAAS layer also allows developers to design and maintain their specific operating systems and service platforms. SAAS is the highest layer that allows applications to be a service delivered to users.

In cloud computing, if a user attempts to exchange her/his digital content with the others in another cloud, she/he may need to perform the traditional watermarking protocols. However, these protocols are not suitable for the digital content exchange and cannot guarantee the transaction fair among users. In order to solve the above problems, author proposed novel efficient and fair digital content exchange scheme. Their scheme provided many nice properties such as, mutual authentication, watermark exchange, fair transaction, and low computation. In addition, their proposed scheme also can provide many nice properties, such as, fair exchange, ownership exchange, and low computation.

There were many researches which were conducted in the same discipline. But most of the researches carry the same problem for the exchange of digital content. Also, a user may face the situation that she/he has sent her/his digital content to the exchanging user, but do not obtain the corresponding digital content. None of the mentioned schemes can offer the solution of the unfair situation. In addition, since the Internet is not secure and any attacker may impersonate the others to exchange digital content, mutual authentication must be guaranteed between clouds and users.

C. Fan, W. Juang and M. Chen [10], proposed scheme satisfied the following properties including mutual authentication and key agreement, without WCA (Watermark Certificate Authority) participation, low computation cost, fair exchange, and optional usage for robust watermark or reversible watermark in cloud computing. In their proposed solution there are four phases including the setup phase, the authentication and key agreement phase, the content exchange phase, and the resolution phase. There are five participants in our scheme that are the public cloud C, AS1 (Authentication Server 1), AS2 (Authentication Server 2), Alice, and Bob. In setup phase, the system chooses a security parameter l and we assume that the KGC (Key Generation Centre) generates all key pairs for all parties. The authentication and key agreement phase determined the authentication phase of the services and set up the key exchange mechanism. In content exchange phase, if Alice and Bob have been authenticated by their authentication server successfully, then Alice performs the content exchange with Bob. In dispute resolution phase if Alice honestly forward her exchange digital content or Bob does not receive from Alice, then he forwards to AS2 for asking the dispute resolution. In mutual authentication phase Alice and AS1 choose their nonce's to authenticate each other. Bob and AS2, on the other side, authenticate each other by the same way. In the key agreement phase, we claim that an attacker cannot have the non-negligible probability to guess. In the dispute resolution phase, if Alice does not honestly send her exchange digital content to Bob or her digital content cannot be decrypted, then Bob can ask AS2 to resolve the dispute with the help of AS1

This literature review has proposed a secure and efficient fair content exchange scheme in cloud computing. In their proposed scheme, users can be authenticated by the authentication server of a cloud and the proposed scheme can provide the digital content exchange delivery efficiently and fairly [10].

H. Jumaa, P. Rubel, J. Payn [11], conducted a research on an XML based framework for automating data exchange in healthcare organizations. They provided a transparent and automatic communication between health information systems for the purpose of exchanging patients' data among healthcare professionals is deemed as one of the most challenging problems in eHealth. Indeed, data storage in health information systems is mainly performed in relational databases, whereas extensible Mark-up Language (XML) is seen as the de facto standard for exchanging data among health organizations. Automating data interchange between relational databases and XML documents remains however a challenge. In this paper, they proposed a general mediation framework to facilitate the storage of the new incoming data in XML format into the relational databases of the legacy health information systems and vice versa. The proposed mediation architecture is based on the XML technology and its related languages and derivatives (XML Schema, extensible Style sheet Language Transformations (XSLT)), which provide powerful tools for sharing, converting and exchanging information. The adopted methodology consisted in converting the database model into an XML schema and in performing an automatic, reliable and efficient mapping between the schemas representing the exchanged source and target data by means of the XSL T language. Their approach has the capacity to preserve the integrity constraints of the relational schema, which allows to check the XML info sets for anomalies or incoherencies before updating the relational database from the XML document. It also captures the hierarchy of the tables in the target database, which guarantees that the automatically generated Structured Query Language (SQL) queries will be correctly performed. Moreover, our mediator includes a rule base allowing a coherent and secure mapping between the exchanged data sources for ensuring the database integrity.

Data Interchange (ED!) solutions, using Health Level Seven (HL 7) for example, health institutions and organizations are increasingly adopting Web services and its extensible Mark-up Language (XML) based technologies to communicate and exchange data. The authors proposed an XML based mediation framework for automating the data exchange between any XML document and any existing relational database. Their proposed mediation framework is based on the development of generic components, which will ease the setup of specific interfaces adapted to any XML source and any target relational database. The mediator components are independent of any application domain, and need to be customized only once for each couple of source and target data storage format. The main role of this architecture is to automatically store the data embedded in XML documents into a target relational database as well as to retrieve the results of any query over the database and to embed them into XML documents (or messages) structured according to a requested interchange format.

This literature review has demonstrated an XML-based mediation framework to enable an automatic and transparent data exchange between any XML source and any relational database. The goal of their framework was to facilitate the storage of any new incoming data, more and more produced in XML format, into the relational databases of the legacy health information systems [11].

X. Zhang and D. Graˇcanin [12], conducted a research on service-oriented-architecture based framework for multi-user virtual environments. They came to the conclusion that Service-Oriented Architecture (SOA) is an application framework used for creating complex enterprise systems by integrating distributed services. The SOA standards are primarily focused on the service compos ability and data interoperability. Because of the featured capabilities of SOA, it is also used in distributed simulations. However, SOA has its limitations in terms of the performance of real-time message exchanging. In order to address the disadvantages and improve the application performance, they proposed a framework that combines the streaming technology and SOA. The proposed framework is used for constructing multi-user Virtual Environment (VE) applications by integrating the application content from distributed services. The additional streaming channels applied to SOA enable the services to actively propagate the real-time messages. The VE applications constructed using the framework have better performance. However, due to the distributed architecture of SOA and the heavy payload of message exchange in the framework, the application performance needs to be evaluated. They described the metrics used to evaluate the performance and present the evaluation results.

Applying Web Service technology to incorporating services as components in the applications has been explored in distributed simulations. The application components are distributed services that maintain their own states and generate messages corresponding to the invocations. Integrating services as application components is the feature of Service- Oriented-Architecture (SOA). The service integration and interoperation have been the major focuses for distributed simulations and other SOA applications. SOA has its advantage in terms of cross-platform integration and open standards for interoperation. However, the processing efficiency sets a big performance barrier for SOA because of its distributed architecture and the XML-based messaging. As the performance of SOA remains a big issue, applications that require real-time response might face performance challenges on our platform. In order to verify the real-time performance of our SOA, they designed an experiment to evaluate the interaction latency of the developed application. Based on the experiment results, we discuss the applications that can fit well on the framework.

X. Zhang and D. Graˇcanin [12], presented that incorporating streaming to compensate the service data interaction could take the advantage of both the SOA and real-time streaming technology. Therefore, they extend their framework on SOA for constructing DVE applications from services. As there is little effort in investigating the impact of the latency in SOA for real-time 3D applications, we introduce our SOA incorporated framework and evaluate the performance of the real-time interactions for the VE applications. The framework adapts both the SOA and EDA (Event-Driven-Architecture) to drive the service integration and interoperation. The seamless integration is addressed by introducing ontology to abstract the application domain knowledge. The streaming technology is applied to reduce the interaction latency and enables the services to actively push the updates.

III. Data Exchange Model of Patient in KSA

The proposed system is based on Service oriented, cloud computing architecture. The end users will be sending their national ID number to request for their previous medical records. This request will go to the National Information centre. The verification process will be done to verify the incoming user national ID number. There are number of medical records which are located in different parts of kingdom and they are operating on different platforms. The system architecture is composed of three layers as shown in Fig. 1:

1) Presentation Layer
2) Business Logic Layer
3) Middle-Ware Technology Layer

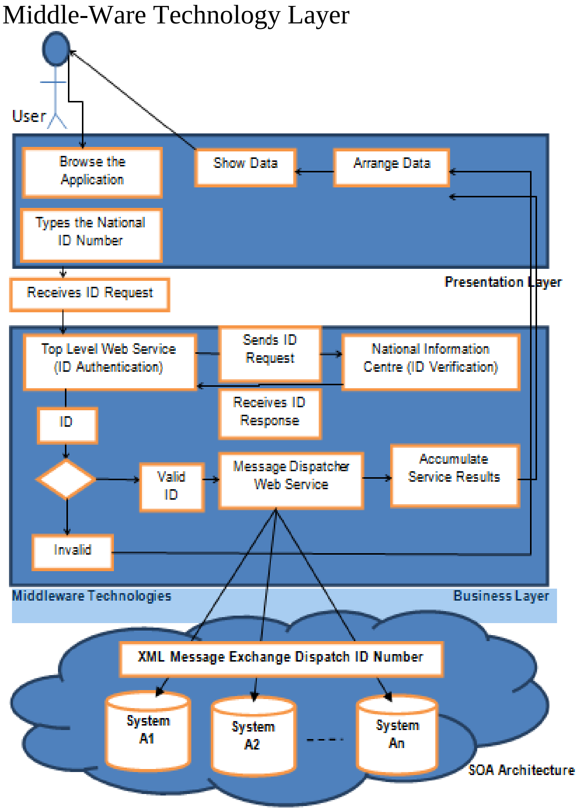

Fig. 1. System architecture

Presentation Layer: The end users will be using the web client to access the electronic patient record web portal. The end results of the patient inquiry will be displayed to the patient on the web browser.

Business Logic Layer: This layer is logically divided into multiple sub-layers. The layer receives the national ID number of the patient and will dispatch it to National Information centre to verify the patient identity. Once this request is approved the request will be dispatched to the middleware technology layer of business layer.

Middleware Technology Layer: This is responsible for proving the interoperable mechanism among this different platform. This is an abstraction layer which will hide the platform details of other systems and will use the general mechanism to utilize the services of the other system. SOA will be used here along with the XML based definition to define the message exchanging among these system. Broadly speaking there will be a generic web service which will be modeled to some XML document.

This XML document has defined the methods along with the return parameters. This XML document will be used as a generic language for the communication among this different system. This XML document will be having all the information regarding the services it required from other platform in the form of well-structured Unicode format. Potentially speaking there will be a service level agreement among these different platforms to utilize their platform resources to form one cloud computing architecture of service oriented architecture. These services are loosely coupled services which will not invoke each other rather there will be message exchanging among the top layer and these systems inside the cloud architecture to share their resources and exchange and retrieve data of the patient. The whole process will be done in the following way.

1) The users will send their National ID number to the electronic data exchange system.
2) The top level web service will dispatch the request to the national ID centre to verify the national ID number. The web service will determine this operation as a web application and return the result to the message-dispatcher layer. This layer is part of middleware business logic layer. If the national ID is valid national ID then the message-dispatcher layer will dispatch the request to the systems inside the cloud architecture.
3) Message-dispatcher layer has defined the web service for the patient records. This service is modeled to XML document. This XML document has defined the message exchanging mechanism among these different platforms. WSDL (web service definition language) has defined the web service XML document. This XML document has specified the location of the service and the operations (or methods) the service exposes. The data type, messages, port types and operations are well defined inside this document for the patient record. This web service will dispatch the national ID number to the other systems which are part of service level agreement term. This service level agreement will grant this web service to access the service of the requested system inside the cloud architecture. The transaction will be done in the following steps.

- The message dispatcher layer will dispatch the national ID to all the systems inside the cloud architecture to request for the patient record service.
- The message dispatcher layer has web service which will

exchange the messages to all the systems inside the cloud computing service oriented architecture.

- The response of this request will be a service which is requested from these systems inside the cloud architecture. Service is a component of the system which can be used to utilize the system resources of the platform to get the patient data.
- The result of these systems will be in the form of service. This result will be accumulated in the form of service and will be sent back to the message dispatcher layer.
- The message dispatcher layer will indentify the result of this web request which is potentially a service (a software component) which can be used to get the data from that particular system.
- The data will be accumulated by this web service and will be sent to the presentation layer to show it to the patient.
- The results will be shown on the client browser screen.
- It is assumed that medical records are identified by national ID number.

IV. Quantitative Evaluation, and Analysis

The sample survey was conducted in different public and private sector domains. The response was positive from different group of masses regarding the XML web services approach towards implementing data exchange systems. The sample question was distributed to 20 persons for the evaluation. Here we present some of the questioner questions, discuss, and graph the results of the survey. Table I, shows the survey results of the sample questions.

TABLE I: Survey Results of the Sample Questions

Question	Agree	Not agree	May Be	Don't Know
Q1	17	1	1	1
Q2	18	1	1	0
Q3	16	1	2	1
Q4	16	2	1	1
Q5	17	2	0	1
Q6	16	1	2	1
Q7	15	2	1	2
Q8	16	2	1	2
Q9	5	14	1	0
Q10	3	16	1	0
Q11	14	3	2	1
Q12	14	2	3	1
Q13	15	4	1	0
Q14	15	2	1	2
Q15	16	1	2	1
Q16	16	1	1	1
Q17	15	2	2	1
Q18	2	17	1	0
Q19	10	5	2	3
Q20	16	1	1	2

Q1: Do you think that XML web services approach prove useful in implementing data exchange among different system?

Q2: Do you think that SOA allow the services to successfully exchange messages among data exchange systems?

Q3: Do you think that cloud computing architecture along with the XML, SOA based architecture can help data exchange systems to utilize the services of another system as a result of services level agreement among the data exchange systems?

Q4: Do you believe that data exchange systems can help patients to successfully retrieve their medical profiles electronically?

Q5: Do you think that cloud computing along with the XML, SOA based architecture can save patient's time?

Q17: Do you believe that the cloud computing along with the XML, SOA based architecture has friendly defined user interface for the presentation of patient's medical record?

Q18: Do you believe that cloud based computing along with the XML, SOA based architecture has loop holes inside its design?

Q19: Do you think it's hard to implement service level agreement policy among data exchange systems?

Q20: Do you believe that cloud computing along with the XML, SOA based architecture is an ultimate electronic patient record service for patients to fully utilize their old medical history?

The sample questions were asked from the people of different domains. On the basis of above gathered raw facts, following graph is plotted. The 20 questions were answered by 20 people of different sectors. The details of the graph are shown below.

A. Discussion

85 percent of the people answered “Agree” for the XML based web services approach in implementing data exchange systems. 90 percent of the people answered “Agree” for the SOA to allow services to exchange messages. They believed that SOA can successfully allow exchanging of messages among data exchange systems. 80 percent of the people answered “Agree” for the services to utilize services of another system inside the SOA architecture, if there is a service level agreement. 80 percent of the people answered “Agree” for the system to retrieve patient’s medical records efficiently and effectively over the electronic communication channel. They believed that, this system can help patients to successfully retrieve their medical profiles efficiently and effectively over the electronic communication. 85 percent of the people answered “Agree” for the system to save patient’s time. 80 percent of the people answered “Agree” for the system to save patient’s medical cost. 75 percent of the people answered “Agree” for the system on helping patients to refrain themselves from going for the medical test again and again, as they can easily locate their medical profile electronically. 80 percent of the people answered “Agreed” on system helping medical organizations to better organize their medical procedures. 70 percent of the people answered “Not Agreed” for the system having the technical feasibility issues in its implementation. 80 percent of the people answered “Not Agree” for the system having the operational feasibility issues. 70 percent of the people answered “Agree” on the system following security policies and procedures. 70 percent of the people answered “Agree” on system following the confidential transmission for the patients’ medical records electronically. 75 percent of the people answered “Agree” on system resolving the data exchanging among different platforms. 75 percent of the people answered “Agree” on system having the abilities to grant access to both

rural and urban area people to utilize the system resources for their medical profiles. 80 percent of the people answered “Agree” on system guarantying maximum availability of data exchange service among different platforms. 80 percent of the people answered “Agree” on system following standard procedures, policies and protocols for data exchange among different platforms. 75 percent of the people answered “Agree” on system having the friendly defined user interface for the presentation of patient's medical record. 85 percent of the people answered “Not Agree” on system architecture having loop holes inside its design. 50 percent of the people answered “Not Agreed” on system having difficulties in implementing service level agreement policy among data exchange systems. 80 percent of the people answered “Agreed” for the system to be an ultimate solution for patients to fully utilize their old medical history

Fig. 3 illustrates the relationship between different responses taken from the survey results. Fig. 2. a, and Fig. 2. b have shown the minimum number and maximum number of result against the survey criteria.

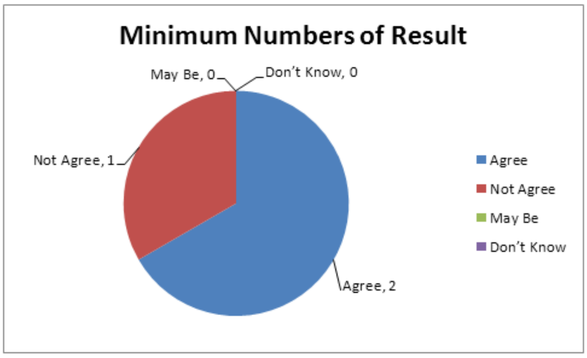

Fig. 2. a

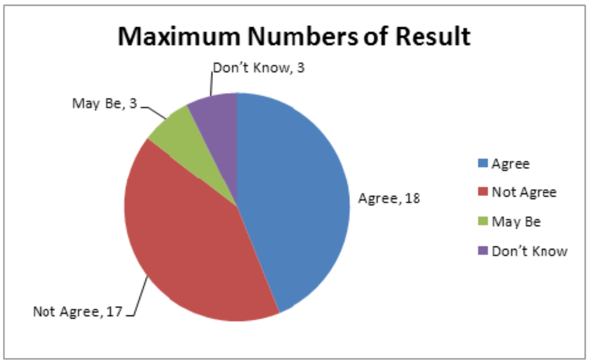

Fig. 2. b

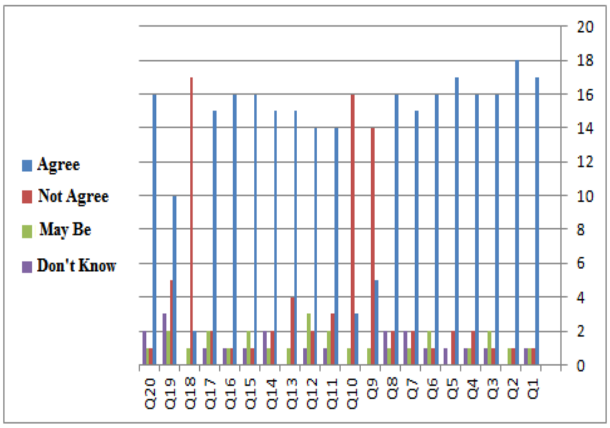

Fig. 3: Survey analysis

V. Limitations

- The research has indicated the following assumptions to be taken into consideration while developing or implementing the data exchange communication systems among different platforms.
- There should be a service level contract between the client and the service provider.
- It is assumed that patient records are stored with a national ID number inside the medical health care centers in order to utilize the patient medical record electronically.
- The system design should focus on writing standard well structured XML documents, and communication interfaces. These interfaces along with the message passing protocols should be defined in accordance to the service level agreement. There should be standard mechanism of communication among all of these different platforms to utilize each other service.

VI. Conclusion and Future Works

In this research study we have highlighted the methods, techniques, procedures, standards and policies by which data should be searched and retrieved efficiently from the different health information systems irrespective of the platform details. The aim of our contribution is the proposed system architecture and design which is based on cloud computing service oriented Architecture which utilized by the XML based web services approach. As a result of this study the model can be used to save the patient medical cost, time, and helps patient to get its medical record history from any location by simply using a web client.

To continue our work in the future, we will try to enhance the model, develop, and implement various solutions to specific problems we have identified based on our observations and the data we have collected from the system evaluation.

Acknowledgments

The authors acknowledge the full support received from Arab East Colleges for Graduate Studies, especially they very thankful to Professor Dr. Abd Allah Alfaisal, Dean of the Colleges, for his encouragement and support.

References

[1] D. G. Kernen, “Connecting for health. Global vision, local insight,” *Report for the World Summit on the Information Society*, World Health Organization Press, Geneva, 2005.

[2] A. Dogac, G.B. Laleci, S. Kirbas, Y. Kabak, S. S. Sinir, A. Yildiz, and Y. Gurcan, “Artemis: deploying semantically enriched web services in the healthcare domain,” *Information Systems Journal*, vol. 31, pp. 321-339, 2006.

[3] C. Oberdan Rolim, F. Luiz Koch, C. Becker Westphall, J. Werner, A. Fracalossi, and G. S. Salvador, “A Cloud Computing Solution for Patient’s Data Collection in Health Care Institutions,” *Second International Conference on eHealth, Telemedicine, and Social Medicine*, 2010.

[4] T. Takase and K. Tajima, “Efficient Web Services Message Exchange by SOAP Bundling Framework,” *11th IEEE International Enterprise Distributed Object Computing Conference*, 2007.

[5] L. Hu, P. Li, and Y. Wang, “The Design and Implementation of a SOA-based Data Exchange Middleware,” *2010 International Conference on Service Sciences*, 2010.

[6] S. Qian and Z. Fu, "Design and Implementation of the Data share and Data Exchange System Based on SOA," *Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference*, 2011.

[7] H. Xu, Y. Tian, G. Dong, and Y. Wang, "A Schema of Data Exchange for Heterogeneous Data," *Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), 2011, 2nd International Conference on*, 2011.

[8] Q. Guo, G. Chang, D. Sun, and X. Wang, "Shared Service Architecture for Emergency anagement System Development," *Genetic and Evolutionary Computing (ICGEC), 2010 Fourth International Conference*, 2010.

[9] A. Yahia, S. Kotidis, and Y. Res, "A Web-services architecture for efficient XML data exchange," *Data Engineering*, 2004.

[10] C. Fan, W. S. Juang, and M. Te Chen, "Efficient Fair Content Exchange in Cloud Computing," *Computer Symposium (ICS)*, 2010.

[11] H. Jumaa, P. Rubel, and J. Payn, "An XML-Based Framework for Automating Data Exchange in Healthcare," *e-Health Networking Applications and Services (Healthcom)*, 2010.

[12] X. Zhang and D. Graˇcanin, "Service-Oriented-Architecture based Framework for Multi-User Virtual Environments," *Simulation Conference*, 2008.

Hosam F. El-Sofany is Assistant Professor of Computer Science in Arab East Colleges for Graduate Studies, Department of Computer Science, Riyadh, Kingdom of Saudi Arabia. The e-mail address is hosam_elsofany@hotmail.com.

Khalid F. Al-Otaibi is a Student in MSc. Program in Arab East Colleges for Graduate Studies, Department of Computer Science, Riyadh, Kingdom of Saudi Arabia. The e-mail address is khalidfalotaibi@hotmail.com.

Majed S. Alsanea is a Student in MSc. Program in Arab East Colleges for Graduate Studies, Department of Computer Science, Riyadh, Kingdom of Saudi Arabia. The e-mail address is majed.alsanea@gmail.com.