# HL7 V3 Laboratory Messaging using SOA Infrastructure

By
**Mehtab Alam Khurshid**
**2007-NUST-MS-PhD IT-13**

Supervisor
**Dr. Hafiz Farooq Ahmad**
**Department of Computing**

A thesis submitted in partial fulfillment of the requirements for the degree of Masters of Science in Information Technology (MS IT)

In
School of Electrical Engineering and Computer Science,
National University of Sciences and Technology (NUST),
Islamabad, Pakistan.

(April 2011)

# Abstract


The need of interoperability in healthcare information systems has given rise to the development of various healthcare standards. However, most of these standards focus on the interoperability requirements of a particular healthcare department.Health Level 7 (HL7) is a singular standard as it focuses on the interface requirements of the entire health organization, thus facilitating interoperability and integration across multiple organizations with heterogeneous interfaces. HL7 V3 communication infrastructure aka Messaging Infrastructure (MI) enables interoperable communication using well-defined interactions, but is unable to; provide flexible communication across heterogeneous healthcare environments, capture rapidly changing healthcare business needs and maintain business sessions. In this research work, we aim to provide a flexible approach for integrating healthcare information systems using Service Oriented Architecture (SOA).

To cope with the deficiencies of HL7 V3 existing MI, we have proposed architecture and a methodology to perform the challenging task of integrating SOA to HL7 V3 models (without changing the legacy infrastructure). The architecture is realized for HL7 V3 laboratory domain and patient administration domain, by creating the composition of HL7 V3 standard services viz. Entity Identification Service (EIS), Result Query Service (RQS) and Order Placer/Fulfiller Service (OPFS). The services are deployed on multiple point-of-cares, and can be invoked locally as well as externally.

This research work highlights the weaknesses of HL7 V3 MI and key strengths of our proposed system. The proposed framework has been tested for EIS, RQS and OPF services on the span of two remotely residing point-of-cares and the results showed that the workflows generated by these services show 65 - 70% conformance with real healthcare business scenarios. The web service technologies such as SOAP, WSDL and XML along with SOA Enterprise Service Bus (ESB) provide enhanced interoperability in terms of data and platform, and increased communication (in and out data flow). The proposed methodology is also extendable to other healthcare care domains.

# Certificate of Originality

I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person nor material which to a substantial extent has been accepted for the award of any degree or diploma at SEECS or at any other educational institute, except where due acknowledgement has been made in the thesis. Any contribution made to the research by others, with whom I have worked at SEECS or elsewhere, is explicitly acknowledged in the thesis.

I also declare that the intellectual content of this thesis is the product of my own work, except for the assistance from others in the project's design and conception or in style, presentation and linguistics which has been acknowledged.

Author Name: **Mehtab Alam Khurshid**

Signature: ____________________

# Acknowledgments

All Gratitude to Almighty Allah, Who enabled me to undertake and accomplish this hectic challenge, in sound health. His blessings always transformed hardships into feasibilities. I offer my humble thanks to the Holy Prophet Muhammad (Peace Be upon Him), who showed the enlightening path and inculcated the marvelous knowledge to mankind.

I would like to express the deepest gratitude to my parents, who continually and convincingly conveyed a spirit of motivation to me throughout my research. I am heartily thankful to my supervisor, Dr Hafiz Farooq Ahmad, whose deep perception demonstrated to me the global concern and modern trends, which provided zeal to my work. I also extend my heartily thanks to my team leads whose guidance and support from the initial to the final level enabled me to develop an understanding of the subject. Thanks to my committee members for their kind cooperation in discussions and proof-reading. It is also a pleasure to thank those friends and colleagues whose persistent help and morale-building enabled me to achieve this dissertation. I am also thankful to NUST for providing me merit scholarship due to which I carried out my research activities with good mind set. Lastly, I offer my regards and blessings to all of those who supported me in any respect during the completion of the project. May Allah bless all of us! Amin .

**Mehtab Alam Khurshid**

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction and Motivation

## 1.1 Introduction

Since long time, paper based record keeping has been in usage for healthcare. As the technology progressed, the concepts of digital storage started its origination. In healthcare domain, a physician named Lawrence L. Weed first described the concept of computerized electronic medical records in 1960 [7]. According to Weed's proposal, there was a need of automated and organized patient medical data thereby improving the patient care and healthcare management.

In 1970s and 1980s, the academic institutes carried out research and development in creating electronic medical record systems. "The Technicon system was hospital-based, and Harvard's COSTAR system had records for ambulatory care. The HELP system and Duke's 'The Medical Record' are examples of early in-patient care systems. Indiana's Regenstrief record was one of the earliest combined in-patient and outpatient systems" [7]. However the electronic systems were still not in practical usage of hospitals, as the doctors and hence hospitals were still reluctant to use any system other than paper-based.

In 1990s, the medical computer applications started to become in practice, but they were highly complex. By the end of 1990s, 14 percent of US hospitals and clinics deployed electronic health record-keeping systems [25].

With the start of 21st century, the implementation practices of electronic healthcare systems get started in real environment. The worth of such systems has been realized globally and the international healthcare community began to get inclined towards electronic healthcare systems. The electronic systems were developed with a vision of storing healthcare data in an organized and co-related manner, but the technology advancements also made it possible to manage and process data efficiently.

Information and Communication Technology (ICT) has been in place from a long time, but now it has come to its revolutionary execution. Like other service industries, healthcare is also getting integrated with IT. Healthcare information and management cannot give real benefits until they are shared and interpreted in a meaningful manner. The role of ICT can be quoted by the following statement; "As Information Technology increasingly penetrates the healthcare industry, physicians and patients are experiencing the benefits of on-demand access to medical information where, when and how it is needed.", stated Ramez Shehadi, Booz Allen Principal leading the IT Strategy Practice in the Middle East [23]

To access information from anywhere and anytime, healthcare systems need to communicate with each other. The data- which is communicated- also needs to be used in a meaningful manner. A common language is required to enable healthcare systems to share information efficiently and effectively, and to establish a common information management system.

## 1.2 Healthcare Interoperability and Integration

Previously, healthcare was considered as patient's sickness and doctor's treatment. With the passage of time, this concept has been changed to many other requirements too. But the systems fulfilling these requirements are still very few.

Healthcare passed through mainly three phases; the manual systems i.e. paper based information storage, machine-based transfer of data (start-up of IT involvement) and machine-based data management (more involvement of IT) [35]. The second phase i.e. machine based transfer of data has actually paved path for the real IT involvement in the long run. Organizations started using Personal Computers, scanners and fax machines etc to transfer medical documents and other information. The third phase was meaningful sharing of information through the development of interfaces, called Level 3 interfaces. Those interfaces were responsible of interpreting the incoming information according to organization's local system. There is another phase; a fourth phase which is in process of development and involves standard based structured information sharing that can be interpreted by both parties (sharing information between each other). Such systems need strong basis of information and automation.

A large proportion of hospitals and healthcare organizations are still using traditional approach i.e. the paper-based systems. Such systems keep healthcare information in a dispersed form (in bits and pieces) without any correlation between the available data. This issue is enhanced when a patient moves to some other hospital (other than the local one) and is asked for bringing his/her medical history or conduct tests from which he/she has already passed through previously. The major demerits are high costs, difficult and painful process for patients, time delays and medical errors.

There is a need of such system which can introduce a simpler better technology for the overall healthcare sector with less costs, reliability and real-time transfer of information among organizations. If interface development between healthcare organizations is taken as solution, even then it is difficult to manage and from an expert opinion, it costs $50,000 per interface for hospitals, labs, radiology centers, pharmacies, and public health departments [35]. Healthcare standards are a way to establish a network of information management systems among organizations. Health Level 7 (HL7) is a healthcare communication standard which enforces total enterprise healthcare integration, through the use of one interface between all. It enables the entire ecosystem of healthcare data exchange interoperable with less costs, quick implementation and easy management. Founded in 1987, HL7 is a standards developing organization which produces interoperability standards for hospital information systems [22]

With progressive changes, HL7 has made the healthcare organizations free from developing customized interfaces and lots of costs on their management. It encompasses interface requirements of multiple healthcare domains. Being an interoperable standard, HL7 facilitates healthcare providers and patients to access required medical data from anywhere and anytime.

## 1.3 Motivation

The main challenge of healthcare is to share data among heterogeneous entities that are disseminated far apart. Healthcare information from multiple sources needs to be integrated and retrieved efficiently. Various systems (proprietary, vendor dependent) are in place to bring interoperability and integration but utilizing them on a broader level is a major issue. These systems are not scalable and extendable enough to accommodate changes. Moreover healthcare community needs such solutions that should be cost-effective and flexible.

Approaching a standard is rationalized solution but to select the appropriate one is vital. HL7 V3 is an ANSI standard; claims to provide interoperability through healthcare interfaces. SOA4HL7 is another metaphor which asserts flexible integration and business workflows communication across healthcare environment through the use of services. To achieve a framework that combines business and IT solutions at one place without effecting underlying infrastructures and getting maximum benefits of both (SOA and HL7 V3) is a success mark for healthcare.

## 1.4 Main Contributions

In this research we aim to provide a SOA framework for standard-based healthcare systems in order to enable cost-effective and flexible communication across the healthcare organizational boundaries. The proposed system brings business orientated solutions in healthcare IT intensive systems. Research objectives set to follow above stated problem statement are given below:

- To design a new communication framework for healthcare HL7 based systems in order to overcome the limitations of existing communication infrastructure.
- To provide detailed analysis of healthcare real case workflows that makes complex procedure understandable enough, to be implemented and deployed in real environment.
- To compare the performance of traditional approach (i.e. Messaging Infrastructure of HL7 V3) and SOA communication infrastructure in order to provide better understanding of their key strengths and weaknesses.
- To investigate the flexibility and scalability trends of SOA based healthcare systems.
- To document the resolving techniques of challenging exceptions that are faced while dealing with Business Process Execution Language (BPEL).
- To design and document the SOA based healthcare solutions which will provide knowledge foundations to the researchers and healthcare professionals, and enable them to extend the work on same lines.

## 1.5 Thesis Organization

This thesis is organized as follow:

- Chapter 2 provides pre-requisite knowledge required to understand; the existing issues in healthcare, the evolution of various standards and the comparative analysis of existing standards with HL7 standards. It also describes the various aspects of HL7 standard.
- Chapter 3 gives the detailed analysis of HL7 V3 traditional communication approach and evaluates it against real environment communication needs.
- Chapter 4 elaborates the benefits of SOA in healthcare environment and proposes a solution to the limitations of existing HL7 communication infrastructure. This chapter also justifies the need of SOA through various critical requirements.
- Chapter 5 presents the analysis and design initiatives of healthcare business workflows. It also derives the analogies of healthcare environment processes and artifacts with standard artifacts (SOA and HL7 artifacts).
- Chapter 6 presents the proposed methodology and architecture with detailed description of the components' design. The communication workflows designed for the services (present in architecture) are designed in BPEL and described as a step by step process.
- Chapter 7 again gives a detailed description of the implementation of the system. It includes HL7 V3 specifications and Java implementation tools. The capability of each service is discussed from implementation perspectives.
- Chapter 8 concludes with the summary of points made and describes the directions that have been decided upon for future research with justification and references to a set of work packages that would be required to complete the research.

# Chapter 2

# Background Overview of Healthcare

This chapter discusses the evolution of various healthcare standards and their capabilities in resolving issues of healthcare systems. The discussion eventually results on the comparison of standards and preference of HL7 standard over the previous one.

## 2.1 Introduction

With the technology progression, each and every field has undergone rapid growth. The only industry that showed a gradual progress in implementing newer technologies is healthcare industry. In most of the developing countries (and several developed countries), the health related professionals and doctors still move from one place to another for acquiring real time information on each patient.

Existence of multiple standards slowed down healthcare improvements, as it became difficult to select the appropriate one that fits in almost all situations. Data exchange among doctors and hospitals through the use of computers and networking has become critical but healthcare professionals and patients usually do not feel comfortable with computers, even though they can realize the essential impact of accurate record keeping and communication on improved healthcare system. The general communication flow in a healthcare environment is discussed as below:

- Patient comes to the hospital and usually after waiting for long hours, consults a physician.
- Physician suggests a prescription on paper and gives it to the patient.

- The patient takes the prescription to the pharmacist, and waits for him to fill the prescription.

This process is slow and unreliable which can be improved by bringing electronic communication between the patient and physicians, and most preferably between physician and the pharmacist. One option is to provide voice-enabled human computer interfaces for the physicians, nurses, pharmacists and other healthcare professionals [21] ,but it can also have accuracy issues.

According to Carmen Catizone of the National Association of Boards of Pharmacy, there are as many as 7,000 deaths from incorrect prescriptions in the United States each year [21]. These numbers show that a quality healthcare system having access to the right information at the right time has become vital to healthcare. The challenges posed are due to the large-scale distributed, complex and heterogeneous nature of healthcare systems.

Inside a healthcare organization, various healthcare processes need coordination and automation [33]. But the dynamic changes in healthcare environment make the integration difficult among various departments [36]. On the other hand, if we see external aspects of a healthcare organization then there also exists a need of sharing medical information among organizations. Healthcare standards are required to bring about a uniform way of carrying out medical transactions and healthcare executions.

HL7 standard claims to provide true interoperability, integration and automation requirements of overall healthcare environments. It also provides common vocabulary for multiple healthcare domains' communications.

## 2.2 Evolutionary Phases of Healthcare Standards

From 1950 to 1990, US healthcare sectors have gone through various progressions viz. telephones, punched cards, large mainframe and time-sharing computers, lower-cost minicomputers, local area networks (LANs) with mini- and microcomputers (to link local databases), online patient monitoring processes and internet self-care and telemedicine [4]. Not a single development was covering overall healthcare sector.

By the advent of information technology concepts and use of internet, healthcare industry moved towards the development of healthcare standards. The standards helped remote patients and healthcare community to get quality healthcare services with less delay and minimized costs. Following are some eminent healthcare standards that played an important role in healthcare evolution.

- NCPDP SCRIPT (1977)
- ASC X12 (1979)
- DICOM (1993)
- LOINC (1994)
- IEEE/CEN/ISO 11073 (2002)
- HL7 (1994, 2005)
- **NCPDP SCRIPT (1977)**

  National Council for Prescription Drug Program is a not-for-profit ANSI-Accredited drug standard [28]. It encompasses overall pharmacy sector services. The twelve work groups involved in this standard developing body ensure standard transmission of pharmacy data.
- **ASC X12 (1979)**

  National Council for Prescription Drug Program is a not-for-profit ANSI-Accredited drug standard [2]. It encompasses overall pharmacy sector services. The twelve work groups involved in this standard developing body ensure standard transmission of pharmacy data.
- **DICOM (1993)**

  "Digital Imaging and Communications in Medicine (DICOM) is a standard for handling, storing, printing, and transmitting information in medical imaging" [6]. DICOM is popular for addressing interoperable transmission of medical images, by specifying file format and protocols. It also integrates various medical devices which confirm DICOM standard.
- **LOINC (1994)**

  Logical Observation Identifiers Names and Codes (LOINC) is a database which contains universal code system [24]. This system facilitates electronic reporting of laboratory and clinical observations.
- **ISO/IEEE 11073 (2002)**

  "ISO/IEEE 11073 is an interoperable Medical / Health Device Communication Standard(s) and is developed by the joint effort of ISO, IEEE, and CEN " [5]. ISO/IEEE 11073 standard family defines parts of a system, with which it is possible, to exchange and evaluate vital signs data between different medical devices, as well as remote control these devices.

- **HL7 (1994, 2005)**

  HL7 is a health informatics standard that allows communication and integration of health care information systems and applications permitting the sharing of data around the globe or across the street [10].

  Compared to other standards (such as NCPDP and DICOM), HL7 aims to provide interoperability in overall healthcare organization through interface communication, rather than focusing a particular aspect. Moreover HL7 specifies standard communication artifacts and not just the standard vocabulary codes.

  ASC X12 in combination with HL7 becomes capable to contribute to healthcare interoperability; in a way that X12 carries out inter-enterprise, administrative communications, while HL7 handles external research and public health communication.

  DICOM also needs HL7 to provide base layers to "Integrating the Healthcare Enterprise" (IHE) in order to develop ultimate piece of interoperable medical imaging.

## 2.3 Role of HL7 in Healthcare

HL7 is the leading community standard in healthcare for data exchange and interoperability. Interoperability in healthcare enables heterogeneous health information systems to exchange data accurately [9] and to use that data effectively [27] within and across the organizational boundaries [29].

HL7 is a not-for-profit organization and has brought more than 55 countries (around 4,000 worldwide members, including healthcare vendors, providers, payors, government agencies, consultants and others) [14] and 34 affiliates under its umbrella. This shows the widespread popularity and universal effectiveness of HL7 standards.

HL7 presents a comprehensive framework, which is based upon standard specifications for exchange, integration and delivery of clinical and administrative data [30]. A large proportion of health community uses HL7 messaging standards with the purpose of exchanging information among heterogeneous systems. Apart from that, HL7 also develops specifications for

Electronic Health Records (EHR) systems and Service Oriented Architecture (SOA).

HL7 is available in two main versions [8], which are based upon different contexts. The HL7 V2.x represents a whole series of version 2, and HL7 V3 is a single but the most comprehensive version with capabilities of strong information basis and context understandings. Being the pioneer version and an ISO standard, HL7 V2.x messaging standard is the most used healthcare standard around the globe. In U.S, the HL7 V2.x is widely used (more than 90 percent) in most of healthcare organizations [14], while HL7 V3 is being used by U.S. government agencies (Food and Drug Administration, Department of Veterans Affairs). Version 3 is also getting wide coverage in Canada, UK, Netherlands, Germany, Mexico and Asian countries [14].

Although the focus area of HL7 standard is to exchange clinical and administrative data among healthcare organizations, yet it facilitates integration of different healthcare modules (patient administration, finances, pharmacy, specimens etc).

HL7 V2.x message is structured upon multiple segments, which in turn contain data fields. Segments specify message type and trigger events, while data types are included in data fields [34].

HL7 V3 message is structured on XML language and has basically three layers, which specify message headers and payloads. V3 message also includes message type and trigger events. V3 as opposed to V2.x is built upon strong information models i.e. Reference Information Model (RIM) and the derived ones [34]. The HL7 V3 is aimed to develop consistency among different information objects which are used in healthcare messages, in order to streamline requirement-gathering (in conformance of standard) and implementation process. An important aspect of V3 standard is that, its standard modeling approach makes the information structure flexible enough to incorporate decision-support mechanisms, electronic patient records and service oriented architecture structures.

SOA enables syntax-independent communication similar to HL7 V3 messaging standard. Both domains have their own strong communication infrastructures; however HL7 V3 messaging lacks to provide flexible communication across various healthcare units.

# Chapter 3

# Review of State-of-the Art in HL7 V3 Domain

HL7 V3 provides specifications for data communication. The data is basically the actual medical contents, which are generated by using HL7 specifications. In this chapter we are going to discuss HL7 V3 communication infrastructure which includes protocols, interactions, actors (responsible for exchanging messages) and control events.

## 3.1 HL7 V3 Communication Infrastructure

### 3.1.1 Process Artifacts

Before going into details of communication infrastructure, it is important to understand V3 artifacts. HL7 V3 contains several process artifacts (concepts) [16], which specify healthcare domain knowledge and procedures. These artifacts include; *Storyboards*, *Application Role*, *Message Type*, *Interaction*, *Trigger Event*, *Hierarchical Message Descriptor* (HMD), *Reference Information Model* (RIM), *Domain Information Model* (D-MIM) and *Refined Message Information Model* (R-MIM) The basic concepts of these artifacts are narrated as follows:

- **Storyboard:** It represents a particular story in the form of sequence of events, performed by the application roles (In general, users).
- **Application Role:** It represents system component responsible for sending and/or receiving interactions.
- **Trigger Events:** These are a set of conditions that initiate an interaction between application roles.

- **RIM:** RIM is an abstract model containing set of UML classes, attributes and their associations. It is the key information structure behind development of V3 standard.
- **D-MIM:** It is a subset of the Reference Information Model (RIM), and is used to create messages for any particular domain.
- **RMIM:** It is a subset of D-MIM and is required to compose the set of messages derived from the HMDs.
- **HMD:** It is a tabular representation of R-MIM's sequenced elements.
- **Message Types:** It defines the message payload.
- **Interaction:** it is a one-way transfer of information, associated with a specific message type, particular trigger event and application roles.

### 3.1.2 Communication Infrastructure

HL7 V3 communication infrastructure puts forth the following aspects which are used by all V3 messaging implementations:

- HL7 V3 message creation and interpretation rules, called messaging infrastructure.
- A protocol for message transmission.
- Generic "Communication Roles" that support the modes of HL7 messaging [15].

#### HL7 V3 Messaging Infrastructure (MI)

**A. Message Composition**

V3 Message is composed of three parts:

- **Transmission Wrapper:** It acts as message header and includes information of packaging and routing V3 Composite Message to the destination (designated receiver).
- **Control Act Wrapper:** It conveys information regarding logical operations which are required to be carried out at receiving side healthcare application, for instance, the request response interactions.
- **Payload:** Actual data contents which are responsible for carrying out required interactions.

The MI also involves HL7 V3 information models (for the creation of HL7 V3 composite message), messages and interactions.

**B. Protocols for HL7 V3 Data Transfer**

V3 messages are transmitted by using certain protocols. HL7 V3 specifies lower layer protocols for ensuring reliable delivery of messages. In addition to that, it also uses HTTP, SOAP and business communication protocols.

**C. HL7 V3 Message Control**

HL7 V3 addresses message control by HL7 V3 domain contents, identification of sending and receiving entities (for an interaction), relationship between interactions and participating roles.

## 3.2 Limitations of HL7 V3 MI

The communication in HL7 V3 is currently taking place through MI. The creation of message involves HL7 V3 Reference Information Model (RIM) [3] and V3 interaction specifications. HL7 V3 message acquires a composite structure before being transmitted, i.e. it is wrapped in two wrappers viz. Control Act Wrapper and Transmission Control Wrapper. The composite message is then transmitted using MLLP [18], ebXML [17] or web services [26] protocols (the specifications of ebXML and web services are in-process of being defined in HL7) [29].

Since the crux of interoperability lies in the meaningful exchange of data, the implementation of HL7 V3 is primarily based on a messaging system that utilizes the industry standard XML format for data exchange. V3 messages enable interoperable communication using well-defined interactions initiated by trigger events, but it cannot provide set of related interactions and also the triggering order of those interactions. On one hand, HL7 V3 provides specification for message contents and underlying transport that indicates that it covers whole stack of intercommunication. But on the other hand, it does not provide specification that can create message session according to a business case.

HL7 V3 MI though works successfully in carrying out transmission processes but is not capable to capture all functional requirements. The consumer entity- while conducting an interaction- needs to (more or less) know implementation details of messaging. This minimizes the enforcements of transparency and interoperability. Moreover, the healthcare realm is in desperate need of making healthcare processes simple while not compromising on generating required full-fledged business sessions.

SOA is a solution to provide interoperable communication, providing transparency and global access to the required information anytime anywhere. Table 3.1 highlights the issues and preferences required in healthcare communications. By merging HL7 V3 and SOA, we can get maximum benefits in healthcare realm. SOA enables HL7 V3 to provide standard based healthcare services that will have well-defined granularity and coupling in a distributed environment.

| **Messaging Infrastructure** | **Service Oriented Architecture** |
|---|---|
| Deals with data. | Deals with both data and functionality. |
| Deals with a particular invocation of interaction. | Deals with overall workflow of related interactions in a business workflow. |
| Incorporating any new aspect needs changes in infrastructure. | Adaptable and scalable for the incorporation of any new aspect without any change to overall infrastructure. |
| Focuses mainly on message structure, its contents and transmission. | Supports service creation and hosting in addition to message structure, contents and its transmission [32]. |
| A message has no interface, thus no abstraction. | A service has interface(s) which exposes its behavior and thus high level abstraction is achieved. |
| Less cost-effective. | Cost-effective [13] |
| Focus is often on data movement or replication rather than functional reuse [12]. | Services in SOA are functionally reusable. |
| Provides specifications of message contents and transmission. | Provides considerations from service design level to overall infrastructure development level. |
| Message focus (Information Viewpoint) for development and governance [20]. | Focuses on Contract (Service, Operation and Behavior - Computational Viewpoint) [20]. |

Table 3.1: Messaging Infrastructure vs. SOA Framework

## 3.3 SOA4HL7

Healthcare Service Specification Project (HSSP) is a joint effort by HL7 and OMG, in order to provide baseline for service based healthcare environment [19]. HSSP provides service interface specifications in the form of Service Functional Models (SFMs), but doesn't provide implementation considerations. So designing the services using HSSP guidelines and implementing them in an own way, gives flexibility in the realization of healthcare SOA. The vision of HSPP is to make healthcare free from multiple vendors and give "all-in-one" solution (in terms of easy integration and flexible communication) that fits to overall healthcare environment.

In the next chapter, we will discuss benefits of SOA in healthcare and how HSSP is playing its role in this area.

# Chapter 4

# SOA in Healthcare

As a testament to our understanding of the healthcare SOA and to deliver complete solutions, the real healthcare environment is analyzed. This chapter explains the healthcare needs with respect to SOA solutions.

## 4.1 Impact of SOA in Healthcare

Healthcare is a complex industry, making thousands of transactions per day. For accomplishing these transactions, communication within and across the organizational boundary is required. For such communications, integration with other systems is required which is not an easy task. Integrate-able communications need the development of new interface everytime which costs a lot.

SOA provides such a framework that gives an integrated environment which causes interoperable communication. Using standards like HL7, the interoperability becomes enhanced. Healthcare environment needs availability and reusability of services, to all healthcare users (Consumers and providers). SOA framework doesn't change the overall existing infrastructure of an organization; rather it formats the functionality in the form of services' collection, which is available and reusable. SOA in healthcare also provides a means to achieve:

- efficient information management,
- effective accessibility,
- flexible and interoperable communication.

One major impact of SOA in that it interprets healthcare complex processes in well-defined business language rather than technology-specific terminologies and grammar.

SOA is a not a new concepts for the community and healthcare stakeholders and professionals know its importance, yet they are reluctant as they have put lots of investments in building legacy systems. These legacy systems contain the healthcare business logic and behavior. Moreover the healthcare has its own specific policies and constraints for various domains, like accounting and billing, pharmacy, clinical procedures and information retrieval etc. There is a need to analyze the critical needs, viabilities and adaptability requirements and accommodate them in standardized and reusable SOA services. The SOA framework should be in conformance with healthcare strategies; its services should have clear granularities, transparency (separation of the roles of consumers and producers) [12] and implementation of software engineering principles i.e. coupling and cohesion.

## 4.2 SOA Services

SOA consist of components and components can be services. A service constitutes mainly two parts; interface and implementation. The interface is the point which is used for interactions with external modules in a particular environment, while the implementation sets the working logic behind that interface.

Existing system processes can give improved functionality if packaged in services. A service is a unit of work [31] but SOA service is a unit of business task. It is usually such kind of service which is repeatedly being used by multiple parties (applications, services etc). Generally it makes up a part of organized workflow called “choreography”. If taking an insight of SOA service itself, the unit of work it performs can be further decomposed in small units, which define the granularity of this service. It is not always required to use all functions of service at one time.

SOA is an architectural style which involves the concept of services for all, and business relationships between services’ providers and consumers. Using Web Service technology as an underlying infrastructure, it is required to generate HL7 V3 business transactions through SOA web services. Web services based SOA makes the application development swift and adaptable. It is suitable for healthcare environment where changes occur frequently.

## 4.3 Health Business SOA Standards

HSSP [19](by HL7 and OMG) is intended to produce standard services that should have well-defined functionalities, behavior, and interfaces. The main objective is to develop standardized healthcare systems for all, with no dependency on vendors and proprietors.

When developing services, it is important to specify granularities. The HSSP community felt that coarse-grained interface specification could best support the implementation and deployment needs; allowing the users to make implementation decisions about the granularity of the service instance.

In this research work, the granularities of services are specified according to HSSP guidelines in order to align them with standard SOA for healthcare. The granularities are of two types;

- coarse-grained
- fine-grained

Table 4.1 defines the granularity of SOA services according to HSSP guidelines [12].

One of the services, specified by HSSP is Entity Identification Service (EIS) [11] which is designed and implemented in this research work. the next section further elaborates functionality of EIS and other proposes services.

| **Coarse-grained** | **Fine-grained** |
|---|---|
| Deals with data. | Deals with both data and functionality. |
| Coarse-grained services are mostly used for local as well as intra enterprise communication among businesses applications. | Fine-grained services are suitable for intra-enterprise communication where components communicate with each other, with faster network. |
| Less communication | Frequent communications |
| Coarse-grained services are generic and embed legacy applications in them. Services are reusable. | Fine Grained services are most often associated with particular functionality (so they are not much "generic" as coarse-grained services). Services are less reusable. |

Table 4.1: Granularity of Services; Coarse-grained vs. Fine-grained

## 4.4 HL7 V3 based SOA Services

The provision of flexible communication structure with the help of SOA-enabled HL7 V3 standard based services eases;

- the process of healthcare systems and application integration, and
- enable users to take advantage of efficient healthcare system.

SOA is the most suitable framework for achieving an interoperable, integrate-able and flexible HL7 V3 based communication.

In this research, HL7 V3 based SOA framework is proposed which mainly focuses on two healthcare domains (also HL7 V3 specified domains) viz. laboratory domain and patient administration domain.

These proposed services exhibit business process behaviors while being invoked. Following is the description of proposed services in HL7 V3 domains.

### 4.4.1 Patient Administration Domain Service

#### Entity Identification Service (EIS)

EIS is an HSSP specified service which is responsible for retrieving identification information of entities (e.g. patient). It basically focuses on patient's demographics which are usually required in every healthcare environment.

EIS in this research work is designed with certain guidelines of HSSP, and is implemented by composing several services which communicate with one another. The purpose of EIS is to search patient's profile information locally and (if record not found) then on other system (healthcare unit) which can reside remotely. In case it is not found on other system also, then a new patient demographic profile is created. The EIS in turn can communicate with other lab domain services, based on the requirements.

### 4.4.2 Laboratory Domain Service

#### Result Query Service (RQS)

The Objective of Result Query Service is to retrieve test results of a particular patient (which are stored for him/her over time). RQS sends a query to a particular point of care, for accessing patient's test history or derive his/her immunity towards some particular treatment. If there exists no such record, the result query response is generated empty or the query is sent back to the requesting party.

#### Order Placer/Fulfiller Service (OPFS)

In a real healthcare environment, a patient comes to a collection point and asks for some particular tests. His/her sample(s) is taken and sent to some test center for performing tests on it. Associated with that, the HL7 message is sent to the test center for providing knowledge of the required manipulation. The test center then performs the test on sample(s) and generates results which are transformed in HL7 message format and sent back to the requesting party. OPFS is aimed to provide another real case business workflow.

OPF is basically designed to cater multiple business goals such as;

- Order Confirm - Promise Confirm Response
- Order/Promise Reject
- Order/Promise Replace
- Order/Promise Nullify
- Order/Promise Revision
- Order fulfillment Request-Promise fulfillment

# Chapter 5

# Analysis of Healthcare Workflows and HL7 Specifications

In order to approach the goal of aligning HL7 V3 with SOA infrastructure, it is important to analyze the real case requirements of healthcare environment. This chapter explains the analogy mapping between real healthcare workflows and HL7 V3 standard specifications.

## 5.1 Healthcare Requirements Elicitations

The real clinical healthcare environment has usually a specific setup, consisting of mainly two entities viz. the point of cares:

- Small Clinical Labs : the Collection Points(CPs)
- Large Clinics and Hospitals : the Test Centers(TCs)

In this research work, a couple of renowned clinical systems are analyzed to extract their requirements. Usually the clinical labs have their laboratory information systems deployed at each branch to store and process the medical data but when it is required to share that data with other labs/hospitals, these information systems lack the capability of integration. Specifically, it brings great problem when patients' test orders are required to be sent from a collection point (where test orders are taken) to some test center (where these test orders are actually fulfilled). Though several mechanisms exist for placing test order to some remote test center, such as:

- Mailing/ emailing

- Fax
- Or electronic information systems having some sending mechanism

But these ways cannot generate flexible and interoperable communication, anywhere anytime. Moreover, the data exchange doesn't prove to be meaningful, as the interpretation at the receiver side might not always be the same which is expected at the sender side.

HL7 V3 based information system can develop such interfaces for the point-of-cares, that can fulfill interoperability requirements. The system can help them in sending and receiving the test reports in an efficient and cost effective way, and also connect them with other collections points and test centers. However, the real workflow management is not possible with HL7 V3 interfaces alone. It also cannot maintain communication sessions according to business requirements. It is inevitable for healthcare environment to manage business workflows through HL7 V3 based information systems. SOA services are capable of aligning real world complex healthcare procedures into integrate-able, flexible and adaptable workflows.

## 5.2 Healthcare Workflows

Taking renowned clinical systems in consideration, the real environment healthcare work procedures (the procedures at collection points and test centers) are captured. The extracted work flows are illustrated in Figure 5.1 and Figure 5.2 and are discussed in the next section.

## 5.3 Mapping Analogies of Healthcare Procedures and HL7 V3

The key requirements are; mapping real life requirements to HL7 V3 specifications and SOA concepts. Such kind of similarities helps in bringing integration and enhanced interoperability factors. Alignments of SOA and HL7 V3 concepts are shown in Table 5.1 [1].

Figure 1 shows a flow of a particular real case. A patient comes to a collection point and asks for the test (or a medical information). Based on some identifier (for instance Patient_ID), his demographics are searched locally, and if not found then query is sent to external system (another collection point or test center). If it doesn't exist there too, then a new patient profile is created for him/her, including all his required demographics. This real case can be mapped with EIS procedure. The flowchart entities in figure

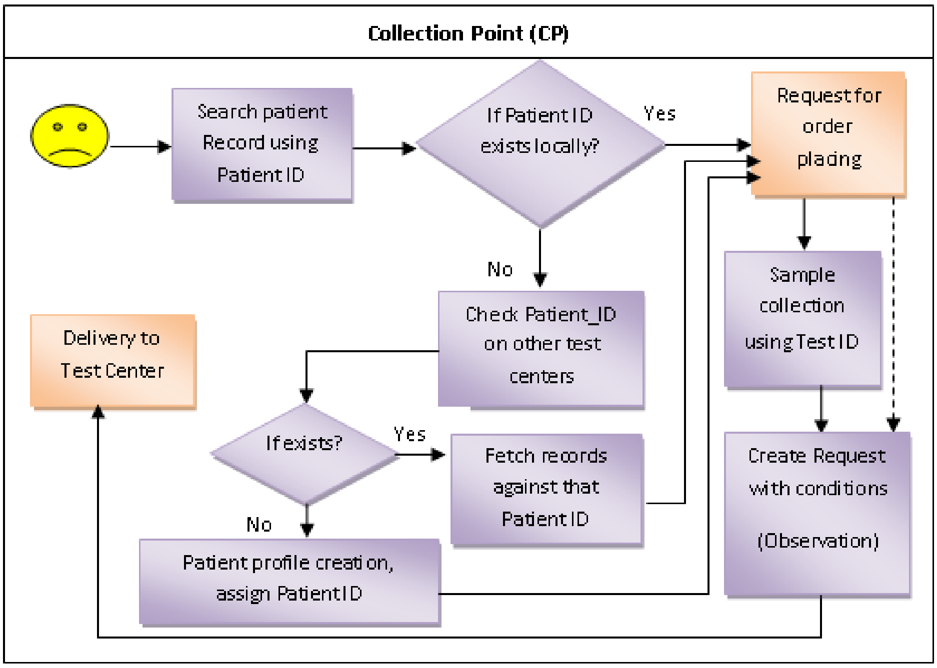



Figure 5.1: Activities related to Patient Test at a Collection Point

5.1 show the activities related to one business goal i.e. "Identifying a patient based upon his/her ID".

Further processing is captured in figure 5.2, when test order of a patient is forwarded to a test center along with specimen collected and information regarding those specimen(s) (Observations/instructions). If the consulted test center is capable of performing the test, it processes and generates results otherwise the test order is sent to some other capable site (test center). Results are verified by the doctor and sent back to collection point.

The entities in figure 5.1 and figure 5.2 are related to place a test order and get the results of that order, while the overall procedure is also generally used for querying demographics and test results of a particular patient. The functional entities (of illustrated workflows) can make up loosely coupled services of SOA, with well-defined granularities. While the alignments of SOA workflows and HL7 V3 specifications are carried out on the basis of conceptual mappings of real environment workflows and interactions.

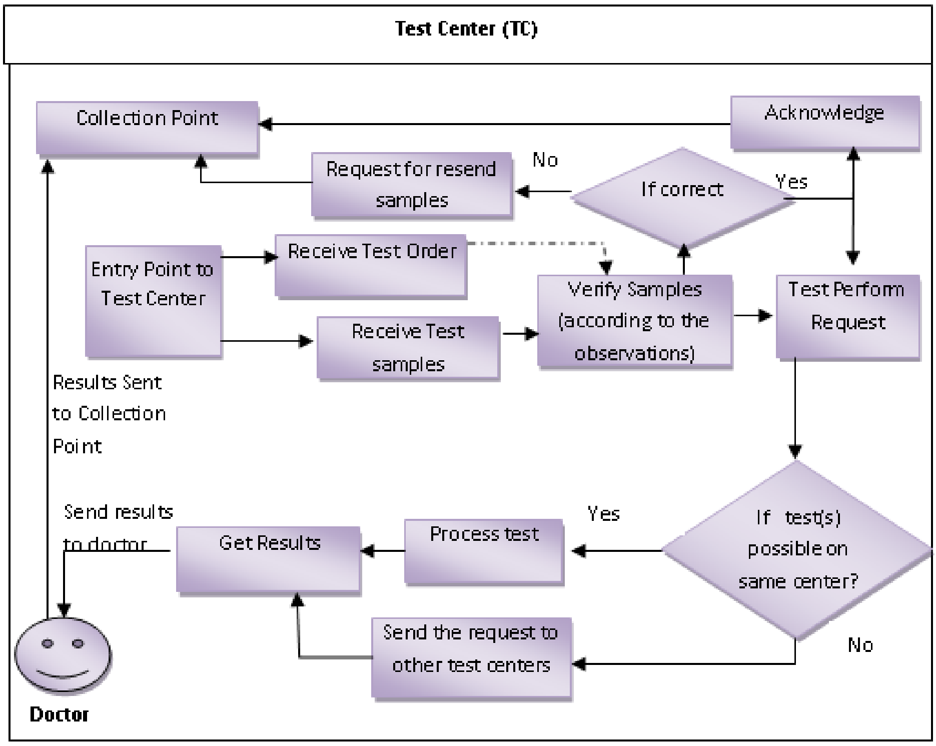



Figure 5.2: Activities related to Patient Test at a Test Center

| **Items** | **SOA** | **HL7 V3** |
|---|---|---|
| Business Aspects: | | |
| Business Scenarios | Business Use Case | Storyboard |
| Business Roles | Business Roles | Storyboard Actors |
| Analysis: | | |
| Analysis Information Model | Refined Domain model | Domain Information Model / Domain Message Information Model |
| Interaction | Service Invocation / Usage | Interaction |
| Actor | Service | Application Role (s) |
| Interface | Service Interface | Application Role (s) |
| Action | Operation | Interaction |
| Structure and Implementation: | | |
| Policy | Contractual / Declarative (e.g. WS-policy framework) | Transmission Wrapper (per message) |
| Message | Message | Message |
| Message Header | Message Header (e.g. SOAP Header) | Transmission Wrapper |
| Interface | Service Interface | Message Source/Destination |

Table 5.1: SIMILARITIES OF SOA and HL7 V3 CONCEPTS

# Chapter 6

# System Design and Architecture

This chapter explores the design of proposed system architecture and describes approach of structuring and organizing components to achieve that architecture. The components are the composite services (web services' composition) which presents the business goals as well as users' (patients, physicians, and stakeholders) needs.

## 6.1 Process Model and Methodology

The implementation details of the major components of the system are discussed as follows:

### 6.1.1 Process Model:

Before going into details of methodology, the process model of the proposed work is illustrated in order to impart basics for aligning HL7 V3 and SOA domains. The steps revolve around anatomy and analogies of the two; IT intensive and business oriented realms:

**Steps for creating Process Model:**

- Gather and Assess:
  - Knowledge base of HL7 V3 standard
  - Knowledge of SOA
- Analyze and Process:

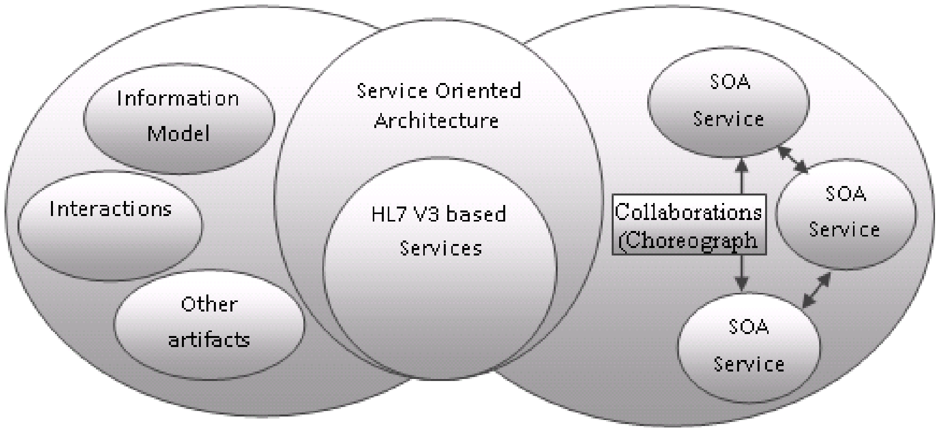



Figure 6.1: Process Model of HL7 V3 and SOA Integration

- Analogies of HL7 V3 and SOA
- Mapping the analogies

To create a roadmap strategy towards designing and developing SOA for HL7 V3, it is required to build up clear knowledge foundations of SOA infrastructure as well as HL7 V3 standard. Both have vast infrastructure and lots of structures and specifications; it is also required to identify business goals and narrow down the span. Further, the scaled knowledge is then mapped to develop the proposed system.

Figure 6.1 shows the process model, based upon the conceptual mapping of HL7 V3 messaging infrastructure and SOA specifications. The detailed methodology is explained in next section.

### 6.1.2 Methodology:

The methodology for designing and developing the proposed system, i.e. HL7 V3 messaging using SOA infrastructure is scaled around two healthcare domains;

- Patient Administration domain
- Laboratory domain

The specifications for both of the (above mentioned) domains are provided by HL7 V3 standard. This research work is carried out for both domains, so the devised methodology covers overall objectives and goals.

The methodology for designing and implementing the proposed system is as follows:

- Analyze the requirements of real healthcare environment.
- Analyze and find V3 laboratory domain artifacts, which can fulfill real requirements.
- Identify services by:
  - Application roles which can accomplish real requirements services are made accordingly.
  - Considering the main software attributes (cohesion,coupling and granularity).
- Decide on operations/interactions by:
  - Real environment workflows and HL7 V3 Interactions.
- Decide on transmitting contents (payload) by:
  - Refined Reference Information Model (RMIM) and Message types.
- Design and Implement services by using HL7 V3 RMIM and Java SIG API.
- Design and implement business workflows by Business Process Execution Language (BPEL)

## 6.2 System Architecture

The architecture shows the structure and organization of components and services. This organization visualizes a consistent and effective approach towards information flow. The proposed architecture also introduces the design of involved services in subsequent sections.

#### Architecture Description

Figure 6.2 shows the proposed architecture at abstract level. The architecture exposes the following features, discussed briefly:

- Distributed environment
- Services deployed on web servers at each point-of-care, to minimize load.

- Each PoC site also contains its local database.
- Each PoC can act as Service Consumer as well as Service Provider.

For the current real environment setup, the test center is main entity that is supposed to handle test orders processing.

The key components of the architecture are services, which are required to be carefully designed in order to instill IT and business approach in them. Each service is the outcome of composition of multiple services with one particular business goal.

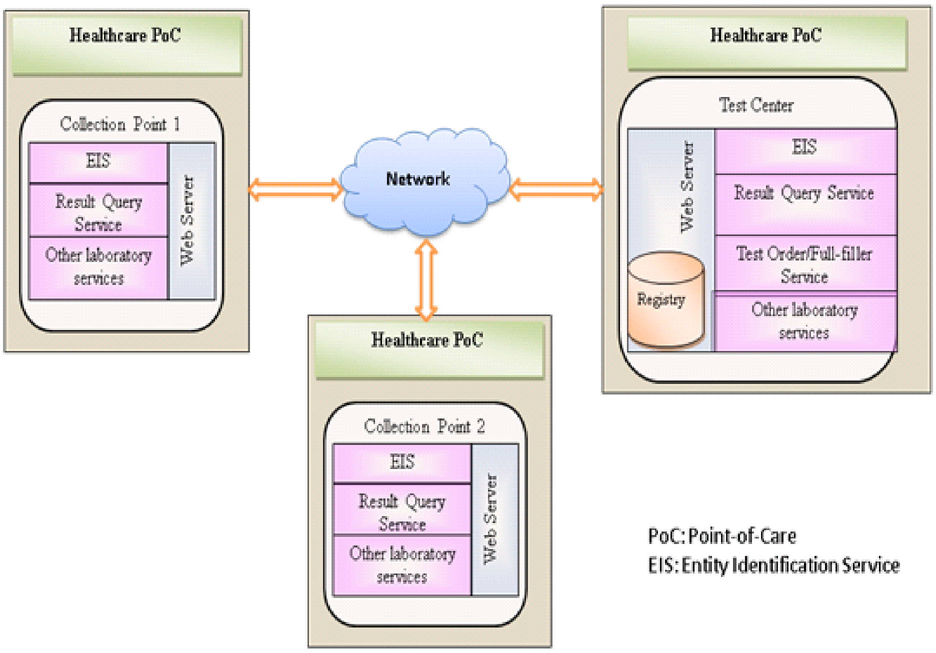



Figure 6.2: Abstract view of Proposed Architecture

### 6.2.1 Services

The composition of web services make up a workflow in which information flows with the intension of accomplishing particular objective. The design of architecture services with respect to HL7 V3 specifications, SOA tools and required pre-requisites are described as follows:

#### Design of Services' Workflows

Design of the system includes the following:

- Activity diagram,
- BPEL diagram

**A. Workflow of Entity Identification Service (EIS)**

EIS is composed of several services which communicate with one another in order to identify a patient locally or remotely. The action states of EIS services and their functionalities are depited in figure 6.3. The dynamic behavior shown through UML activity diagram helps in traversing complex business rules through step-by-step activities. Figure 6.4 shows the orchestration in EIS which is designed and implemented using BPEL. The workflow consists of several BPEL constructs, which together with web services make an overall composition. The web services and/or WSDLs act as application roles and are known as partner links in BPEL workflow. The specifications for this service are derived from HL7 V3 patient administration domain and SOA.

The steps of communication session along with BPEL constructs are described as follows:

**EIS_EntryPoint Service** represents entity identification service Partner Link, from which the request for patient identification is sent to *Receive* construct.

The Receive construct invokes **Parser Service** through *Invoke* construct. This parser service is designed to parse HL7 message on the receiver side. The receiver extracts required information from the message and processes it against the request.

The **Database Service** is designed to take patient identifier (patient ID) from the parsed message and checks patient's record against that identifier, in local database. *If sequence* captures the conditional flow, i.e. if patient information is found locally, then the retrieved patient information has to be sent to **Message Generator Service**, to format it according to HL7 V3.

The generated response message is sent to receiver i.e. **EIS_EntryPoint Service**, through *Reply* construct.

If patient record doesn't exist locally then External EIS (XEIS) (entry point) service is invoked in order to check patient's information on other point-of-care (collection point or test service). The XEIS represent another BPEL flow deployed on another remote center; receiving request for patient's demographics against his/her ID. If record is found at that remote site then the patient record is sent back to EIS_EntryPoint Service. Otherwise the same request message or response message with null data is sent back (to EIS_EntryPoint Service) and a new profile for patient is created and saved

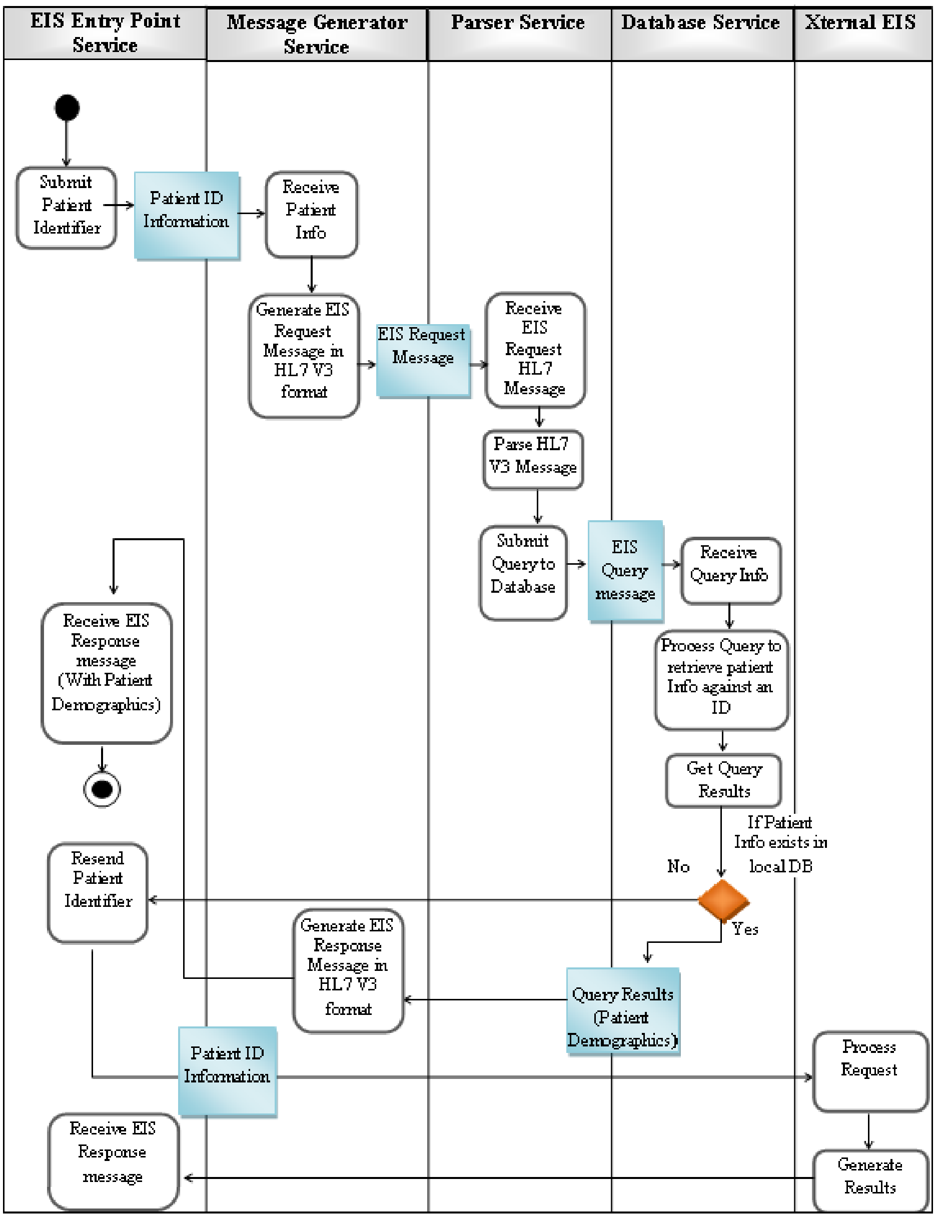



Figure 6.3: EIS Activity Diagram

in database, using database service.

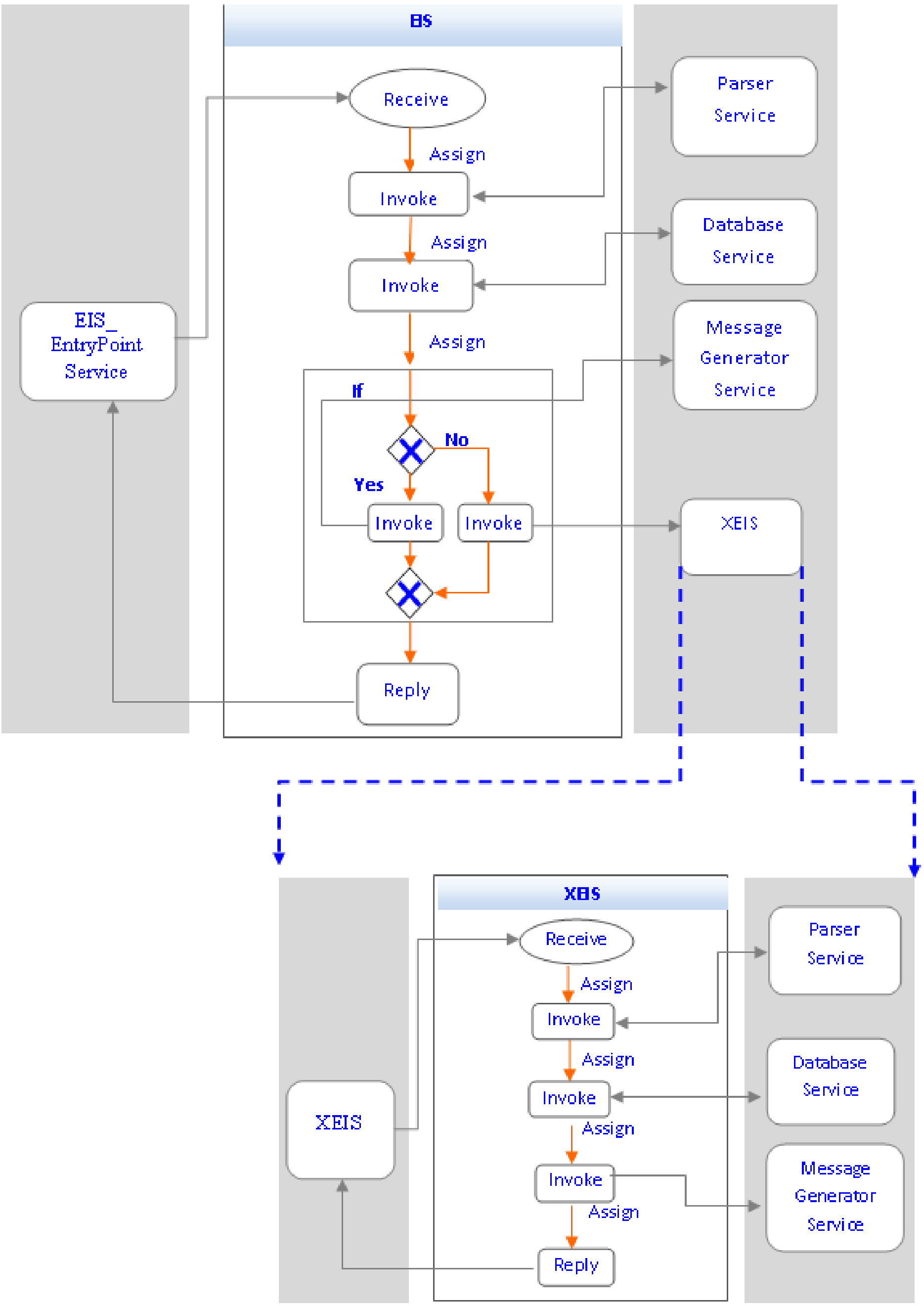



Figure 6.4: EIS Workflow, designed in BPEL

**B. Workflow of Result Query Service (RQS)**

The workflow of RQS is similar to EIS, with the exception of particular business goal. Moreover the objective of RQS drives it to follow the communication flow as illustrated in figure 6.5. Figure 6.6 illustrates the BPEL workflow of RQS. The specifications for composing this service are derived from HL7 V3 laboratory domain and SOA.

A real healthcare environment usually needs to inquire about patient results against the medical test(s), which he/she has undergone previously. To fulfill this requirement, RQS follows the subsequent steps:

**RQ_EntryPoint Service** represents the sender side partner Link, from which the request for patient test results is sent to *Receive* construct.

The Receive construct invokes **Parser Service** through *Invoke* construct. This service extracts meaningful information from the received message.

Based upon the acquired information, the query is processed on local database using **Database Service**. The role of database service is different from the one in EIS; it searches multiple records and gives a combined and comprehensive result at the end. The results are populated with values if the data exists against that patient tests otherwise the request is sent to some other remote point-of-care.

The HL7 response message is generated by **Message Generator Service**, with null values if no recorded test results exist in external database also. The generated response message is sent to receiver i.e. **RQ_EntryPoint Service**, through *Reply* construct, in XRQS and then in RQS workflows respectively.

**C. Order Placer/Fulfiller Service (OPFS):**

OPFS is a kind of service that generates asynchronous communication. When the test order is placed at receiver site, the test is either queued or executed. At completion, the results are generated and sent back to fulfiller through a call back mechanism. The specifications for OPFS are extracted from HL7 V3 laboratory domain and SOA. Figure 6.7 shows the action states of OPF service.

The "order" activities are mostly conducted by order placer, while the "promise" activities are accomplished by fulfiller. From design perspective, an application role sends a request to receiver side. A placer at receiver side places the request order by delegating the request to some service which has business logic of handling that request. After the test order has been performed, the results are generated and sent back to sender side. The sender is basically a fulfiller side which takes the results and shows them to

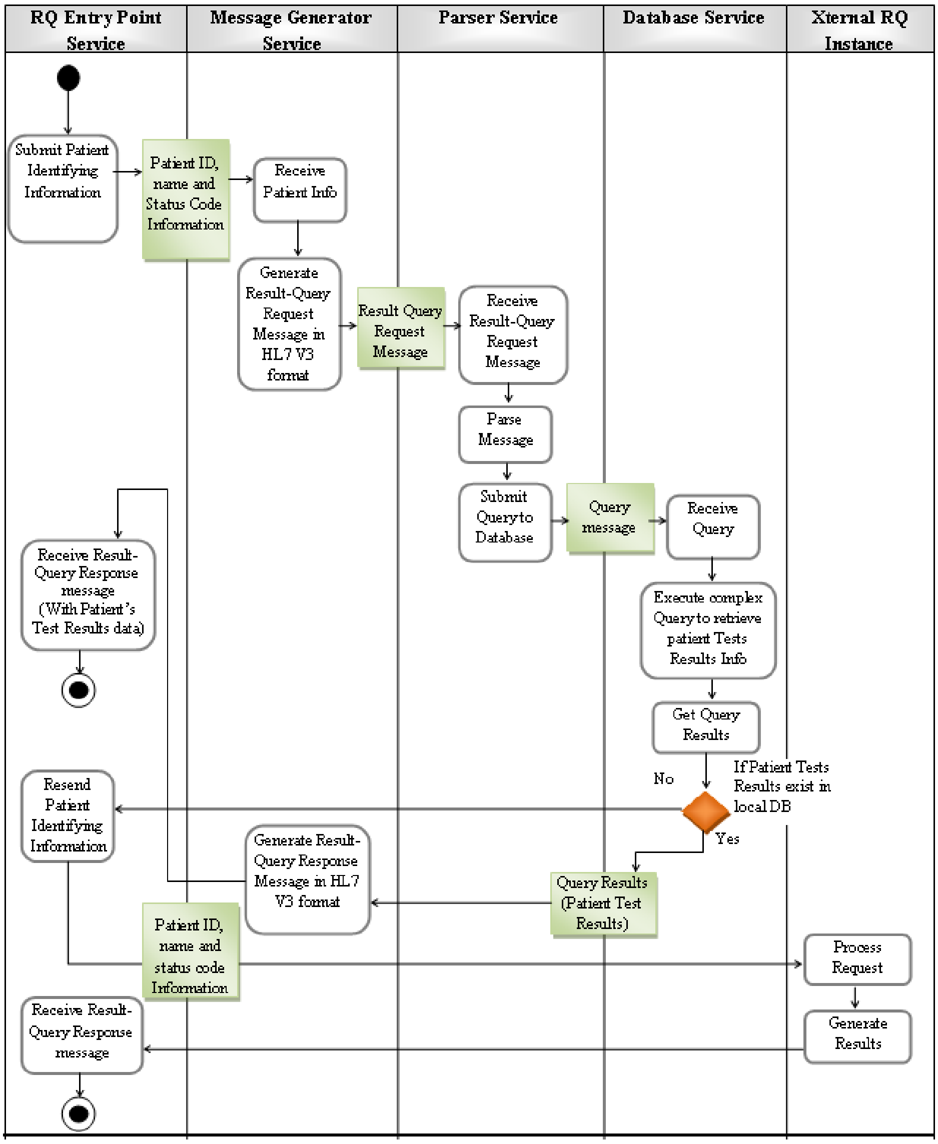



Figure 6.5: RQS Activity Diagram

user.

The workflow of OPFS is illustrated in figure 6.8. **OPF Request Sender Service** represents the sender side partner Link, from which the test order message is sent to **Placer Service**. which in turn invokes its workflow at receiver side. At receiver side the **Parser Service** is invoked and message gets parsed. The parsed information is sent to **Process Handler Service** through *Invoke* activity. Based upon the information received, the Process Handler Service manipulates the request and sends the results back to *Invoke*

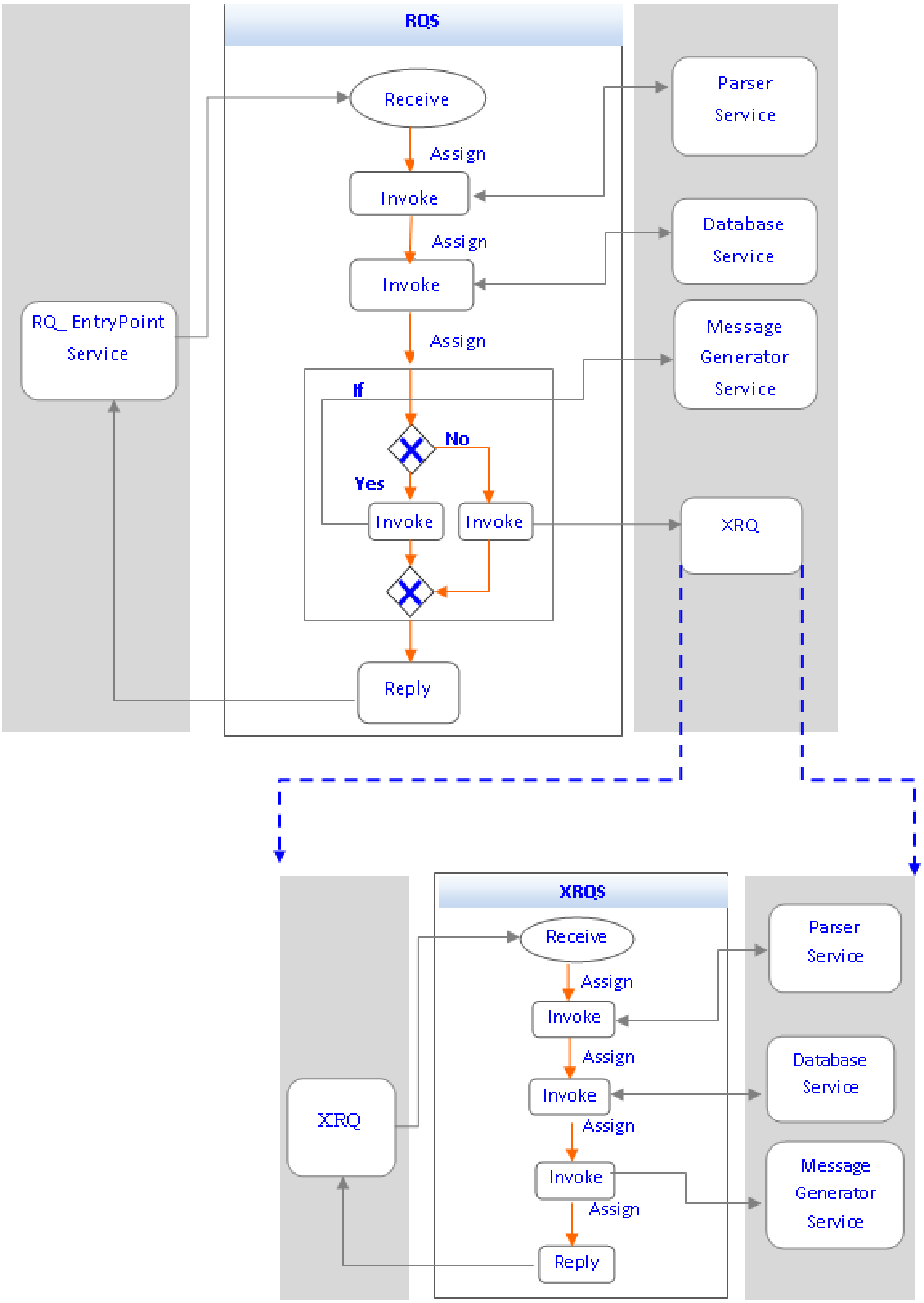



Figure 6.6: RQS Workflow, designed in BPEL

activity. These results are then assigned to **Message Generator Service** and then back to sender side afterwards.

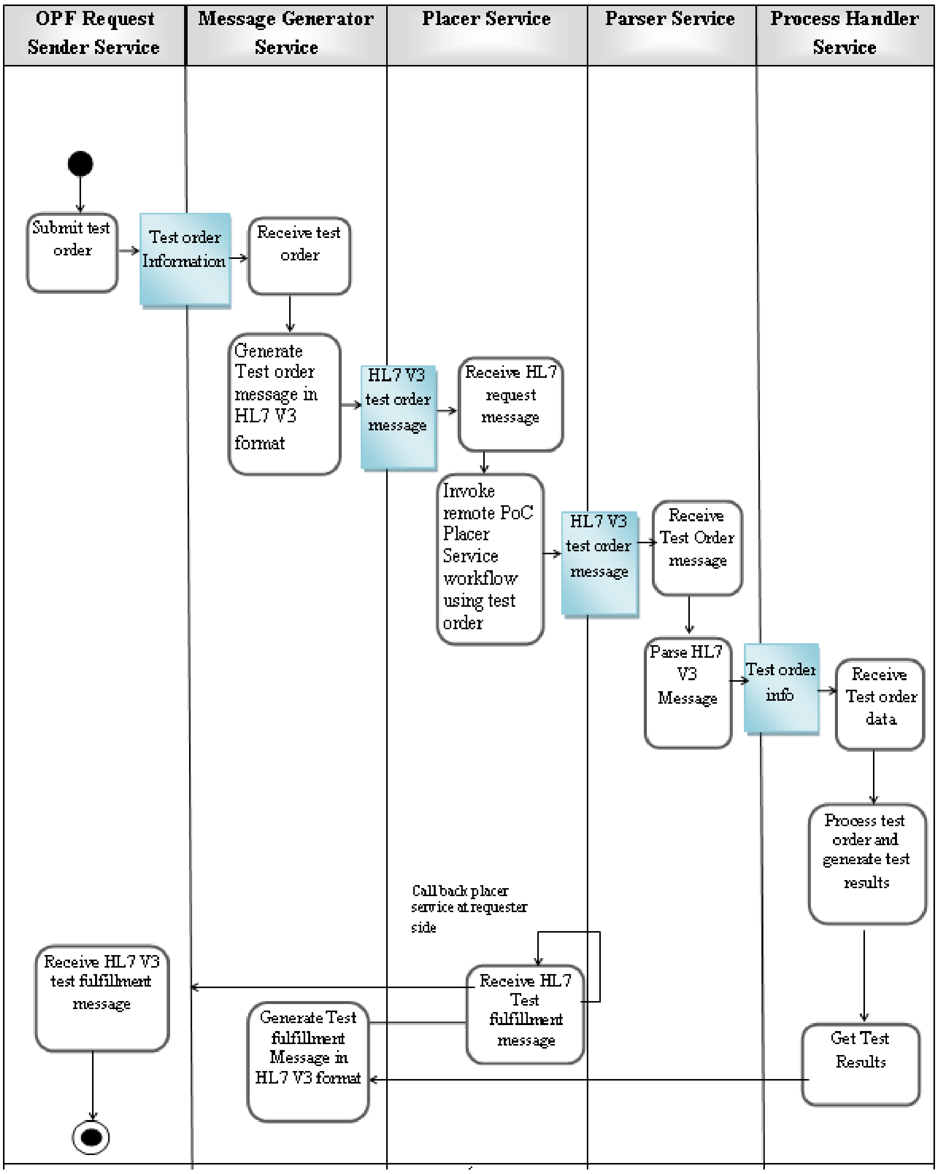



Figure 6.7: OPFS Activity Diagram

At sender side, there is a call back mechanism which is handled through another *Receive* activity. This receive activity is correlated with the first one (i.e. starting *receive* activity), so as to ensure that the received results are related to particular request. Likewise the results are sent to the OPF Request Sender Service. This service in turn forwards these results to the requesting user (that can be any client application).

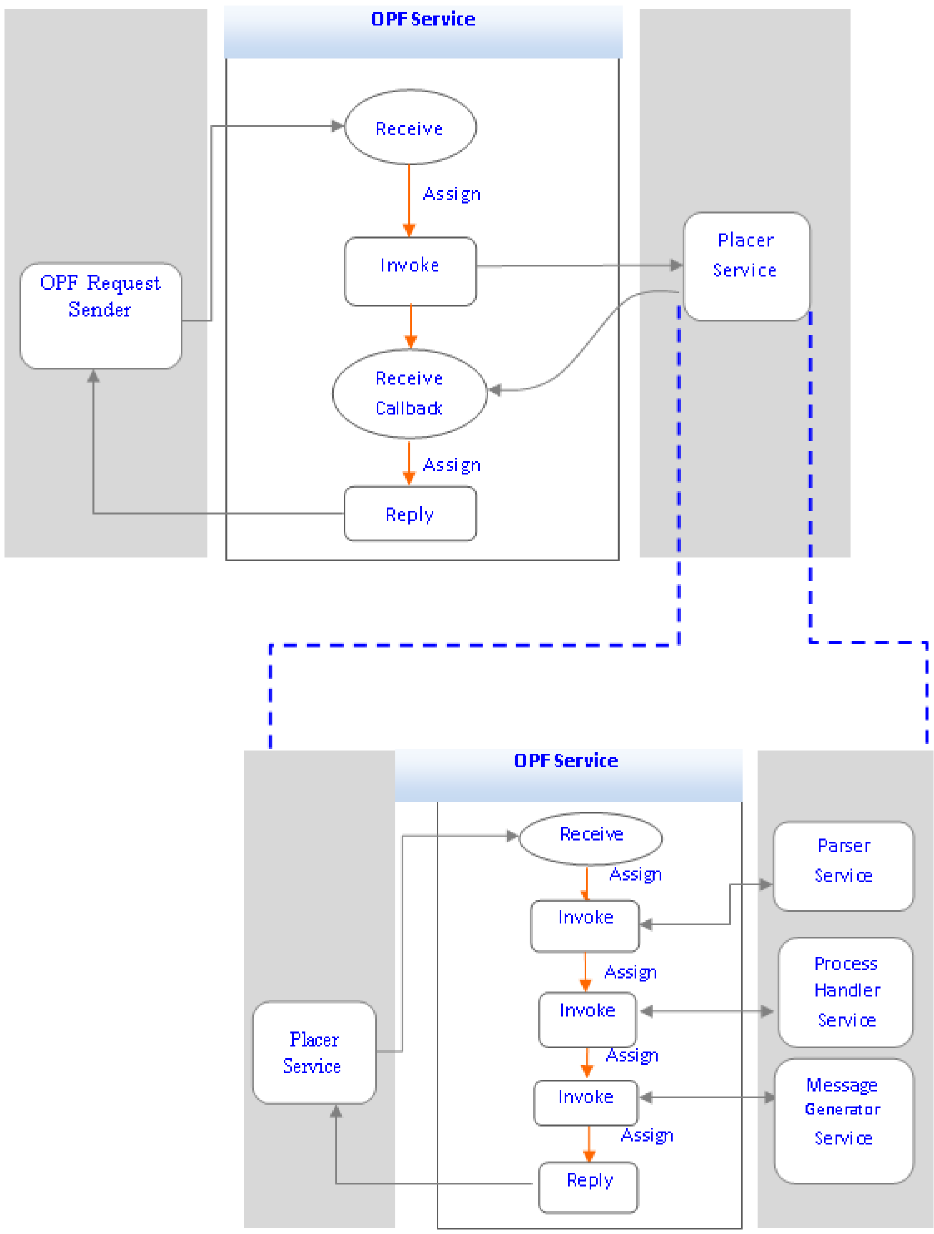



Figure 6.8: OPFS Workflow, designed in BPEL

# Chapter 7

# System Implementation and Evaluation

This chapter signifies the realization of proposed system through the use of HL7 specifications (artifacts and UML information models) and SOA artifacts. The implementation is categorized into six main phases and sub phases. Each service follows these phases and illustrates its interface and implementation specifications. At the end, the proposed system is evaluated against the traditional approach.

## 7.1 Implementation Strategy

The implementation approach of this research work is based on the following major phases:

- Message Generation/Parsing
  - Using HL7 V3 Refined Reference Information Model (RMIM)
  - Java SIG API
- Web services development
  - Designing services using Bottom-Up approach
  - Embedding generated messages in web services
- Workflows creation
  - Placing the developed web services in workflows
  - Designing services workflows using BPEL.

- Workflows Integration
  - Integrating workflows in Java Business Integration (JBI) Environment.
- Test Cases
  - Execution of Test Cases in JBI environment
- Client Application Development
  - Invoking Services workflows through Clients

### 7.1.1 Entity Identification Service:

Following the implementation strategy phases, EIS is described as follows:

**A. Message Generation/Parsing:** EIS mainly involves two types of messages, i.e. EIS request and EIS response message. In HL7 V3 domain, these request and response messages are mapped with Patient Administration: Patient topic interactions and named as follow:

- Patient Registry Get Identifiers Query (EIS request message)
- Patient Registry Get Identifiers Query Response (EIS response message)

The Patient Registry Get Identifiers Query message is generated by its message type, which is based upon a particular RMIM (PRPA_RM201307UV).

Figure 7.1 shows the RMIM of request message, which formulates the basis for generation of Patient Registry Get Identifiers Query message. The message generation in this system is implemented through Java SIG API.

The generated XML message is shown in figure 7.2. The main attribute is <queryId root="632...> from which the Id value is extracted in parsing process and then sent to the database service for finding record against that ID.

The RMIM for generating Patient Registry Get Identifiers Query Response is shown in figure 7.3, while the generated message is illustrated in figure 7.4.

The generated response message retrieves the name and gender information under <name xsi:type="EN"> specific tag, for the same ID "632". Other information can also be retrieved using same RMIM, including detailed demographics of the patient.

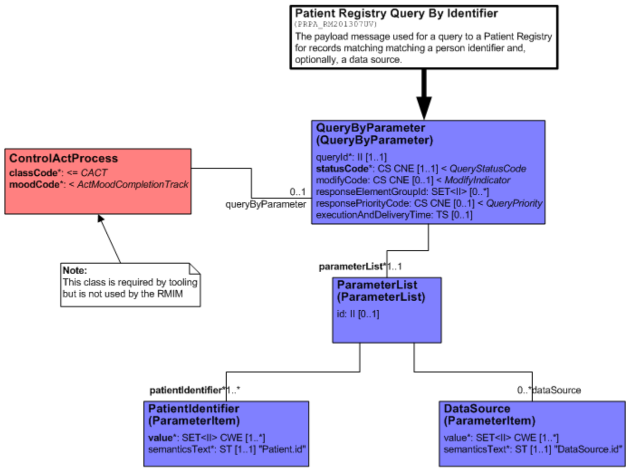



Figure 7.1: RMIM of Patient Registry Get Identifiers Query interaction

```
<?xml version="1.0" encoding="UTF-8"?> <QueryByParameter
xmlns="urn:hl7-org:v3"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
   <queryId root="632" extension="" assigningAuthorityName="" displayable="false"/>
   <statusCode code="new"/>
   <parameterList>
      <id root="192.3.18.33.1" extension="" assigningAuthorityName="" displayable="false"/>
   </parameterList>
</QueryByParameter>
```

Figure 7.2: Generated HL7 V3 "Patient Registry Get Identifiers Query" message for EIS

**B. Web service Development:** Figure 6.4 shows the services involved in EIS workflow. The EIS session starts from client application which sets patient ID as input information and forward it to Message Generator Service. The Message Generator Service takes the value and generates the required "Patient Registry Get Identifier Query" using Java SIG API. The generated message is sent to the intended destination where it is parsed using the same Java SIG API. The parsed message is then sent to the Database Service. The business logic of Database service consist of (patient) demographics' retrieving functions.

Each service involved in EIS workflow has its own well-defined interface. Some services are fine-grained such as:

- EIS_ EntryPoint Service

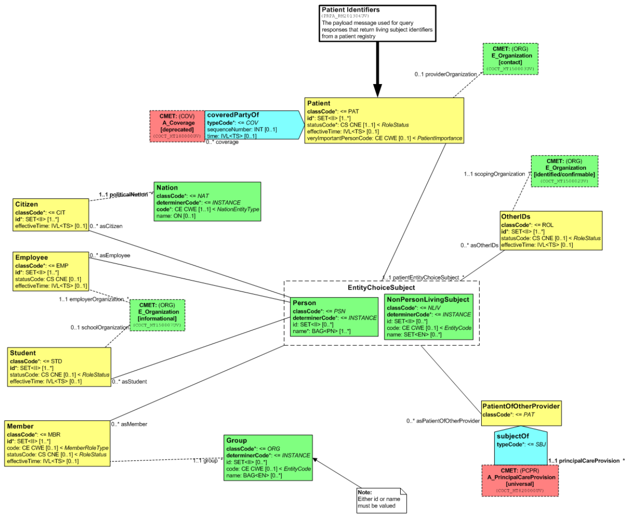



Figure 7.3: RMIM of Patient Registry Get Identifiers Query Response interaction

```
<?xml version="1.0" encoding="UTF-8"?>
 <Patient xmlns="urn:hl7-org:v3"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 classCode="PAT">
   <id xsi:type="II" root="632" extension="patientRemarks " assigningAuthorityName=""
   displayable="false"/>
   <statusCode code="active"/>
   <patientPerson classCode="PSN" determinerCode="INSTANCE">
      <name xsi:type="EN">
         <prefix>Ms</prefix>
         <given>SISTER</given>
         <family>OF ASIM</family>
         <suffix>I</suffix>
      </name>
   </patientPerson>
</Patient>
```

Figure 7.4: Generated HL7 V3 “Patient Registry Get Identifiers Query Response” message for EIS

- XEIS

While some are coarse-grained such as:

- Message Generator Service
- Parser Service
- Database Service

The services description including their interface specifications are shown in the following WSDLs.

### (i) ***EIS Entry Point Service***:

**WSDL:**

```
<?xml version="1.0" encoding="UTF-8"?><!-- Published by JAX-WS RI at http://jax-ws.dev.java.net. RI's version is JAX-WS RI 2.1.3.1-hudson-417-SNAPSHOT. --><!-- Generated by JAX-WS RI at http://jax-ws.dev.java.net. RI's version is JAX-WS RI 2.1.3.1-hudson-417-SNAPSHOT. --><definitions xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://HLH/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://schemas.xmlsoap.org/wsdl/" targetNamespace="http://HLH/" name="EISService">
<types>
<xsd:schema>
<xsd:import namespace="http://HLH/" schemaLocation="http://localhost:9090/IdentificationService/EISService?xsd=1"></xsd:import>
</xsd:schema>
</types>
<message name="EIS_Operation">
<part name="parameters" element="tns:EIS_Operation"></part>
</message>
<message name="EIS_OperationResponse">
<part name="parameters" element="tns:EIS_OperationResponse"></part>
</message>
<portType name="EIS">
<operation name="EIS_Operation">
<input message="tns:EIS_Operation"></input>
<output message="tns:EIS_OperationResponse"></output>
</operation>
</portType>
<binding name="EISPortBinding" type="tns:EIS">
<soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"></soap:binding>
<operation name="EIS_Operation">
<soap:operation soapAction=""></soap:operation>
<input>
<soap:body use="literal"></soap:body>
</input>
<output>
<soap:body use="literal"></soap:body>
</output>
</operation>
</binding>
<service name="EISService">
<port name="EISPort" binding="tns:EISPortBinding">
<soap:address location="http://localhost:9090/IdentificationService/EISService"></soap:address>
</port>
</service>
```

```
</definitions>
```

The WSDL of EIS_EntryPoint service consist of one web method i.e. EIS_Operation which takes patient's identifying data from the client application and forwards it to the Message Generator service.

### (ii) ***Parser Service:***

**WSDL:**

```
<?xml version="1.0" encoding="UTF-8" ?>
<definitions
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
    wssecurity-utility-1.0.xsd"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://parse_package/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://parse_package/" name="ParseWSService"> -
  <types>
    <xsd:schema>
      <xsd:import namespace="http://parse_package/"
      schemaLocation="
          http://localhost:9090/WebApp_Parsing/ParseWSService?xsd=1" />
    </xsd:schema>
  </types>
  <message name="Parse_Operation">
    <part name="parameters" element="tns:Parse_Operation" />
  </message>
  <message name="Parse_OperationResponse">
    <part name="parameters" element="tns:Parse_OperationResponse" />
  </message>
  <portType name="ParseWS">
    <operation name="Parse_Operation">
      <input message="tns:Parse_Operation" />
      <output message="tns:Parse_OperationResponse" />
    </operation>
  </portType>
  <binding name="ParseWSPortBinding" type="tns:ParseWS">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style=
        "document" />
    <operation name="Parse_Operation">
      <soap:operation soapAction="" />
      <input>
        <soap:body use="literal" />
      </input>
      <output>
        <soap:body use="literal" />
      </output>
        </operation>
      </binding>
  <service name="ParseWSService">
  <port name="ParseWSPort"
binding="tns:ParseWSPortBinding">
```

```
    <soap:address location="http://localhost:9090/WebApp_Parsing/ParseWSService" />
  </port>
 </service>
</definitions>
```

The parser service also acts as a central and reusable service. It has one web method i.e. "Parse_Operation" which take the HL7 message and invokes the particular parser code. For each message, different parsing code is required to be implemented, using Java SIG API. Likewise the parsing code is invoked relative to the received HL7 message.

### (iii) *Message Generator Service:*

**WSDL:**

```
<?xml version="1.0" encoding="UTF-8" ?> <definitions
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://mg.org/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://mg.org/"
name="MessageGeneratorServiceService">
  <types>
    <xsd:schema>
      <xsd:import namespace="http://mg.org/" schemaLocation="http://localhost:9090/WebApp_MsgGenerator
      /MessageGeneratorServiceService?xsd=1" />
    </xsd:schema>
  </types>
  <message name="EIS_Request">
    <part name="parameters" element="tns:EIS_Request" />
  </message>
  <message name="EIS_RequestResponse">
    <part name="parameters" element="tns:EIS_RequestResponse" />
  </message>
  <message name="EIS_ResponseOp">
    <part name="parameters" element="tns:EIS_ResponseOp" />
  </message>
  <message name="EIS_ResponseOpResponse">
    <part name="parameters" element="tns:EIS_ResponseOpResponse" />
  </message>
  <message name="RQ_RequestOp">
    <part name="parameters" element="tns:RQ_RequestOp" />
  </message>
  <message name="RQ_RequestOpResponse">
    <part name="parameters" element="tns:RQ_RequestOpResponse" />
  </message>
  <message name="RQ_ResponseOp">
    <part name="parameters" element="tns:RQ_ResponseOp" />
  </message>
  <message name="RQ_ResponseOpResponse">
```

```
  <part name="parameters" element="tns:RQ_ResponseOpResponse" />
</message>
  <portType name="MessageGeneratorService">
  <operation name="EIS_Request">
      <input message="tns:EIS_Request" />
      <output message="tns:EIS_RequestResponse" />
  </operation>
  <operation name="EIS_ResponseOp">
      <input message="tns:EIS_ResponseOp" />
      <output message="tns:EIS_ResponseOpResponse" />
  </operation>
  <operation name="RQ_RequestOp">
      <input message="tns:RQ_RequestOp" />
      <output message="tns:RQ_RequestOpResponse" />
  </operation>
  <operation name="RQ_ResponseOp">
      <input message="tns:RQ_ResponseOp" />
      <output message="tns:RQ_ResponseOpResponse" />
  </operation>
</portType>
<binding name="MessageGeneratorServicePortBinding" type="
    tns:MessageGeneratorService">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style=
      "document" />
  <operation name="EIS\_Request">
      <soap:operation soapAction="" />
      <input>
          <soap:body use="literal" />
      </input>
      <output>
          <soap:body use="literal" />
      </output>
  </operation>
  <operation name="EIS\_ResponseOp">
      <soap:operation soapAction="" />
      <input>
          <soap:body use="literal" />
      </input>
      <output>
          <soap:body use="literal" />
      </output>
  </operation>
  <operation name="RQ\_RequestOp">
      <soap:operation soapAction="" />
      <input>
          <soap:body use="literal" />
      </input>
      <output>
          <soap:body use="literal" />
      </output>
  </operation>
  <operation name="RQ\_ResponseOp">
      <soap:operation soapAction="" />
      <input>
          <soap:body use="literal" />
      </input>
```

```
        <output>
            <soap:body use="literal" />
        </output>
    </operation>
  </binding>
  <service name="MessageGeneratorServiceService">
    <port name="MessageGeneratorServicePort" binding="
        tns:MessageGeneratorServicePortBinding">
      <soap:address location="http://localhost:9090/WebApp\
          _MsgGenerator/MessageGeneratorServiceService" />
    </port>
  </service>
</definitions>
```

The Message Generator Service acts as a central reusable service. It has several web methods which are implemented in order to generate HL7 messages particular to any communication session. This service is flexible enough to accommodate any number of methods in it. The web methods invoke those modules which are implemented using Java SIG API.

### (iv) *Database Service:*

### WSDL:

```
<?xml version="1.0" encoding="UTF-8" ?> <definitions
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
    wssecurity-utility-1.0.xsd"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://dbPackage/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://dbPackage/" name="DBServiceService"> <types>
<xsd:schema>
  <xsd:import namespace="http://dbPackage/" schemaLocation="
      http://localhost:9090/WebAppDB/DBServiceService?xsd=1" />
  </xsd:schema>
  </types>
<message name="connect">
  <part name="parameters" element="tns:connect" />
</message>
<message name="connectResponse">
  <part name="parameters" element="tns:connectResponse" />
</message>
<message name="releaseResources">
  <part name="parameters" element="tns:releaseResources" />
</message>
<message name="releaseResourcesResponse">
  <part name="parameters" element="tns:releaseResourcesResponse" />
</message>
<message name="searchPatientById">
  <part name="parameters" element="tns:searchPatientById" />
</message>
```

```
<message name="searchPatientByIdResponse">
  <part name="parameters" element="tns:searchPatientByIdResponse" />
</message>
<portType name="DBService">
<operation name="connect">
  <input message="tns:connect" />
  <output message="tns:connectResponse" />
</operation>
<operation name="releaseResources">
  <input message="tns:releaseResources" />
  <output message="tns:releaseResourcesResponse" />
</operation>
<operation name="searchPatientById">
  <input message="tns:searchPatientById" />
  <output message="tns:searchPatientByIdResponse" />
</operation>
</portType>
<binding name="DBServicePortBinding" type="tns:DBService">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="
      document" />
  <operation name="connect">
    <soap:operation soapAction="" />
    <input>
       <soap:body use="literal" />
    </input>
    <output>
       <soap:body use="literal" />
    </output>
  </operation>
  <operation name="releaseResources">
    <soap:operation soapAction="" />
    <input>
       <soap:body use="literal" />
    </input>
    <output>
       <soap:body use="literal" />
    </output>
    </operation>
   <operation name="searchPatientById">
     <soap:operation soapAction="" />
     <input>
       <soap:body use="literal" />
     </input>
     <output>
       <soap:body use="literal" />
     </output>
   </operation>
</binding>
<service name="DBServiceService">
<port name="DBServicePort" binding="tns:DBServicePortBinding">
   <soap:address location="
       http://localhost:9090/WebAppDB/DBServiceService" />
</port>
</service>
</definitions>
```

The database service mainly involves searchPatientById web method. In this research work, Database service needs to search various records, but its functionality can be extended as per the healthcare environment needs. This service also acts as a reusable service, the instances of which can be deployed at multiple point-of-cares.

### (v) ***XEIS***

**WSDL:**

```
<?xml version="1.0" encoding="UTF-8" ?>
<definitions xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://is.org/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://is.org/" name="IS_InstncService">
<types>
    <xsd:schema>
        <xsd:import namespace="http://is.org/"
        schemaLocation="http://10.3.18.66:9090/IdentificationService_Instance/IS_InstncService?xsd=1"
            />
    </xsd:schema>
</types>
<message name="IS_InstanceOp">
    <part name="parameters" element="tns:IS_InstanceOp" />
</message>
<message name="IS_InstanceOpResponse">
    <part name="parameters" element="tns:IS_InstanceOpResponse" />
</message>
<portType name="IS_Instnc">
<operation name="IS_InstanceOp">
    <input message="tns:IS_InstanceOp" />
        <output message="tns:IS_InstanceOpResponse" />
    </operation>
</portType>
<binding name="IS_InstncPortBinding" type="tns:IS_Instnc">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document" />
  <operation name="IS_InstanceOp">
    <soap:operation soapAction="" />
    <input>
        <soap:body use="literal" />
    </input>
    <output>
        <soap:body use="literal" />
    </output>
  </operation>
</binding>
<service name="IS_InstncService">
```

```
<port name="IS_InstncPort" binding="tns:IS_InstncPortBinding">
    <soap:address location="http://10.3.18.66
        :9090/IdentificationService_Instance/IS_InstncService" />
</port>
</service>
</definitions>
```

XEIS is representing the eXternal EIS instance deployed on some remote point-of-care, for instance, on test center. XEIS is a service description (WSDL) of overall flow of EIS which is invoked in case of unavailability of patient's demographic data on the local site.

### C. Workflows Creation:

The workflow of EIS is depicted in figure 6.4, and is designed in BPEL which comes with Netbeans environment. The BPEL engine is used for executing and deploying BPEL workflows.

### D. Workflows Integration:

BPEL workflows are integrated in JBI environment. JBI provides a standard application framework for integration and deployment of various kinds of services (can be local or external). Glassfish server provides openESB solutions using JBI at its base, in order to compose multiple web services in a loosely coupled manner.

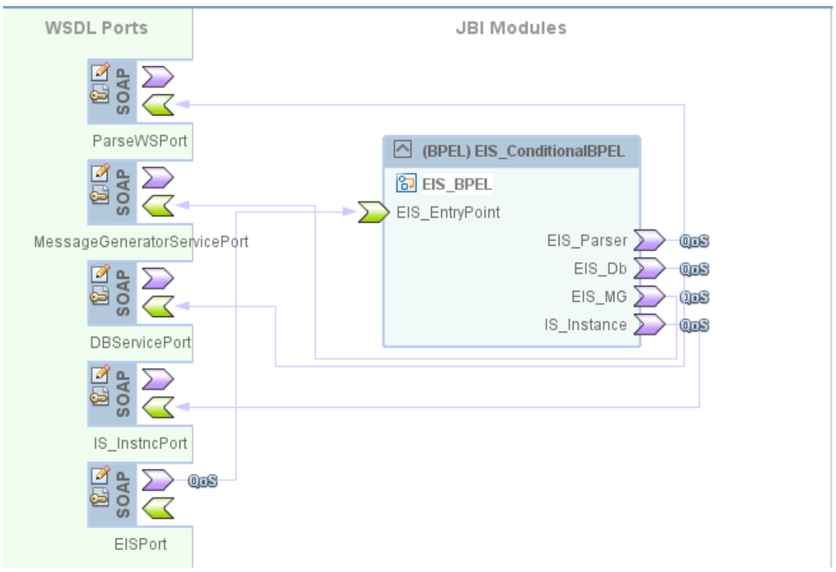



Figure 7.5: EIS Assembly of service units

Figure 7.5 shows EIS composite application having assembly of service units, deployed in JBI server.

**E. Test cases:**

The test cases on deployed BPEL workflows are created and executed in same EIS composite application project. The input and output XML files are shown in figure 7.6 and 7.7.

```
<soapenv:Envelope
xsi:schemaLocation="http://schemas.xmlsoap.org/soap/envelope/
http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:hlh="http://HLH/">
  <soapenv:Body>
    <hlh:EIS_Operation>
      <!--Optional:-->
      <EIS_ReqMsg>
          <QueryByParameter xmlns="urn:hl7-org:v3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
          <queryId root="611" extension="" assigningAuthorityName="" displayable="false"/>
              <statusCode code="new"/>
              <parameterList>
                 <id root="192.3.18.33.1" extension="" assigningAuthorityName="" displayable="false"/>
                    <patientIdentifier>
                        <value xsi:type="ST" representation="TXT" mediaType="text/plain">Ali</value>
                        <semanticsText representation="TXT" mediaType="text/plain">pName</semanticsText>
                    </patientIdentifier>
              </parameterList>
          </QueryByParameter>
      </EIS_ReqMsg>
    </hlh:EIS_Operation>
  </soapenv:Body>
</soapenv:Envelope>
```

Figure 7.6: EIS Test Case input.xml

**F. Client Application Development:**

The EIS client application can be a Java application or web application, in which web reference of EIS composite application WSDL is given. Using that reference the request is travelled across multiple services and brings results back to the client application in a transparent manner.

### 7.1.2 Result Query Service:

**A. Message Generation Parsing:**

```
<?xml version="1.0" encoding="UTF-8"?> <SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://schemas.xmlsoap.org/soap/envelope/
http://schemas.xmlsoap.org/soap/envelope/"> <SOAP-ENV:Body>
  <ns1:EIS_OperationResponse xmlns:msgns="http://HLH/" xmlns:ns1="http://HLH/">
    <return xmlns="">
       &lt;?xml version="1.0" encoding="UTF-8"?&gt;
          &lt;Patient xmlns="urn:hl7-org:v3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          classCode="PAT"&gt;
          &lt;id xsi:type="II" root="611" extension="patientRemarks " assigningAuthorityName=""
          displayable="false"/&gt;
             &lt;statusCode code="active"/&gt;
             &lt;patientPerson classCode="PSN" determinerCode="INSTANCE"&gt;
               &lt;name xsi:type="EN"&gt;
                 &lt;prefix&gt;Mr&lt;/prefix&gt;
                 &lt;given&gt;C/O&lt;/given&gt;
                 &lt;family&gt;GHULAM RABBANI&lt;/family&gt;
                 &lt;suffix&gt;I&lt;/suffix&gt;
                 &lt;/name&gt;
               &lt;/patientPerson&gt;
             &lt;/Patient&gt;
    </return>
  </ns1:EIS_OperationResponse>
</SOAP-ENV:Body> </SOAP-ENV:Envelope>
```


Figure 7.7: EIS Test Case Output.xml

RQS mainly involves two types of messages, i.e. RQS request and RQS response message. In HL7 V3 domain, these request and response messages are mapped with Laboratory domain interactions and named as follows:

- Find Result Query (RQS request message)
- Find Result Query Response (RQS response message)

The Find Result Query message is generated by its message type, which is based upon a particular RMIM; POLB_RM300000UV.

Figure 7.8 shows the RMIM of RQS request message, which is called as "Find Result Query" message.

The illustration of Find Result Query RMIM follows the generated message in figure 7.9. The main attributes in the message are <value xsi:type="II" root="819" ....> and <patientName>.....</patientName> (XML code block) from which the identifying information is extracted in parsing process and then sent to the database service for finding tests records against that values.

The RMIM for generating Find Result Query Response; POLB_RM004000UV is shown in figure 7.10, while the generated message is illustrated in figure 7.11.

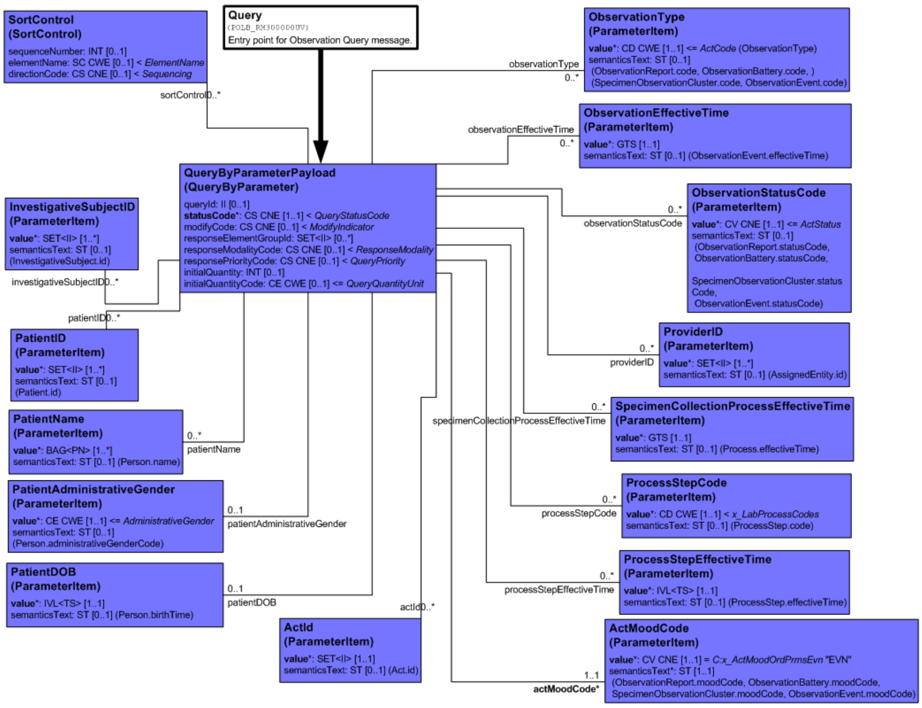



Figure 7.8: RMIM of "Find Result Query" message

```
<?xml version="1.0" encoding="UTF-8"?>
<QueryByParameterPayload xmlns="urn:hl7-org:v3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
   <queryId root="127.6.5.3" extension=".." assigningAuthorityName="" displayable="false"/>
    <statusCode code="Nullified"/>
    <actMoodCode>
      <value xsi:type="CS" code="EVN"/>
      <semanticsText representation="TXT" mediaType="text/plain">ResultQuery</semanticsText>
    </actMoodCode>
    <patientID>
      <value xsi:type="II" root="819" extension="" assigningAuthorityName="" displayable="false"/>
      <semanticsText representation="TXT" mediaType="text/plain">PatID</semanticsText>
    </patientID>
    <patientName>
      <value xsi:type="EN">
         <prefix>PrefixName</prefix>
         <given>GH</given>
         <family>SUGHRA</family>
         <suffix>I</suffix>
      </value>
      <semanticsText representation="TXT" mediaType="text/plain">PatName</semanticsText>
    </patientName>
</QueryByParameterPayload>
```


Figure 7.9: Generated "Find Result Query" Message in HL7 V3 format

The generated response message contains the status code as <statusCode code="Nullified">, test result value (for a particular case study data) as <value xsi:type="ST">10.0</value>, name and gender information under <name xsi:type="EN"> for the same ID "819", shown in <id xsi:type="II" root="819"...> Other information can also be generated using same RMIM (reference; figure 7.10); including detailed test(s) values for a particular patient.

**B. Web service Development:**

Figure 6.6 illustrates the services involved in RQS workflow. The session starts from client application which gives some patient's identifying information and forward them to Message Generator Service. The Message Generator Service takes the values and generates the required "Find Result Query" message using Java SIG API. The generated message is sent to the required destination and is parsed using the same Java SIG API, after which it is sent to the Database Service. The business logic of Database service consist of multiple functions which retrieve the previous tests results (records) of a particular patient. The services in RQS workflow are well-defined with clear granularities. RQ_ EntryPoint Service and XRQS are fine-grained services, while the coarse-grained are same as for EIS.

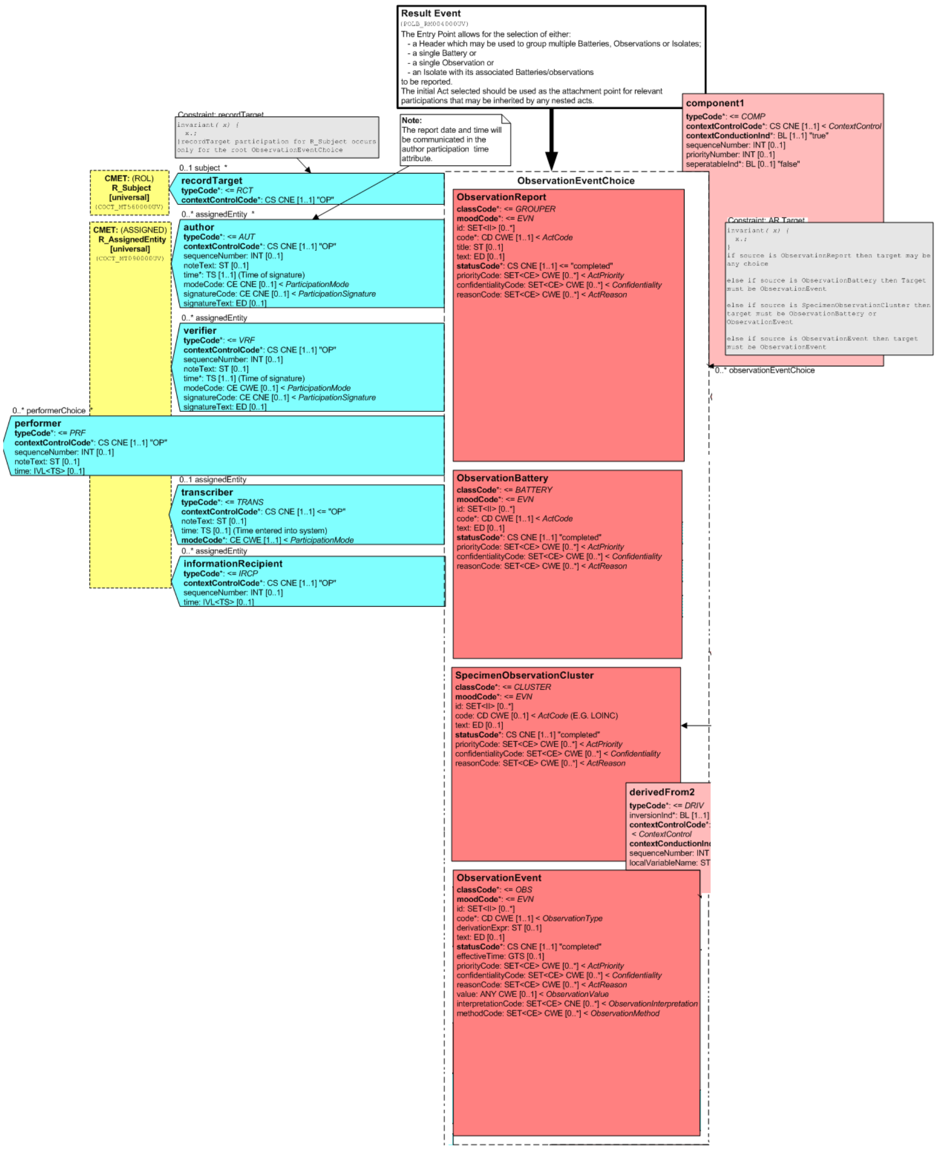



Figure 7.10: RMIM of "Find Result Query Response" message

```xml
<?xml version="1.0" encoding="UTF-8"?> <ObservationEvent
xmlns="urn:hl7-org:v3"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
classCode="OBS" moodCode="EVN">
   <code code="ObservationType" codeSystem="ActCode"/>
   <statusCode code="Nullified"/>
   <value xsi:type="ST" representation="TXT" mediaType="text/plain">10.0</value>
   <recordTarget typeCode="RCT" contextControlCode="OP">
      <subject classCode="PAT">
         <id xsi:type="II" root="819" extension="patientRemarks " assigningAuthorityName="" displayable="false"/>
         <statusCode code="RoleStatus"/>
         <patientPerson classCode="PSN" determinerCode="INSTANCE">
            <name xsi:type="EN">
               <prefix>Ms</prefix>
               <given>GH</given>
               <family>SUGHRA</family>
               <suffix>I</suffix>
            </name>
            <administrativeGenderCode code="Female" codeSystem="10173"/>
            <addr xsi:type="AD">w-6-b-7</addr>
         </patientPerson>
      </subject>
   </recordTarget>
</ObservationEvent>
```


Figure 7.11: Generated ”Find Result Query Response” Message in HL7 V3 format

(i) ***RQ Entry Point Service***:
**WSDL:**

```xml
<?xml version="1.0" encoding="UTF-8" ?> <definitions
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://ws.rq.org/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://ws.rq.org/" name="ResultQueryWSService"> –
<types>
    <xsd:schema>
        <xsd:import namespace="http://ws.rq.org/" schemaLocation="http://localhost:9090/ResultQueryWebApp/ResultQueryWSService?xsd=1" />
    </xsd:schema>
</types>
<message name="QueryingResult">
    <part name="parameters" element="tns:QueryingResult" />
</message>
<message name="QueryingResultResponse">
    <part name="parameters" element="tns:QueryingResultResponse" />
</message>
<portType name="ResultQueryWS">
    <operation name="QueryingResult">
```

```
        <input message="tns:QueryingResult" />
        <output message="tns:QueryingResultResponse" />
    </operation>
</portType>
<binding name="ResultQueryWSPortBinding" type="tns:ResultQueryWS">
<soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document" />
    <operation name="QueryingResult">
    <soap:operation soapAction="" />
        <input>
            <soap:body use="literal" />
        </input>
        <output>
            <soap:body use="literal" />
        </output>
   </operation>
</binding>
<service name="ResultQueryWSService">
<port name="ResultQueryWSPort" binding="tns:ResultQueryWSPortBinding">
    <soap:address location="http://localhost:9090/ResultQueryWebApp/ResultQueryWSService" />
</port>
</service>
</definitions>
```

**(ii) *XRQS*:**

XRQS is a service description (WSDL) which represents the eXternal RQS instance (same as XEIS), deployed on some remote point-of-care (which can be test center in real healthcare environment).

The service descriptions for Message Generators, Parser and Database services are already given in EIS implementation section.

**C. Workflows Creation:**

Figure 6.6 shows RQS workflow with multiple services, designed in Netbeans BPEL editor. The source code of BPEL workflow is validated by XML checker present in the same editor.

**D. Workflows Integration:**

Composite applications represent composition of services that are federated in JBI environment. RQS composite application itself acts as a service. After being deployed, a WSDL is generated for overall composed services. Figure 7.12 shows RQS assembly of service units, deployed in JBI server.

**E. Test cases:**

The test cases for RQS executed on RQS are shown in figure 7.13 and 7.14.

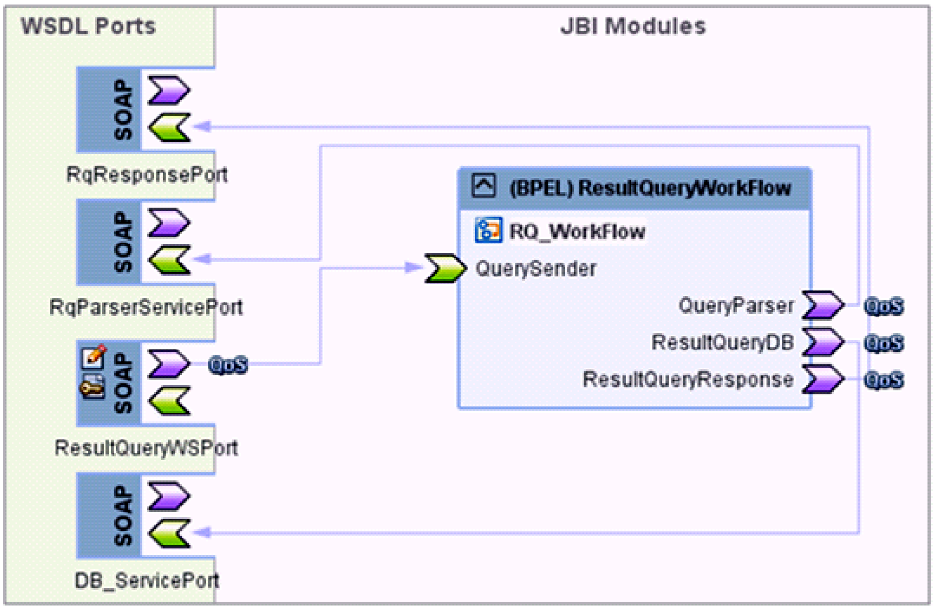
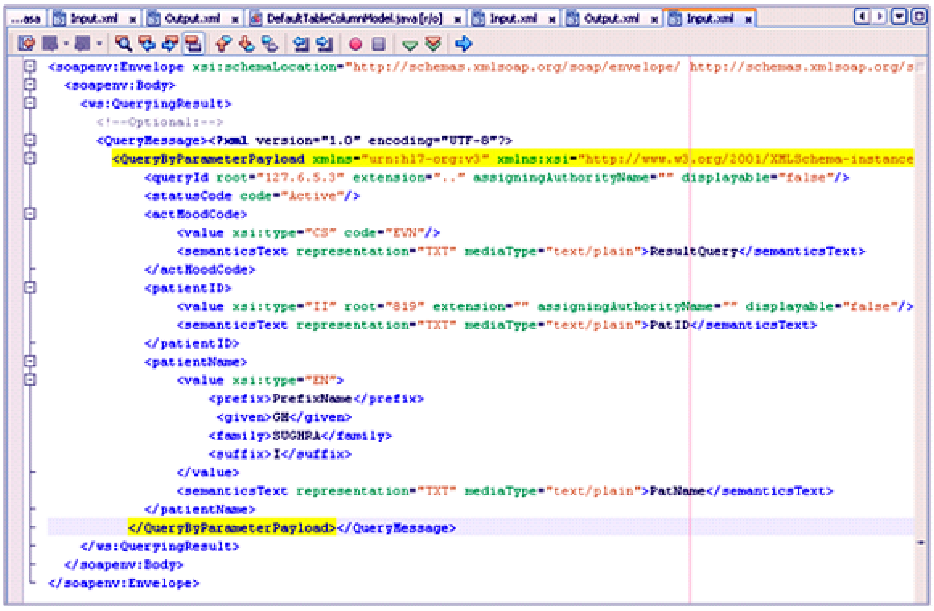



Figure 7.12: RQS Assembly of service units




```xml
<soapenv:Envelope xsi:schemaLocation="http://schemas.xmlsoap.org/soap/envelope/ http://schemas.xmlsoap.org/s
  <soapenv:Body>
    <ws:QueryingResult>
      <!--Optional:-->
      <QueryMessage><?xml version="1.0" encoding="UTF-8"?>
        <QueryByParameterPayload xmlns="urn:hl7-org:v3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance
            <queryId root="127.6.5.3" extension=".." assigningAuthorityName="" displayable="false"/>
            <statusCode code="Active"/>
            <actMoodCode>
                <value xsi:type="CS" code="EVN"/>
                <semanticsText representation="TXT" mediaType="text/plain">ResultQuery</semanticsText>
            </actMoodCode>
            <patientID>
                <value xsi:type="II" root="819" extension="" assigningAuthorityName="" displayable="false"/>
                <semanticsText representation="TXT" mediaType="text/plain">PatID</semanticsText>
            </patientID>
            <patientName>
                <value xsi:type="EN">
                    <prefix>PrefixName</prefix>
                     <given>GH</given>
                    <family>SUGHRA</family>
                    <suffix>I</suffix>
                </value>
                <semanticsText representation="TXT" mediaType="text/plain">PatName</semanticsText>
            </patientName>
          </QueryByParameterPayload></QueryMessage>
    </ws:QueryingResult>
  </soapenv:Body>
</soapenv:Envelope>
```

Figure 7.13: RQS Test Case input.xml

**F. Client Application Development:**

The RQS client application is a java application (can be web application), in which web reference of RQS composite application's WSDL is given. Using that reference the request is travelled across multiple services and brings query test results back to the client application. Figure 7.14 is also showing the desired output of a client application.

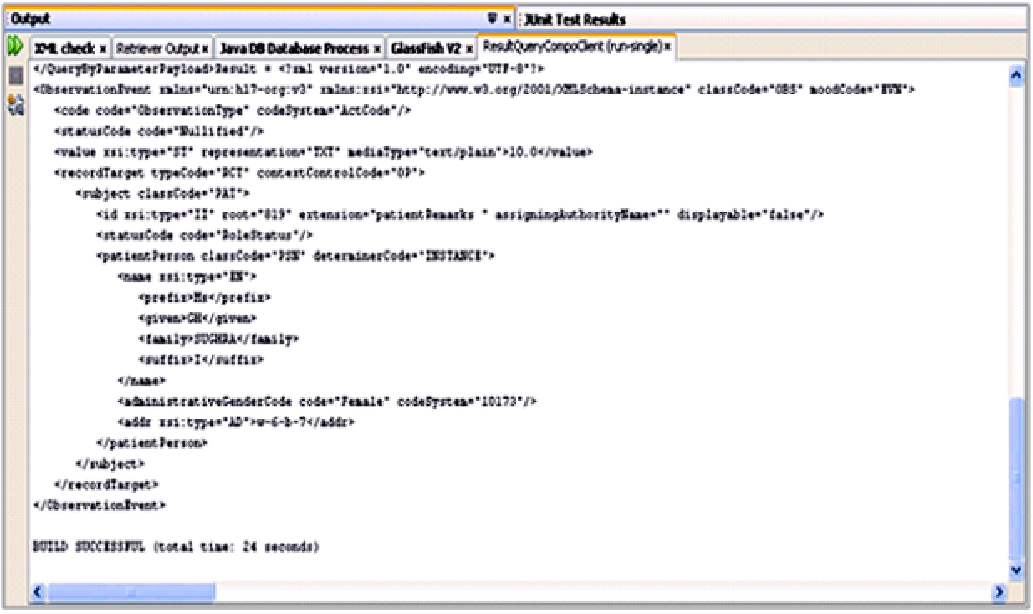



Figure 7.14: RQS Test Case, Output.xml

## 7.2 System Evaluation

SOA4HL7 is a new paradigm and there are very few implementations in "SOA for Healthcare" domain. Providing comparison with related applications is important to evaluate the proposed system. The developed system is first evaluated against the traditional approach i.e. conventional messaging infrastructure of HL7 V3 shown in Table 7.1.

### 7.2.1 Evaluation Results

HL7 V3 Laboratory Messaging using SOA Infrastructure (HLMSI) system is evaluated on real world clinical system. The evaluation metrics are scalability, flexibility, cost-effectiveness along with ease-of-use and adaptability

| Evaluation Metrics | Communication Infrastructure | |
|---|---|---|
| | HL7 V3 MI | HLMSI |
| Modifiability | Yes(infrastructure change) | Yes(no infrastructure change, easy integration) |
| Performance | Less (overall perspective) | More (System responds to the user in less time) |
| Scalability | Less (difficult to manage) | More (easily managed and governed) |
| Independent of development technology | Yes (semi-independent) | Yes (completely independent) |
| Costs (compilation costs, load balancing) | More | Less |
| Distributed Nature | Less | More |
| Seamless changes/Transparency | Less | More |
| Dynamic processing (Runtime Invocation) | Less | More |
| Reliability (duplicate request sending) | Yes (user is double charged) | Yes (request remains in consistent state) |

Table 7.1: HL7 Laboratory Messaging Infrastructure(HLMSI) Vs HL7 V3 MI

of the system to dynamic changes. Following tables shows the evaluation results by taking several metrics. Table 7.2 shows the evaluation based upon interoperability.

| **HLMSI** | **MI** |
|---|---|
| HLMSI injects interoperability between disparate healthcare organizations by standardizing data representation (XML), business logic definition (WSDL), and message exchange (SOAP). | Conventional system provides standardization in data interoperability. |
| HLMSI can engage in multiple interoperability agreements at once for various services. | Conventional system can take on single interoperability agreement at once. |
| HLMSI provides;<br>Data interoperability — HL7 V3<br>Platform Interoperability— SOA | Data interoperability |

Table 7.2: Evaluation based upon Interoperability Metric

| **HLMSI** | **MI** |
|---|---|
| HLMSI represents communication flows in or out of a kind of "bus" to and from all the parties. | Conventional infrastructure represents the point-to-point integration approach. Here, numerous parties use individual lines of communication, resulting in a complex, spaghetti-like architecture. |

Table 7.3: Evaluation based upon Integration Metric

| HLMSI | MI |
|---|---|
| HLMSI can accommodate any number of services because of the decoupled nature of services and encapsulation of business logic in them. | Conventional system is less scalable as the design of the system is tightly bound to interaction models of healthcare organizations. |

Table 7.4: Evaluation based upon Scalability Metric

| HLMSI | MI |
|---|---|
| In HLMSI, services are reusable units, which expose business functions in standardized way and are consumed across multiple projects. | Conventional infrastructure lacks reusability. For each functionality to be added, it needs to modify the design and development, hence more costs are involved. |
| HLMSI provides the flexibility to the healthcare organizations to coordinate these services as per their business workflow. | The conventional MI lacks flexibility and doesnt give a free hand to the organizations to use the system as per the real healthcare workflows. |

Table 7.5: Evaluation based upon Reusability, Cost-effectiveness and Flexibility Metrics

**Flexibility Another Perspective:** Unlike MI, HLMSI system can be easily extended to other domains.

**Adaptability:** Flexibility also leads to adaptability. Defining descriptive and extensible contract schema. new functionalities can be easily incorporated in the newer version of services without breaking existing service. While MI lacks the capability.

# Chapter 8

# Conclusions and Future Research

This chapter concludes the findings of this research work. It provides an analysis of acquired results and future dimensions for extending the accomplished work. This section presents an overall review of vitality of the work done.

## 8.1 Conclusions

Information and Communication Technology is the key to revolutionize the progression and innovation of healthcare information systems. Healthcare informatics standards are developed to bring interoperability and integration across multiple healthcare units but the objective is not achieved to a wider extent.

HL7 organization along with OMG developed a SOA platform aka HSSP, which specifies the guidelines in the form of interface specifications for healthcare service. HSSP methodology deals with policies, service functional models and profiles of overall healthcare infrastructure. However it does not provide the implementation specifications of services. Moreover it focuses on the high level aspects and mega architecture of healthcare enterprise environment. This research focused on laboratory and patient administration domain and proposed HL7 V3 based SOA framework for healthcare, with certain guidelines of HSSP.

HL7 V3 traditional communication system is not capable to provide flexibility, reusability and adaptability altogether. The alignment of HL7 V3 based systems with SOA specifications can fulfill the major requirement to achieve this objective. HLMSI enables optimization of complex healthcare procedures in the form of business process workflows. The proposed architecture doesnt replace the conventional messaging infrastructure, rather uses same MI in its standardized services workflows.The combined workings of SOA frameworkwith HL7 V3 standard give maximum benefits at one place. The two interoperability standards (SOA & HL7 V3) thus give full-fledged solution for solving interoperability issues and dynamic inaccessibility matters in a widely distributed environment. The proposed system shows business-ICT integration in healthcare processes.

BPEL is a business workflow language considered in the proposed system, to design business process models that are captured from real environment of a local clinical laboratory but based on HL7 V3 laboratory domain specifications.

## 8.2 Future Work

The proposed work has currently focused on laboratory and patient administration domain, but it is intended to extend it over other HL7 V3 domains. Moreover, most of the services in the proposed system are exhibiting synchronous communication. It is required to design and develop asynchronous services in the same framework following similar implementation methodology (as the proposed services followed). The services are not based on semantics. The future work revolves around semantic interoperability thus moving towards Semantic SOA (SSOA) for more automation. HLMSI system can also provide a base for the research areas of cloud computing and semantic SOA .

# Bibliography

[1] Kaiser Permanente Alan Honey. Services and the hl7 v3 dynamic model a discussion document / white paper, October 2006. (last visited in December 2010).

[2] Asc x12. `http://en.wikipedia.org/wiki/ASC_X12`. (last modfied on 13th february 2011).

[3] George Beeler and et al. Hl7 reference information model, ansi/hl7 v3 rim, r1-2003). `http://www.hl7.org/implement/standards/rim.cfm`, HL7 Normative 2009. (last visited in December 2010).

[4] M.F. Collen. A vision of health care and informatics in 2008. volume 6, page 1. BMJ Publishing Group Ltd, 1999.

[5] CE-IT Community. Medical device integration: References, resources, and standards. `http://www.ceitcollaboration.org/docs/GuidanceRecognizedStandards.pdf`. (last visited in January 2011).

[6] Digital imaging and communications in medicine. `http://en.wikipedia.org/wiki/Digital_Imaging_and_Communications_in_Medicine`. (last modified on 7 April 2011).

[7] History of electronic medical records. `http://www.miraclesmagicinc.com/medical/history-electronic-medical-records.html`. (Last visited in December 2010).

[8] Corepoint Health. The hl7 evolution, comparing hl7 version 2 to version 3, including a history of version 2. `http://www.corepointhealth.com/sites/default/files/whitepapers/hl7-v2-v3-evolution.pdf`, 2010. (last visited in January 2011).

[9] K. Heubusch. Interoperability What it Means, Why it Matters. volume 77, page 26. HEALTH INFORMATION MANAGEMENT ASSOCIATION, 2006.

[10] Introduction, health informatics, faculty of medicine. http://healthinfo.med.dal.ca/HL7Intro/introduction.html. (last visited in January 2011).

[11] HL7 and OMG. Hl7 version 3 standard: Identification service, release 2. http://www.hl7.org/v3ballot2010jan/html/welcome/environment/index.htm. (Last Ballot: Normative Ballot 1 - September 2009).

[12] HL7 and OMG. The practical guide for soa in health care. http://hssp.wikispaces.com/file/view/2008-12-29+SOA%2BHealthcare%2BPractical%2BGuide%2Bv1.0.pdf, December 2008. (last visited in December 2010).

[13] K. Holley, J. Palistrant, and S. Graham. effective SOA governance. volume 17, page 2008, 2006.

[14] 2007-2011 Health Level Seven International. Hl7 backgrounder brief. http://www.hl7.org/newsroom/HL7backgrounderbrief.cfm. (last visited in January 2011).

[15] HL7 International. Domain: Transmission infrastructure. http://www.hl7.org/v3ballot2010jan/html/welcome/environment/index.htm, October 2004. (last visited in December 2010).

[16] HL7 International. Hl7 v3 guide. http://www.hl7.org/v3ballot/html/help/v3guide/v3guide.html, 2008. (last visited in December 2010).

[17] HL7 International. Hl7 version 3 standard: Transport specification - ebxml, release 2). http://www.hl7.org/v3ballot2010jan/html/welcome/environment/index.htm, March 2008. (last visited in December 2010).

[18] HL7 International. Transport specification: Mllp, release 2). http://www.hl7.org/v3ballot2010jan/html/welcome/environment/index.htm, July 2009. (last visited in December 2010).

[19] HL7 International and OMG. Healthcare services specification project (hssp). http://hssp.wikispaces.com, October 2004. (last visited in December 2010).

[20] Grahame Grieve John Koisch, Alan Honey. Hl7 interoperability paradigms), September 2007. (last visited in December 2010).

[21] F. Kart, G. Miao, LE Moser, and PM Melliar-Smith. A distributed e-Healthcare system based on the Service Oriented Architecture. pages pp.652–659. IEEE Computer Society, 2007.

[22] Health level 7, from wikipedia, the free encyclopedia. http://en.wikipedia.org/wiki/Health_Level_7, December 2009. (last visited in December 2010).

[23] Information technology impact on the healthcare industry. http://www.booz.com/me/home/what_we_think/40007409/40007869/42008005, May 2008. (last visited in December 2010).

[24] Open clinical knowledge management for medical care. http://www.openclinical.org/medTermLoinc.html, July 2005. (last visited in January 2011).

[25] Lopez-Sanchez lonewolf. History of electronic health records. http://www.ehow.com/facts_4895288_history-electronic-health-records.html#ixzz17IxqXeq4. (Last visted in December 2010).

[26] Microsoft Corporation Mauro Regio. Web services enablement for healthcare hl7 applications). http://msdn.microsoft.com/en-us/library/ms954603.aspx, August 2005. (last visited in December 2010).

[27] Juha Mykknen. Standardization and interoperability in healthcare it. http://his.uku.fi/exporthis/Seminaari/Export-JuhaM.pdf.

[28] National council for prescription drug programs. http://en.wikipedia.org/wiki/National_Council_for_Prescription_Drug_Programs, January 2010. (last visited in December 2010).

[29] HIMSS platform. Integration and interoperability. http://www.himss.org/ASP/topics_integration.asp. (last visited in January 2011).

[30] Andrea Ribick. Health level seven international. http://www.hl7.org/documentcenter/public_temp_EDA3CA7E-1C23-BA17-0C83AE6C1FF7EC76/pressreleases/HL7_PRESS_20100526.pdf. (last visited in January 2011).

[31] J. Schekkerman. Enterprise architecture and service oriented architecture (soa). http://www.enterprise-architecture.info/Images/ServicesOrientedEnterprise/EA_Service-Oriented-Architecture1.htm. (last visited in December 2010).

[32] Special Interest Group (SOA SIG). Service oriented architecture and hl7 v3 methodology, version no. 0.9), November 2006. (last visited in December 2010).

[33] M. Turner, F. Zhu, I. Kotsiopoulos, M. Russell, D. Budgen, K. Bennett, P. Brereton, J. Keane, P. Layzell, and M. Rigby. Using web service technologies to create an information broker: an experience report. IEEE Computer Society, 2004.

[34] HL7 UK. Hl7 version 2. http://www.hl7.org.uk/version2group/v2.asp. (last modified on 20th September 2007).

[35] J. Walker, E. Pan, D. Johnston, J. Adler-Milstein, D.W. Bates, and B. Middleton. Web-Exclusive Abstracts, The Value Of Health Care Information Exchange And Interoperability. volume 24, page 569. Health Affairs, 2005.

[36] B. Wangler, R.M. Åhlfeldt, and E. Perjons. Process oriented information systems architectures in healthcare. volume 9, page 253. SAGE Publications, 2003.