

Service Oriented Architecture: A driving force for paperless healthcare system

Neha Thethi ¹⁺, Usha Batra ² and Saurabh Mukherjee ³

¹ Department of SE, ITM University, Gurgaon, India

² Assistant Professor, Department of CSE, ITM University, Gurgaon, India

³ Associate Professor, Department of CSE, Banasthali University, Banasthali, India

Abstract. India has a huge population which is geographically dispersed over a wide area, resulting in inaccessibility of quality services to all socio-economic levels particularly in rural areas. In this paper, we describe how this challenge can be overcome by integrating the distributed and heterogeneous healthcare systems using SOA technology which provides point-to-point EAI within and across the system through implementation of Web Services. The core services are made available by a layer of middleware – the ESB. The adaptation of the existing healthcare system into a single centralized unit will most significantly enable the rural population to avail medical facilities promptly, systematically and efficiently.

Keywords: service oriented architecture, enterprise application integration, enterprise service bus, web services, paperless healthcare system.

1. Introduction

India is a rapidly developing country with an immensely improving standard of living. In such a scenario, it is difficult for the Healthcare System (HS) to provide quality services to all socio-economic levels particularly the rural population. The Indian HS is a hybrid of private and public providers. The Government of India addresses health policies, regulatory matters and disease control. Some shortcomings of public HS are - the conceptualisation and planning of the all the programs is centralized and the specific needs of the local area are not taken into account, the infrastructure set up by the government is based on population norms rather than habitation and, there is a gap between the demand and supply of human resources. The states, on the other hand, address healthcare delivery, financing, and the training of personnel. In the urban and sub-urban areas private healthcare facilities are preferred whereas, the significant proportion of Indian population, still below the poverty line and residing in rural areas, avails the public healthcare facilities. Convergence of these two worlds presents a major challenge.

This hurdle faced by the healthcare organizations can be overcome by an enterprise-wide, information-sharing platform that meets their day-to-day business needs and also facilitates integration of information

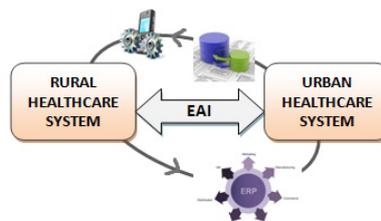


Fig. 1: EAI between healthcare systems.

⁺ Corresponding author. Tel.: + 91-8826520188.
E-mail address: neha.thethi@gmail.com.

and legacy systems. The platform needs to provide interoperability among providers, payers, pharmacies and other stakeholders.

The development of information sharing brings not only opportunities but also new challenges. To handle and manage the B2B application and integration within and across the enterprises is a challenging task. Normally, the information systems focus on only the data and business processing in their own domains. Due to lack of corresponding interface standards and norms, these information systems perform things in their own ways. As a result, information sharing and business integration among these information systems become unavailable and thus “information isolated islands” are formed. As these systems are not interconnected, information sharing becomes difficult for them. In other words, the applications provided by different software providers cannot interoperate with each other. However, this challenge becomes acceptable with the mere introduction of IT in implementing B2B application integration.

The paper is organized as follows. Section 2 explains the need of EAI and its benefits. Section 3 presents an overview of SOA and its implementation using Web Services and SOAP. In section 4, a glimpse of the current artifacts used by organizations for B2B application integration is provided. Section 5 presents the EAI Layers for a typical Healthcare System. Finally, conclusions are drawn in Section 6.

2. Enterprise Application Integration (EAI)

The success of any enterprise initiative relies heavily on the integration between internal and external information systems. Emerging business requirements put additional pressure on technical infrastructures, calling for operational agility and a better alignment between business and IT. The rising complexity and dynamism of the business environment call for more robust IT infrastructures to provide intelligent resource sharing and dynamic service creation and distribution. Under the weight of emerging needs and with organizations increasingly striving to streamline and improve their operations, the adoption of even more comprehensive information systems integration becomes an imperative, namely one that combines corporate activities that traditionally are completely disjoint to propose new services [1].

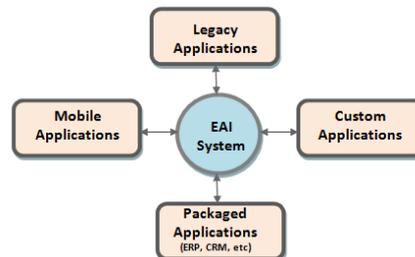


Fig. 2: EAI connecting heterogeneous applications.

EAI is a method and technology used to integrate the heterogeneous applications which are created with different methods and based on different platforms. By building an underlying structure, EAI connects the heterogeneous systems, applications, and data source across the entire enterprise together, and thus meet the enterprises’ requirements for seamless information sharing and data exchange among the ERP, CRM, SCM, database, data warehouse within the enterprise, and other important internal systems [2] as depicted in Fig. 2. With EAI, an enterprise can combine its core applications with new Internet solutions together.

3. Service Oriented Architecture (SOA)

Service-oriented architectures (SOA) is an approach that addresses the requirements of loosely coupled, standards based, and protocol independent distributed computing. Typically business operations running on SOA comprise a number of invocations of these different components, often in an event-driven or asynchronous fashion that reflects the underlying business process needs. The goal of SOA is to allow fairly large chunks of functionality to be strung together to form ad-hoc applications which are built almost entirely from existing software services. The larger the chunks, the fewer the interface points required to implement any given set of functionality.

In SOA, software resources are packaged as “services”, which are well defined, self-contained modules that provide standard business functionality and are independent of the state or context of other services. Services are described in a standard definition language, have a published interface, and communicate with each other requesting execution of their operations in order to collectively support a common business task or process. Services that utilize Web services standards [3].

3.1. Web Services (WS)

Web services provide a means to implement SOA by defining a standardized mechanism to describe, locate, and communicate with online applications [4]. Essentially, each application becomes an accessible Web service component that is described using open standards. The WS framework is divided into three areas — communication protocols, service descriptions, and service discovery. Fig. 3 illustrates the process flow of a Web Service. In this paper, we look at the specifications that are currently the most salient and stable in each area:

- The simple object access protocol (SOAP) which enables communication among Web services;
- The Web Services Description Language (WSDL), which provides a formal, computer-readable description of Web services; and,
- The Universal Description, Discovery, and Integration (UDDI) directory, which is a registry of Web services descriptions.

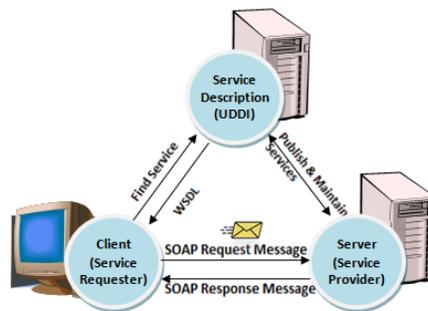


Fig. 3: Process flow of a Web Service.

3.2. Simple Object Access Protocol (SOAP)

SOAP is the protocol specification for message exchange among WS. It is based on the XML data model, and usually Relies on existing application layer protocols (e.g., HTTP,FTP,SMTP...) for message negotiation and transmission. SOAP is an XML-based protocol for messaging and remote procedure calls (RPCs). Rather than define a new transport protocol, SOAP works on existing transports, such as HTTP, SMTP, and MQSeries. At its core, a SOAP message has a very simple structure: an XML

```

SOAP Request Message
POST/healthcareservice
SOAPAction: "http://www.healthfirst.com/patientinfo"
Content-Type: text/xml; charset="utf-8"
Content-Length: nnnn
<SOAP:Envelope xmlns:SOAP=
"http://schemas.xmlsoap.org/soap/envelope/">
<SOAP:Body>
<m:GetPatientInfo
xmlns:m="http://www.healthfirst.com/patientinfo"
SOAP:encodingStyle=
"http://schemas.xmlsoap.org/soap/encoding/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi=
"http://www.w3.org/2001/XMLSchema-instance">
<patientID xsi:type="xsd:int">1002
</patientID>
</m:GetPatientInfo>
</SOAP:Body>
</SOAP:Envelope>
  
```

Fig. 4: A SOAP Request Message to retrieve a patient’s details.

element with two child elements, one of which contains the header and the other the body. The header contents and body elements are themselves arbitrary XML [5]. Fig. 4 shows an example of the SOAP request message which sends the ‘patient ID’ as an argument. The response message to this request will consist of the patient details corresponding to the ‘patient ID’.

4. Current Artifacts

XML in itself does not meet all of the sometimes-complex challenges associated with integration in ‘real world’ environments, but it does provide the basis for this integration. If it is accepted that XML is an alphabet rather than a language, it is clear that genuine integration will require additional technologies to orchestrate business processes, manage transformations within XML and validate and route XML-based messages throughout the organization.

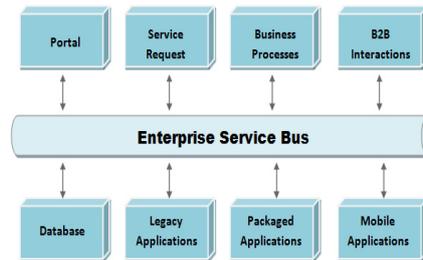


Fig. 5: Enterprise Service Bus

These features are at the heart of the Enterprise Service Bus (ESB). The use of the word “service” acts as a reminder that the ESB is often seen as a central (indeed the central) component of the SOA. ESB is regarded as “a layer of middleware through which a set of core (reusable) business services are made widely available” [6]. The SOA enables chunks of enterprise functionality to be presented as ‘services’ to the ESB, which routes, transforms and validates the XML inputs and outputs from these services. By developing in this way, once the Enterprise Service Bus is in place, following an initial point-to-point integration project, further projects merely involve ‘socketing’ new services onto this backbone or the re-use of existing services as depicted in Fig. 5.

5. EAI Layers for Healthcare System

A system is partitioned into layers in order to define the roles clearly and isolate the functions of each layer [7]. To avoid the overhead of a chain of modifications at a later stage, separating these concerns by dividing them into layers at the designing phase proves to be quite beneficial. An application can contain several distinct layers. A typical is a three-layered architecture which comprises of application, business

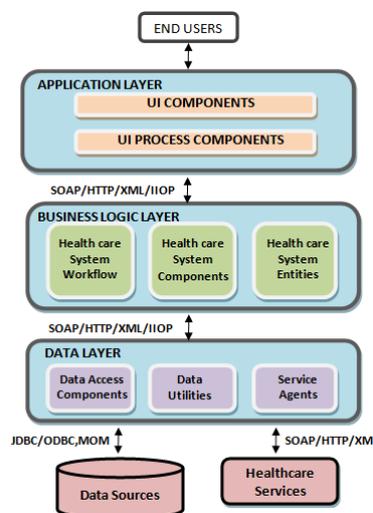


Fig. 6: EAI Layers for a typical Healthcare System.

logic, and data layers. Fig. 6 illustrates a generic model of the EAI layers that can be incorporated by any healthcare system.

- *The application layer* displays the User Interface (UI) and contains the user oriented functionality responsible for managing user interaction with the system. It generally consists of components that provide a common bridge into the core business logic encapsulated in the business layer.
- *The business layer* implements the business logic allowing us to centralize and reuse common business logic functions. It also helps to reduce resource contention and increase performance. Business logic is defined as any application logic that is concerned with the retrieval, processing, transformation, and management of application data; application of business rules and policies; and ensuring data consistency and validity.
- *The data layer* abstracts the logic necessary to access the database. Using a separate data layer makes the application easier to configure and maintain, and hides the details of the database from other layers of the application.

6. Conclusion

The SOA technology provides point-to-point EAI within and across the HS and allows us to integrate the rural and urban healthcare sectors using Web Services. The core healthcare services are made widely available through a layer of middleware – the ESB. The primary purpose of this integration is to reach out to the interior parts of the nation that require medical attention as soon as possible. The adaptation of the existing Indian HS into a single centralized unit will most significantly enable the rural population to avail medical facilities promptly, systematically and efficiently.

This will prove to be valuable not only for the civil population but also for the medical employees as they will have every patient's medical record at their disposal thus eliminating the burden of recording it individually. Also, they can analyse the patterns evaluated from medical histories of patients which will save time, effort and will give accurate results during diagnosis and prescription of medication.

7. References

- [1] C. Zhigang, C. Huiping: Research on Enterprise Application Integration Categories and Strategies. *International Forum on Computer Science Technology and Applications*, 2009.
- [2] Z. Wu, Y. Li. Research on Enterprise Application Integration Based on Web. *International Conference on Mechatronic Science, Electric Engineering and Computer*, 2011.
- [3] M. Papazoglou, W. Heuvel. Service Oriented Architectures: Approaches, Technologies and Research issues. *The VLDB Journal*, 2007, pp. 389-415.
- [4] F. Curbera, M. Duftler, R. Khalaf, W. Nagy, N. Mukhi, and S. Weerawarana. Unraveling the Web Services Web: An Introduction to SOAP, WSDL, and UDDI. *Internet Computing, IEEE In Internet Computing, IEEE*, 2002, pp. 86-93.
- [5] J. Tekli, E. Damiani, R. Chbeir, and G. Gianini. SOAP Processing Performance and Enhancement. *IEEE Transactions On Service Computing*, 2011.
- [6] Understanding the ESB. Retrieved from: www.polarlake.com/files/esb.pdf
- [7] N. Marina, A. Ghada. Generic Service Patterns for Web Enabled Public Healthcare Systems. *7th International Conference on Next Generation Web Services Practices*, 2011.
- [8] U. Batra, S. Mukharjee. EAI (Middleware): Integrating Stovepipe Applications of Varied Enterprises in Distributed Middleware with Service Oriented Architecture. *Electronics Computer Technology (ICECT), 2011 3rd International Conference*, 2011.
- [9] G. Tor-Morten, B. Bendik. A Successful Implementation of Service Oriented Architecture. *26th International Conference on Advanced Information Networking and Applications Workshops*, 2012.