Computing
DOI 10.1007/s00607-012-0239-3

Process interoperability in healthcare systems with dynamic semantic web services

Wajahat Ali Khan · Maqbool Hussain · Khalid Latif · Muhammad Afzal · Farooq Ahmad · Sungyoung Lee

Received: 17 July 2012 / Accepted: 8 November 2012

Abstract Healthcare systems are very complex due to extreme heterogeneity in their data and processes. Researchers and practitioner need to make systems interoperable and integrate for the benefit of all the stakeholders including hospitals, clinicians, medical support staff, and patients. The broader goal of interoperability can only be achieved when standards are practiced.Two different healthcare systems can earn HL7 conformance and compliance but at the same time can be incompatible for interoperability because of varying implementation of HL7 interaction model. This is mainly because workflows in healthcare systems are very complex. Interoperability on one

W. A. Khan · M. Hussain · M. Afzal · S. Lee (✉)
Department of Computer Engineering, Kyung Hee University,
Seocheon-dong, Giheung-gu, Yongin-si, Gyeonggi-do 446-701,
Republic of Korea
e-mail: sylee@oslab.khu.ac.kr

W. A. Khan
e-mail: wajahat.alikhan@oslab.khu.ac.kr

M. Hussain
e-mail: maqbool.hussain@oslab.khu.ac.kr

M. Afzal
e-mail: muhammad.afzal@oslab.khu.ac.kr

K. Latif · F. Ahmad
School of Electrical Engineering and Computer Science,
National University of Sciences and Technology (NUST),
Islamabad 44000, Pakistan
e-mail: khalid.latif@seecs.edu.pk

F. Ahmad
e-mail: farooq.ahmad@seecs.edu.pk

Published online: 13 February 2013

hand requires flexible mechanism for the mapping of business processes to a standard, HL7 in our example. On the other hand it requires deeper understanding of the standard interaction model and gaps created by their incompatible implementations. In this paper we propose a novel technique of dynamically creating semantic web services as overlay on top of the existing services. We used Web Service Modeling Framework as an underlying architecture for HL7 process artifacts implementation as semantic web services. These semantic services are mapped to our proposed interaction ontology. Integrated reasoning mechanism provides necessary execution semantics for more effective and seamless end-to-end communication.The prototype we tested on different processes from the laboratory domain at a local diagnostic laboratory with uninterrupted process flow. The scenario of Result Query Placer interaction flow and its associated process artifacts are executed for the proof of concept.The proposed solution complements the existing data interoperability in HL7 and leads to semantic process interoperability. The achievement of semantic interoperability results in timely delivery of healthcare services to patients saving precious lives.

Keywords Interoperability · Ontology · Workflow · HL7 · Web Services

Mathematics Subject Classification 68Q55 [Computer Science]: Theory of Computing: Semantics

1 Introduction

Semantic web is making significant contribution in healthcare and life sciences [1]. Semantic technologies can help in developing such healthcare systems that can provide timely, reliable and cost effective services for managing clinical knowledge and improved patient care. The main objective of integrating semantic technologies in healthcare systems is to achieve interoperability during information exchange such as exchanging diagnosis data. Improved delivery of services with optimum interoperability can bring highest standard of accuracy and effectiveness. Semantic technologies such as semantic web services can effectively and efficiently cater these requirements for achieving interoperability.

Semantic interoperability is the ability to provide common understanding of processes and data exchanged between communicating systems. It can be seen from two perspectives; data interoperability and process interoperability. Data interoperability is related to the correct interpretation and understanding of the information exchanged between healthcare systems. Process interoperability, on the other hand, ensures seamless communication between different healthcare systems by developing shared understanding of their process artifacts. Most of the healthcare systems do not follow standards and thus lack semantic data as well as process interoperability. Health Level Seven (HL7) is a very prominent standard for the communication of medical records between healthcare systems. There are two commonly used variants of HL7 – V2 and V3 [2]. HL7 V3 supports data interoperability, though process interoperability is still a grey area [3]. Two different healthcare systems can earn HL7 conformance and compliance but at the same time can be incompatible for integration because of varying implementation of HL7 interaction model. Such healthcare systems can introduce delay or errors in medical information exchange. There is a pressing need for bringing process interoperability in healthcare systems for timely delivery of services and better health provision to the patients [4].

Semantic technologies can help in bringing process interoperability in healthcare systems [5]. Our hypothesis is that process heterogeneity can be mitigated by bringing semantics in the provision of process artifacts, more specifically web services. We have used Web Service Modeling Framework (WSMF [6]) in our proof of concept prototype. The process interoperability challenge is unearthed by developing Interaction Ontology which depicts the HL7 process artifacts in a coherent structure aligned with Web Service Modeling Ontology (WSMO) artifacts. The realization of the process interoperability in the proposed system is achieved through semantic web services inferring to locate an appropriate target interaction.

The proposed system is based on accuracy and scalability aspects with strong emphasis on the importance of semantic process interoperability. These evaluation metrics are validated using formal specifications to perform process mediation.

1.1 HL7 and semantic process interoperability

Healthcare workflows are more complicated than industrial workflows as they are non-linear, multi directional, interrupt driven and have unlimited complexity [7]. Healthcare workflows can be categorized into: *Administrative*, related to patients and healthcare organizations; *Financial*, cover financial aspects in a healthcare organization; *Clinical*, manage the operational and therapeutic decisions related to clinic; and *Laboratory*, manage the data for diagnosis [8]. For brevity, in this paper we will only discuss laboratory domain workflows for HL7 V3 compliant healthcare organizations.

HL7 standard is composed of different reference models. Importance of these models can not be denied in bringing semantic interoperability. These models includes *Use Case Model*, *Information Model*, *Interaction Model* and *Message Description Model*. Interaction and Information Models are directly related to the healthcare processes. Information model is mainly responsible for handling the contents in HL7 messages. Interaction model, on the other hand, covers process artifacts for handling the overall communication of HL7 messages [9]. Process artifacts are responsible for handling the behavioral aspects in HL7 compatible processes. These include *application roles, interactions, trigger events and message types*. *Application roles* are the logical components used for the communication of messages between the sender and receiver. *Interactions*, initiated by *trigger events*, define a flow of information between communicating parties. *Messages Type* contains information to be shared with other parties and certain patterns are predefined in the reference model. Figure 1 depicts the process artifacts in HL7 V3 and their relationship with each other. As mentioned earlier, only the process artifacts of the laboratory domain are considered in the presented case study.

The communication process is started by a trigger event and certain interaction is initiated by the sender. Each interaction is associated with a message type and takes place between specific sending and receiving application roles. There are three basic

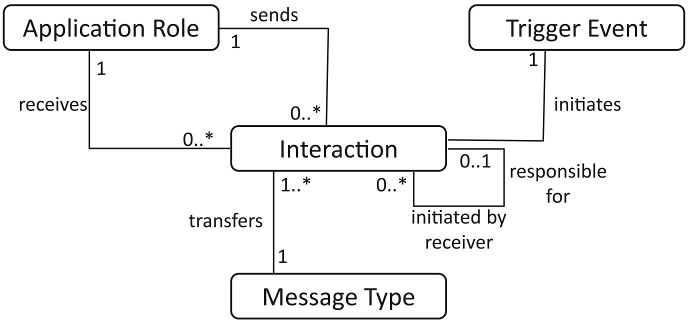

Fig. 1 Interaction model

types of interactions in HL7 laboratory domain: *Order*, *Promise* and *Result* that aligns to request, acknowledgement and response in traditional communication systems. The communicating parties, depending on the requirements of the workflow, can pick any specific interaction associated to one of the three basic types.

Workflows involving flow of information between laboratory domain process artifacts are very complex. Figure 2 depicts the sequence of interactions that may take place among different application roles. Each rectangle shows the interaction and its sending and receiving application role. The initial interaction *Order Fulfilment Request* is communicated by Order Placer (OP) application role behaving as Sender (S) with Receiver (R) application role called Order Filler (OF). OF has three possibilities to respond to *Order Fulfillment Request* interaction based on the conformance criteria of an organization. The possible interactions are *Promise Activate, Promise Cancel and Order Confirm* from OF to OP application roles. *Promise Activate* interaction response is performed by OF, if it is sure of accomplishing the request, otherwise, response would be in the form of *Promise Cancel* interaction. Likewise, OF can directly confirm the request with *Order Confirm* interaction rather than initially doing promise interactions. These interactions further leads to other possible interaction completing the sequence of interaction in laboratory domain as shown in Fig. 2. Healthcare organizations can localize and adapt these workflows as per their requirements as long as the holistic scheme remains in conformance with the overall template of process artifacts. For instance, some organizations may follow all three categories of interactions while others may prefer to omit *Promise* interactions from the workflows, resulting in process heterogeneity.

Heterogeneity issue in workflows of two different healthcare systems is further elaborated in Fig. 3. This scenario assumes that two organizations have implemented different set of interactions. Source organization supports interactions related to all the three categories (i.e. order, promise and result) from HL7 laboratory domain. Receiver organization only supports order and result categories of the interactions. The two application roles (AR) that are involved in performing these interactions are *Order Placer (OP)* and *Order Filler (OF)*. The steps involved in the process to handle heterogeneity are as follows.

Step 1: Source organization interacts with the receiver organization using Order Fulfillment Request interaction communicated by Order Placer AR to

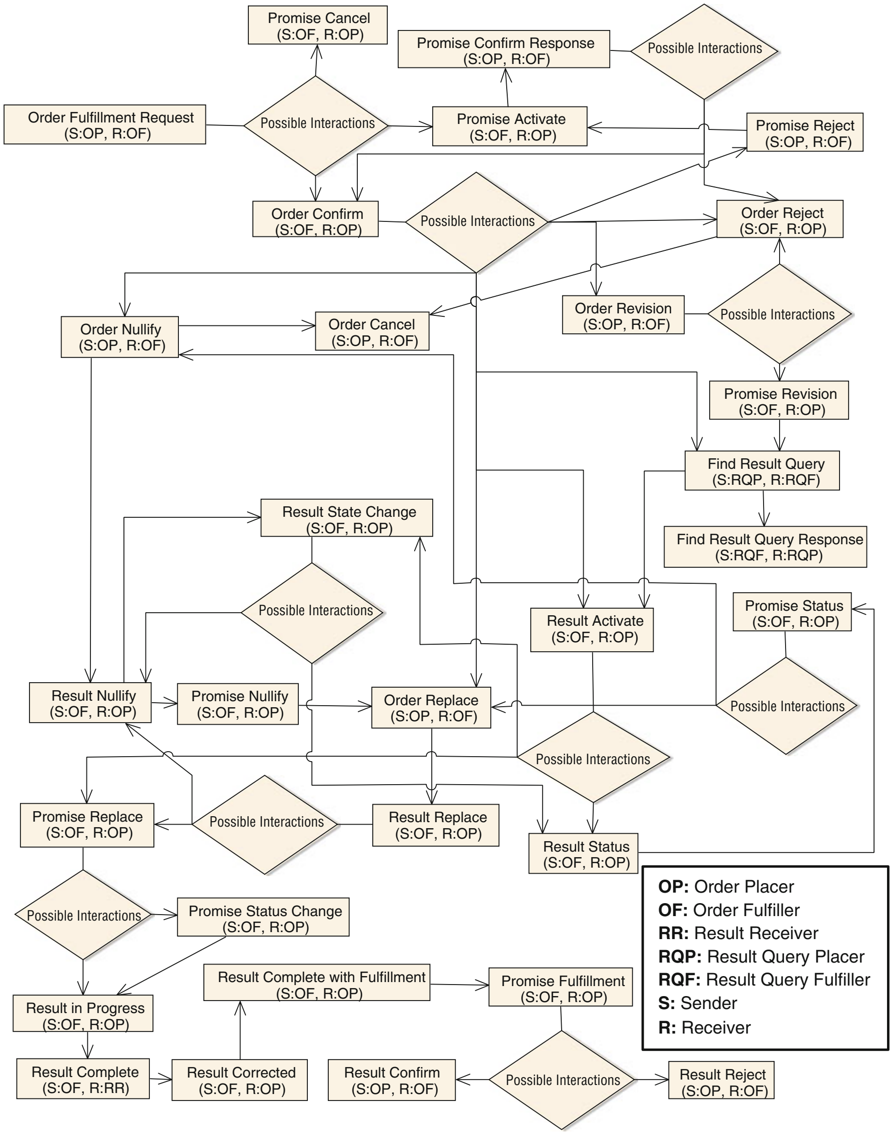

Fig. 2 HL7 laboratory domain interactions workflow

Order Fulfiller AR. Message Type (MT) Placer Order is associated with Order Fulfillment Request interaction and Order Activate Trigger Event (TE) is used for initiating this interaction.

Step 2: Source organization is expecting Promise Activate interaction from the Order Fulfiller AR of the receiver organization to the Order Placer AR of the source organization. It is expecting this interaction because it is part of the workflow of this organization while in workflow of receiver organization; the response

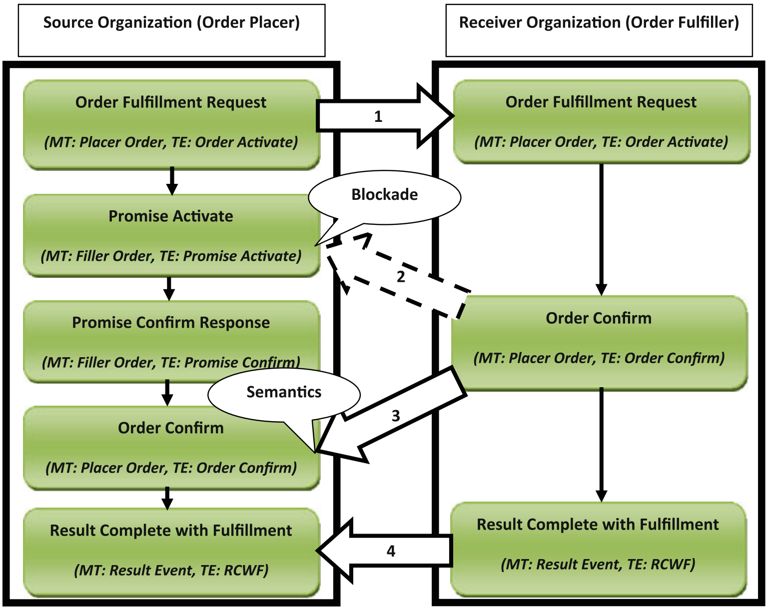

Fig. 3 Workflow heterogeneity in healthcare systems

for Order Fulfillment Request is Order Confirm. Blockade will take place with the system having no semantics. Therefore no further communication will take place until the deadlock is resolved.

Step 3: Semantics can help the system to resolve the deadlock by matching Order Confirm interaction in source organization workflow. Since interaction exists in its workflow therefore it accepts this interaction and waits for Result Complete with Fulfillment interaction.

Step 4: Receiver organization interacts with source organization using interaction Result Complete with Fulfillment as described in its workflow. It uses the MT and TE of Result Event and Result Complete with Fulfillment respectively.

Process Mediation described in above mentioned scenario is related to conformance of organizations to set of interactions based on their requirements. Mediation process is not limited to optional and necessary interactions but also includes acknowledgements as well. Transmission Infrastructure of HL7 V3 standard defines two levels of acknowledgements: *Accept Level Acknowledgement* and *Application Level Acknowledgement* [10]. The decision of whether initiating systems requires *Accept Level Acknowledgement* or not, can be determined by examining HL7 transmission wrapper. *Accept Level Acknowledgement* can be responded by receiving system in any one of the three types: *Commit Accept (CA)* (message accepted for processing), *Commit Reject (CR)* (some values are not acceptable to receiving application) and *Commit Error (CE)* (message cannot be accepted for any reason) [10]. Process mediation is necessary when transmission wrapper suggests one of the form of accept level acknowledgement but the responding system doesn't respond accordingly. In the same way, *Application Level Acknowledgement* is a functional response message passed by receiving system to the sending system. If HL7 transmission wrapper indicates this acknowledgement, then an immediate response should be returned. Also, in scenarios, an *Application Level Acknowledgement* can also require *Accept Level Acknowledgement*. Therefore, not following the rules specified in HL7 transmission wrapper while communication can lead to unnecessary delay or even process termination, thus requiring process mediation for seamless communication.

Two healthcare organizations can only communicate seamlessly provided they are, first and far most important, HL7 compliant, and that they understand the intricacies and heterogeneity in the workflows. The proposed system achieve process interoperability using Transmission Ontology and Interaction Ontology which are consumed by semantic web services. Semantic web services emplace a strong foundation for making HL7 based systems interoperable. Semantic web services can reduce the manual effort required to develop service-oriented applications by enabling machines to understand the functions and interfaces of Web services through semantic annotations [11].

The process artifacts information of HL7 are incorporated using WSMO for seamless communication of information to handle process interoperability. The proposed system achieves semantic process interoperability by integrating WSMF and HLH (Health Life Horizon)[1] architectures. We designed Interaction ontology for resolving heterogeneities among workflows of HL7 compliant healthcare sytems while communication. Semantic web services of HL7 process artifacts are built on the underlying WSMF framework to achieve goal of semantic process interoperability.

2 Related work

2.1 Approaches to semantic web services

Web Service Modeling Framework (WSMF) is a semantic web service based approach that works on two main principles strong decoupling and strong mediation service. It is based on two frameworks Semantic Web enabled Web Services (SWWS) and Web Service Modeling Ontology (WSMO). It focuses on bringing automation in the process by automatically discovering and composition of the web services [6].

OWL-S (Semantic Markup for Web Services) as a framework provides upper ontology for description and reasoning of web services. It has three prominent parts: service *profile* is used for advertising and discovering services; *process models* give detailed description of operations; and *groundings* provide interoperability details. The main goal of OWL-S is to provide automation in the discovery, invocation, composition, and interaction with the web service [12,13].

Internet Reasoning System (IRS) is a semantic web service based framework that is used by applications to bring semantics for the description and execution of web services. Different components of IRS II [14] are server, publisher and client. These

[1] http://hl7.seecs.nust.edu.pk/.

components communicate through SOAP protocol. The underlying framework on which IRS II is based on is Unified Problem Solving Method Development Language (UPML) and is also used for storing knowledge description [12]. The latest version of IRS II is IRS III that is based on WSMO specifications. It mainly uses the WSMO orchestration aspects for bringing automatic discovery and composition [15].

METEOR-S [16,17] and SWSF (Semantic Web Services Framework) [18] are other approaches that are commonly used for semantic web services. METEOR-S is more related to WSMO whereas SWSF is more close to OWL-S.

WSMO is considered appropriate as compare to OWL-S because of handling heterogeneity in a more comprehensive way. OWL-S provides no explicit distinction between choreography and orchestration and thus have no internal mechanism to manage workflows of different processes. Therefore OWL-S is always dependent on external work for defining workflow of process. Due to this OWL-S only supports one way to interact with service as Service Model which is defined per service. WSMO provides multiple interfaces to interact with service therefore handling choreography and orchestration effectively. Service provider and service requestor are not separated in OWL-S thus not supporting high level of the degree of integration of functionalities. On the other hand in WSMO the service requestor is handled through WSMO Goal while the service provider is handled by WSMO web service. WSMO also handles data and process heterogeneity with the help of Mediator, a top level entity of WSMO handling heterogeneity. OWL-S has no concept like mediator for handling heterogeneity [19].

2.2 Healthcare projects based on semantic web services

There are many healthcare projects that are based on semantic web services. Some of the projects are explained to throw light on the importance of semantic web services in the web service discovery and composition for bringing automation in the system.

COCOON is a web services based project aimed at reducing medical errors. This project focusses on resolving the problem of integration in healthcare domain. The problem of integrating components from service discovery to service composition is discussed. It is a WSMO compliant project and uses WSMO compliant service discovery engine for resolving the service discovery issue. In COCOON the most appropriate services are discovered to be used by the specialist hence providing better healthcare services [20].

Artemis [21] is another project based on semantic web services for the semantic discovery and composition of services. It uses OWL -S as the approach for implementing semantic web services and uses HL7 as a standard for communication. Artemis uses OWL mapping tool (OWLmt) for the communication between sender and receiver providing semantic interoperability. OWLmt works as a mediator between sender and receiver by comparing sender ontology instances and receiver ontology instances with each other for making possible the communication [20]. The primary focus of Artemis project is on data interoperability aspect by resolving heterogeneities between HL7 standards V2 and V3.

Integrating the Healthcare Enterprise (IHE)[2] is an initiative by healthcare professionals and industry to coordinate implementation of standards for healthcare systems integration [22]. IHE Integration profiles addresses interoperability assessment criteria based on interoperability at interface level, semantic level, legal and organizational level, and, security level [23]. IHE profiles makes significant contribution towards interoperability among healthcare systems using different standards, still posses limitations mainly related to conformance of organizations to specific aspects based on their needs [24]. Therefore, process mediation is required when specifications lacks definition of these aspects of interoperability.

Plug and Play Electronic Patient Records (PPEPR) is a semantic SOA based platform with the objective of integrating heterogeneous Electronic Patient Records (EPRs). The integration in PPEPR project is based on SOA, web services and semantics. The initial prototype of PPEPR tackles heterogeneity between two types of HL7 standard HL7 V2 to HL7 V3 [25]. They also propose mappings approach based on ontologies of heterogeneous HL7 standards [26].

Health Life Horizon (HLH)[3] is our baseline implementation of HL7 messaging. It covers all details of message generation, storage, parsing, transportation and database mappings. HLH architecture is composed of three main components [27]: *Data Mapper* component, used for the generation of mapping specification [28]; *Core Engine*, used for HL7 message generation and parsing and is also responsible for handling semantic annotations with its *Ontology Core* [3] component; and *Transportation* component, used for communicating message from sender to the receiver. Different communication protocols such as Minimal Lower Layer Protocol (MLLP), SOAP or even ebXML can be used for message exchange [27].

Web Services are enabling applications to communicate with each other and thus help in reducing time and cost, related to web applications. Semantics provided to web services, will lead to automation in the service discovery and composition. This results in the use of semantic web services for which different approaches are currently followed.

3 Formal ontology modeling for process mediation

In order to achieve process interoperability there is a need to align the workflow with standardized communication patterns. These patterns and alignments can be handled with the help of ontologies more appropriately. Figure 3 in Sect. 1.1 describe the heterogeneity issues among healthcare organizations compliant to the same standard with different processes implementation methodologies. Also, resolving these heterogeneities requires semantics as process mediation for the communication of information between healthcare organizations. These semantics are provided by ontologies

[2] http://www.ihe.net/.

[3] Health Life Horizon Project aim was to carry out research in healthcare by developing Health Level 7-based software framework in order to provide health services for diverse communities of the world. It was supported by National ICT R&D Fund, Pakistan under reference number ICTRDF/TR&D/2008/47.

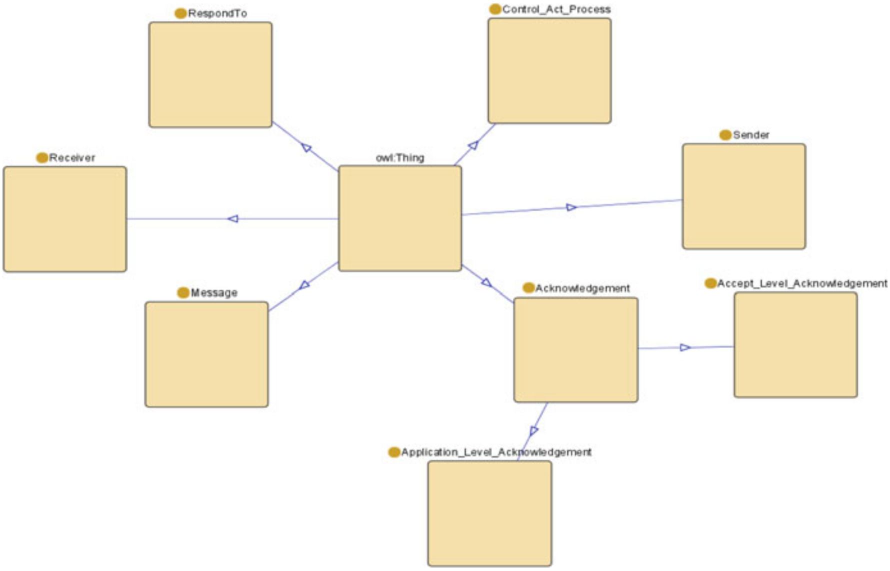

Fig. 4 Transmission ontology

based on standard processes specification, in our case HL7 V3 interaction model. Therefore, we designed and developed *Transmission* and *Interaction Ontology* to support standardized communication patterns. Communication patterns in HL7 are based on process artifacts, the information of these process artifacts is depicted in *Interaction Ontology*.

3.1 Transmission ontology

Transmission Ontology stores the information related to the message to be transferred. Its information is based on the HL7 Message as shown in Fig. 4. HL7 Message is divided into three categories Message Payload, Control Act Wrapper and Transmission Wrapper. It helps in extracting the information related to the process artifact in the message which is then used by *Interaction Ontology*.

The main classes of *Transmission Ontology* are *Message, Sender, Receiver, Control Act Process, Acknowledgement and RespondTo*. *Sender and Receiver* classes handles the message communication between these entities. All the HL7 messages comes under class Message. *Control Act Process* class represents trigger event of the message. In response to every message, application level or accept level acknowledgement that comes under *Acknowledgement* class. *Transmission Ontology* consumes message transmission infrastructure having information about interaction and trigger event. This information is required by *Interaction Ontology* to handle the flow of interactions. In order to handle workflow of the initiated trigger event, interaction ontology consumes related information of transmission ontology.

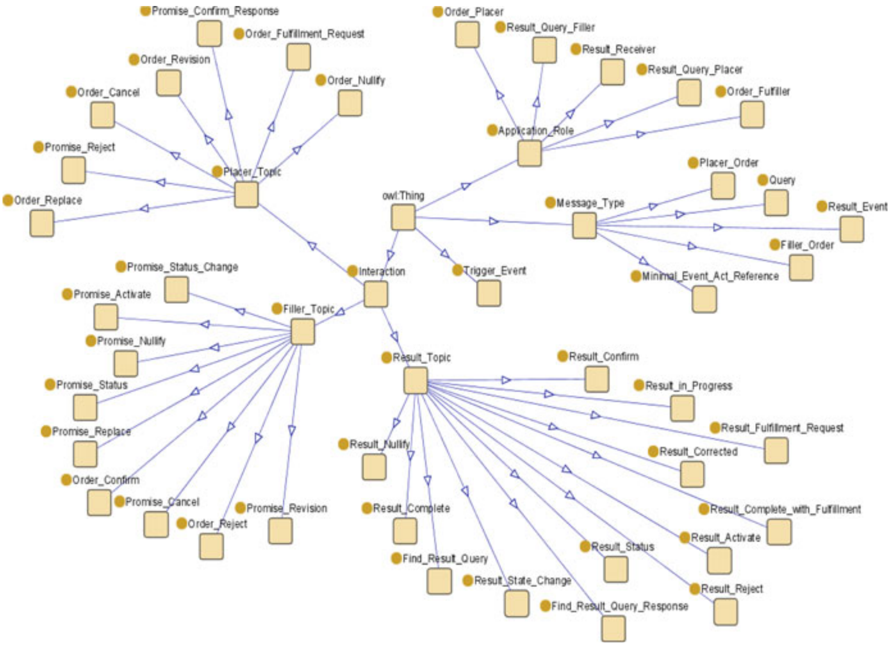

Fig. 5 Interaction ontology

3.2 Interaction ontology

Interaction Ontology contains the information related to HL7 V3 process artifacts and their relationship with each other. The main classes in this ontology are the *Application Roles, Interactions, Trigger Events and Message Types* as shown in Fig. 5.

The application roles in HL7 V3 laboratory domain are then further categorized into different subtypes such as: *Order Placer, Order Fulfiller, Result Query Placer, Result Query Filler and Result Receiver*. The interactions are also further divided into three subcategories: *Order, Promise and Result*, each having own category of sub interactions. Each interaction is initiated by a trigger event and its information is stored by *Trigger Event* class. The *Message Type* class contains the different message types that are required for the information to be stored in the message. The interactions are also categorized by message types. These include: *Minimal Event Act Reference, Query, Placer Order, Filler Order and Result Event*. *Interaction Ontology* also consists of restrictions on the classes. Scope of certain properties is restricted at the lower class hierarchy level. Table 1 shows the restrictions on the process artifacts. These restrictions show the relationship of each of the process artifacts with others and how they work.

4 Methodology

This section explains the proposed architecture of the system and also case study of laboratory domain to show the sequence of activities between components.

Table 1 Restrictions on classes and properties

Class/Property	SubClass	Restriction
Application Role/ playsRoleIn	Order Placer & Order Fulfiller	– ∀ playsRoleIn only (Result_Reject or Result_Status or Result_in_Progress or Result_Complete_with_Fulfillment or Result_Corrected or Result_Nullify or Result_Confirm) – ∀ playsRoleIn only (Order_Fulfillment_Request or Order_Revision or Order_Cancel or Order_Nullify or Order_Replace or Promise_Reject) – ∀ playsRoleIn only (Order_Reject or Order_Confirm or Promise_Activate or Promise_Revision or Promise_Cancel or Promise_Replace or Promise_Nullify or Promise_Nullify or Promise_Status_Change or Promise_Status)
	Result Query Placer & Result Query Filler	– ∀ playsRoleIn only (Find_Result_Query or Find_Result_Query_Response)
Interaction/ senderRole & receiverRole	Result Status, Result Nullify, Result in Progress,Result Corrected, Result Confirm, Result Complete With Fulfillment	– ∃ receiverRole some Order_Placer – ∃ senderRole some Order_Fulfiller
	Result Reject	– ∃ receiverRole some Order_Fulfiller – ∃ senderRole some Order_Placer
	Result State Change, Result Activate, Result Complete	– ∃ receiverRole some Result_Receiver – ∃ senderRole some Order_Fulfiller
	Find Result Query Response	– ∃ receiverRole some Result_Query_Placer – ∃ senderRole some Result_Query_Filler
	Find Result Query	– ∃ receiverRole some Result_Query_Filler – ∃ senderRole some Result_Query_Placer
	Placer Topic	– ∀ receiverRole only Order_Fulfiller – ∀ senderRole only Order_Placer
	Filler Topic	– ∀ receiverRole only Order_Placer – ∀ senderRole only Order_Fulfiller
Message Type/ transferedBy	Minimal Event Act Reference	– ∀ transferedBy only (Result_Confirm or Result_Reject)
	Query	– ∀ transferedBy only Find_Result_Query
	Result Event	– ∀ transferedBy only (Result_Status or Result_State_Change or Result_Nullify or Result_in_Progress or Result_Corrected or Result_Complete_with_Fulfillment or Result_Complete or Result_Activate or Find_Result_Query_Response)
	Placer Order	– ∀ transferedBy only (Order_Fulfillment_Request or Order_Revision or Order_Cancel or Order_Nullify or Order_Replace or Order_Reject or Order_Confirm)
	Filler Order	– ∀ transferedBy only (Promise_Reject or Promise_Confirm_Response or Promise_Activate or Promise_Revision or Promise_Cancel or Promise_Replace or Promise_Nullify or Promise_Status_Change or Promise_Status or Promise_Status)

4.1 Proposed architecture

In order to achieve semantic process interoperability, we have integrated HLH Studio and WSMO architectures. The architecture specified in Fig. 6 is the abstract model of the system. The *MessageGenerator* component generates HL7 V3 messages using JavaSIG API. The generated message is then used by Web Service Execution Environment (WSMX) server for further processing. The *MessageParser* component validates HL7 message. The *Adapter* component transforms HL7 message into WSML form. WSMX process the converted HL7 message as goal for the discovery and invocation of web services. Client communicates with *HLH Manager* and *Adapter Manager* components for message generation and WSML format transformation respectively. *WSMX Manager* controls the information flow between external and internal components. *Parser* validates the WSML description files. Message Parser component is related to parsing of HL7 Message in XML form while the Parser component in WSMX is related to the Parsing of WSML files. *Discovery* component is responsible for the discovery of web services on the basis of *goals*. It matches the capability of the goal with

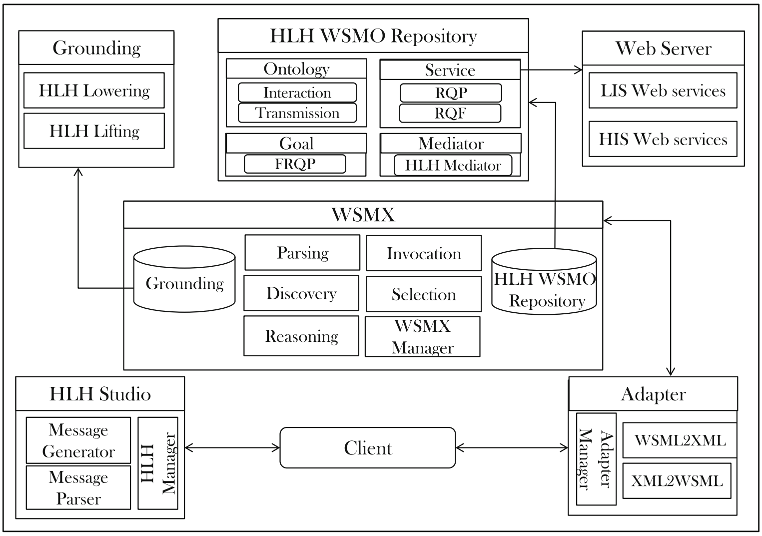

Fig. 6 Proposed system architecture

the web services and rank different services for selection. *Selection* component selects the most appropriate web service for invocation. *Invocation* module is responsible for actual invocation of the web service for the transfer of the message. Choreography of the semantic web service contains state signatures which has grounding to the end point web service to be invoked.

The WSMO entities based on HL7 are stored in the *HLH WSMO Repository*. Ontologies include *Interaction and Transmission Ontology*; Web Services component contains the *Result Query Placer and Result Query Filler* web services; Goal component contains, for example, *Find Result Query Placer* goal. The end point laboratory web services are deployed on a web server. *Grounding* component is responsible for the conversion from Simple Object Access Protocol (SOAP) to WSML and vice versa. There are two components that are responsible for the conversion: *HLH Lifting and HLH Lowering*. Both lowering and lifting is performed using XSLT transformation. *HLH Lowering* is done by converting SOAP request into Resource Description Framework (RDF) and then to WSML for further processing by WSMX. *HLH Lifting* transforms WSML to RDF and then to SOAP Response. The proposed solution is flexible enough to absorb workflow/process following HL7 standard with very minimal changes.

4.2 Laboratory domain case study

WSMO entity, Goal initiates the process of service invocation. Client only provides his desire in the form of Goal. This goal is dependent on ontologies for service discovery and invocation seamlessly. An abstract flow of information for seamless communication is depicted in Fig. 7. The figure describes client providing his/her desire of finding a result query. Goal discovery based on this desire leads to discovery of Find

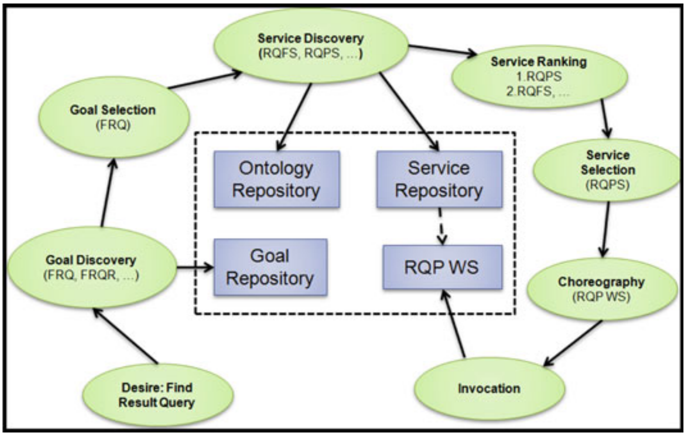

Fig. 7 Seamless communication with goals

Result Query (FRQ) and Find Result Query Response (FRQP) goals. Selection of FRQ goal is made due to more close relationship with client's desire. Service discovery is performed based on FRQ goal using Ontology and Service repositories. Result Query Placer (RQP) service is selected among the list and finally invoked for communication of HL7 message. This whole scenario is further elaborated in four steps in this section.

There are different modeling and testing tools that are used for the implementation of the proposed system. We used Web Service Modeling Toolkit (WSMT) for modeling WSMO entities and SOAP UI tool for testing end point web services and WSMX entrypoints.

Also, we used Enterprise Architect[4] for design of the proposed system. The process is shown with the help of comprehensive sequence diagram, logically divided into four sections for clarity. These sections show the design and implementation of proposed system components with the sequence of information flow explained below:

4.2.1 HL7 message generation and conversion

Initially, client uses *HLHStudio* for generation of HL7 message. We suppose that client wants to inquire about status of test result of a patient. Therefore, *HLH Manager* uses *Message Generator* for generating HL7 test result query message. Client now provides this message to the proposed system for sending the message using appropriate web service for communicating information. *Adapter Manager* converts test result query message to WSML format for the WSMX to understand. The flow of information for

[4] http://www.sparxsystems.com/.

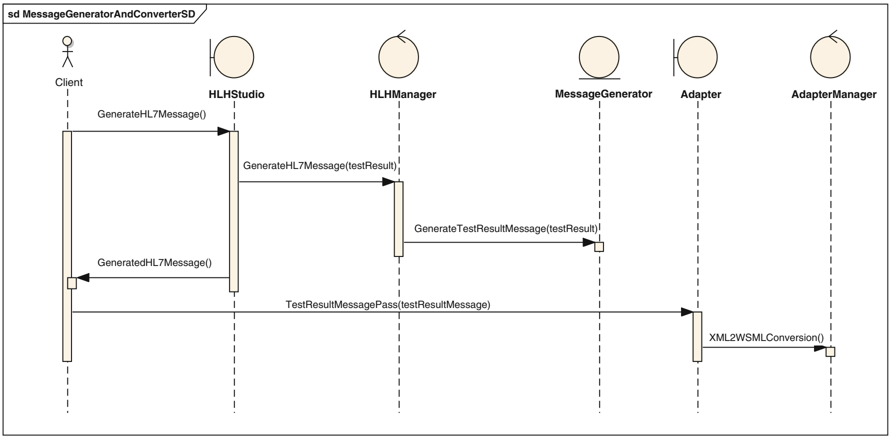

Fig. 8 HL7 message generation and conversion

generating and converting message is shown in Fig. 8. The sequence diagram continues in the next steps to show process for invocation of appropriate web service.

4.2.2 Goal discovery and selection

Adapter Manager provides WSML format of HL7 message to *WSMX Manager* which uses *Discovery* for finding the appropriate goals from *HLH WSMO Goal Repository* and *HLH WSMO Ontology Repository*. Finally, from the appropriate goals discovered, most suitable one, Find Result Query (FRQ) is selected by *Selection* and provided to *WSMX Manager*. FRQ goal uses semantic information to discover, rank and select appropriate service. Achieve Goal entrypoint representing FRQ Goal as input is shown as follows:

```
<se:Envelope xmlns:se="http://schemas.xmlsoap.org/soap/envelope/"
                       xmlns:ws="http://webservices.deri.ie">
  <se:Body>
    <web:achieveGoal>
      <ws:wsmlMessageGoal>http://seecs.hlh.edu.pk/goals#FRQGoal
                       </ws:wsmlMessageGoal>
    </ws:achieveGoal>
  </se:Body>
</se:Envelope>
```

Figure 9 shows the goal discovery and selection process in sequence diagram. Next step describes the mediation and service selection process after goal discovery.

4.2.3 Mediation and service selection

WSMX Manager forwards FRQ goal to *Discovery* to use *HLH WSMO Service Repository* and *HLH WSMO Ontology Repository* for finding candidate services. Suitable service is selected by *Selection* and conformance is validated for the interaction

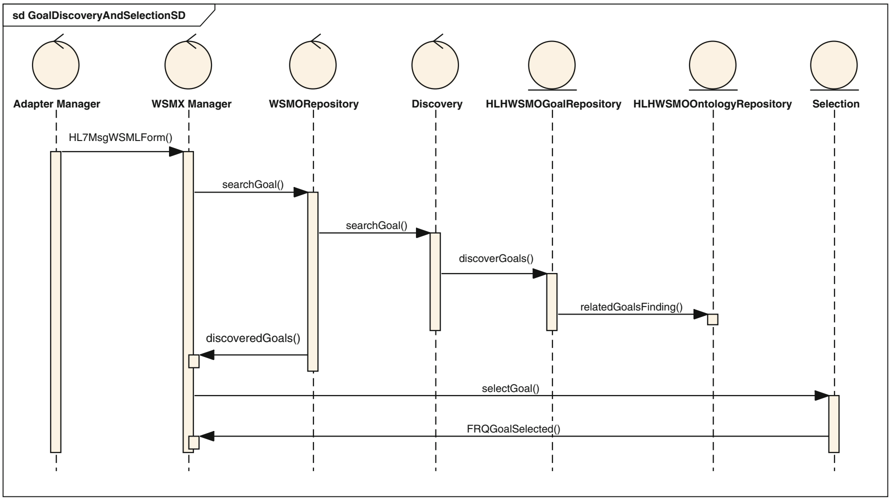

Fig. 9 Goal discovery and selection

to be performed. We suppose that organizations have conformed to set of interactions for application roles (web services in this case), and also we suppose that process mediation is required for completion of our scenario showing heterogeneity in processes. Therefore, if service is suppose to perform a particular interaction and the target organization is not conformed to that interaction, mediation is required to resolve this heterogeneity. *HLH WSMO Mediator Repository* resolves this heterogeneity by using *HLH Mediator* to find appropriate alternate interactions of FRQ goal by verification from *Interaction Ontology*. For example, *Order Placer* web service is required to perform *Promise Activate* interaction, to which receiver has not conformed, *HLH Mediator* uses *Interaction Ontology* to find alternate interactions for the service to execute i.e *Order Confirm* (shown in Fig. 2; Table 1).

We have created so far Result Query Placer (RQP) and Result Query Filler (RQF) semantic web services for our scenario. The application role concept in HL7 V3 developed as semantic web services. RQP is an application role that originates queries for results. It is capable of receiving query response as well without storing the data. On the other hand, RQF is an application role responsible for satisfying request of RQP. Both can perform sending and receiving application roles responsibilities.

The conformance of the system is based on implementation of various application roles. So keeping in view this definition and following HL7 web service basic profile, there is one to one correspondence of application role and web service [29]. The proposed system implements application roles as semantic web services. Result Query Placer (RQP), for instance, is an application role that is implemented as semantic web service in the proposed system and is used for querying the status of the result. *Result Query Placer* web service is shown as follows:

```
wsmlVariant _"http://www.wsmo.org/wsml/wsml-syntax/wsml-rule"
namespace { _"http://www.seecs.hlh.edu.pk/services#",
    transmission _"http://www.seecs.hlh.edu.pk/transmission#",
    dc _"http://purl.org/dc/elements/1.1/",
    interaction _"http://www.seecs.hlh.edu.pk/inter#",
    discovery _"http://wiki.wsmx.org/index.php?title=DiscoveryOntology#",
    wsml _"http://www.wsmo.org/wsml/wsml-syntax#" }
webService ResultQueryPlacer importsOntology interaction#inter
capability ResultQueryPlacerCapability
    nonFunctionalProperties
      discovery#discoveryStrategy hasValue discovery#HeavyweightDiscovery
        dc#description hasValue "Result Query Placer interaction with
        message payload, transmission and control act wrapper"
    endNonFunctionalProperties
    precondition RQPPre
        definedBy
        ?message memberOf transmission#Message
        and ?message[transmission#interactionID hasValue "Find Result Query"]
        and transmission#ControlActProcess(?message, ?controlactprocess).
    postcondition RQPPost
        definedBy
        ?message memberOf transmission#ApplicationLevelAck
        and ?message[transmission#interactionID hasValue
        "Find Result Query Response"] and
        transmission#ControlActProcess(?message, ?controlactprocess).
interface RQPInterface
 choreography RQPChoreography
   stateSignature RQPstateSignature
       importsOntology interaction#inter
          in concept interaction#Find_Result_Query withGrounding
          _"http://localhost:8080/RQPSimpleTesting/
          RQPSimpleTestWSService?wsdl#wsdl.interfaceMessageReference
          (RQPSimpleTestWSPortType/RQPSimpleTestWSOperation/in0)"
    .........................
```

Process mediation for selecting alternate service in case of non availability of first choice service selected is shown in Fig. 10. The sequence diagram shows the selection process of web service based on goal selected and also process mediation when the receiving organization doesn't support first choice interaction to be carried out by the service. An alternate interaction for the selected web service is found by *HLH Mediator* using *HLH WSMO Ontology Repository*. After mediation, the information requires transformation for end point web service invocation as it is still in WSMX native format (WSML). *Grounding* is required for transformations and is described in the continued sequence diagram in next step i.e. grounding and service invocation.

4.2.4 Grounding and service invocation

WSMX Manager forwards selected RQP semantic web service to *Grounding* for conversion from WSML to SOAP by *HLH Lifting*. RQP service is invoked from web server and HL7 test result query message is communicated. HL7 test result query response message is generated and sent back as a result by *Invocation* to *WSMX Manager* in the form of RQF service. *WSMX Manager* forwards SOAP information to *Grounding* that uses *HLH Lowering* for converting SOAP to WSML. Finally, the WSML format

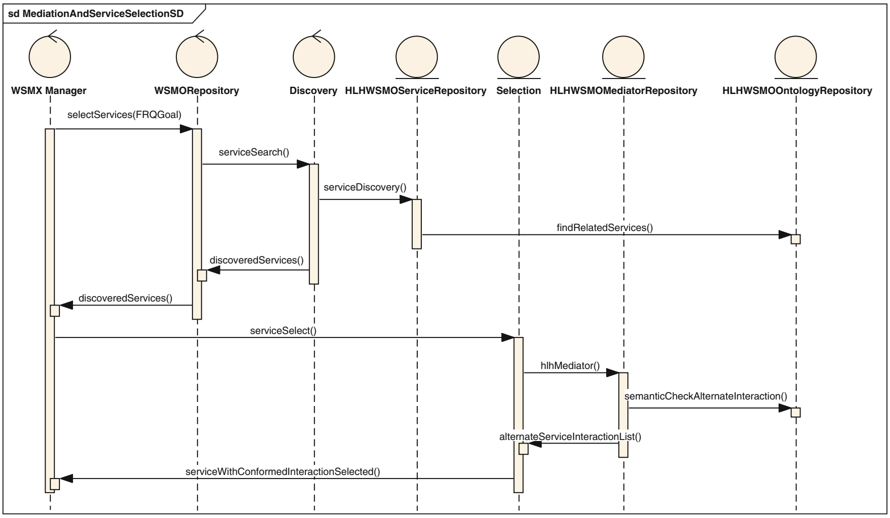

Fig. 10 Mediation and service selection

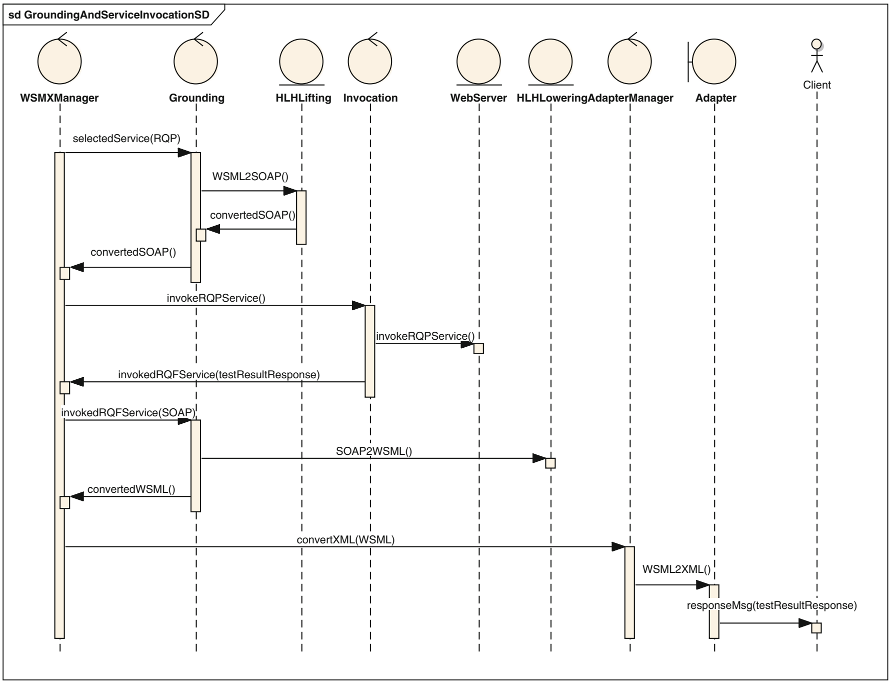

Fig. 11 Grounding and service invocation

is forwarded to *Adapter* for converting WSML to XML, and HL7 test result query response is provided to Client. Fig. 11 shows the process of grounding and service invocation in sequence diagram.

Table 2 Grounding—WSML and SOAP

WSML Message

```
wsmlVariant _"http://www.wsmo.org/
wsml/wsml-syntax/wsml-dl"
namespace { _"http://www.seecs.
hl7.edu.pk/transmission#"}
ontology Transmission
concept Message
     id ofType _integer
     interactionID ofType _string
concept Sender
     name ofType _string
     typeCode ofType _string
     telecom ofType _string
concept Receiver
     typeCode ofType _string
concept ControlActProcess
     classCode ofType _string
concept Ack
     typeCode ofType _string
     telecom ofType _string
concept RespondTo
     typeCode ofType _string
     telecom ofType _string
...
instance message memberOf Message
   interactionID hasValue "POLBIN35000UV"
instance receiver memberOf Receiver
   typeCode hasValue "RCV"
...
```

SOAP Message

```
<soapenv:Envelope xmlns:soapenv=
"http://schemas.xmlsoap.org/soap
/envelope/"
xmlns:rqp="http://j2ee.netbeans.org/
wsdl/RQPService">
   <soapenv:Header/>
   <soapenv:Body>
      <rqp:PlaceResultQuery>
       <part1>
         <Message xmlns="urn:hl7-org:v3"
  xmlns:xsi="http://www.w3.org/2001/
  XMLSchema-instance">
  ...
  <interactionId>
    POLBIN35000UV
  </interactionId>
  <acceptAckCode>
    <code>Ack</code>
  </acceptAckCode>
  <receiver>
    <typeCode>RCV</typeCode>
  </receiver>
  <sender>
    <typeCode>SND</typeCode>
  </sender>
  <controlActProcess>
    <classCode>
        <code>active</code>
    </classCode>
  </controlActProcess>
  ...
</soapenv:Envelope>
```

We used XSLT transformation lowering and lifting purpose. Table 2 shows part of HL7 Message in WSML format and SOAP Message generated using XSLT transformation by *HLH Lifting*. It shows the main constructs of the message that are represented in Transmission Ontology such as: Message, Sender, Receiver, Control Act Process and Acknowledgement. This message is communicated with receiver by web service after grounding is performed.

5 Evaluation

This section discuss the main evaluation aspects of the proposed system that are used for process mediation among healthcare systems. The evaluation metrics includes accuracy and scalability of the system.

5.1 Accuracy

Accuracy is one of the aspect for evaluation of the proposed system. It is determined by the seamless communication of information between healthcare systems. Organizational workflow compliancy is key for seamless communication and requires process

Table 3 Formal representation of concepts and their relationships

Concepts definition	Relationship
HL7_MESSAGE: Set of standardized messages format to represent clinical information	
INTERACTION: Set of flow of information that represents HL7_MESSAGE	HL7_MESSAGE ⤔ INTERACTION (Each HL7 message is associated with exactly one interaction)
APPLICATION_ROLE: Set of web services responsible for exchanging INTERACTION	INTERACTION $\leftrightarrow$ APPLICATION_ROLE (One interaction is associated with two application roles and one application role can support more than one interactions)
WORKFLOW: Set of workflows of sender and receiver that is based on INTERACTION	WORKFLOW $\leftrightarrow$ INTERACTION

mediation. Accuracy of the system is evaluated using formal methods in Sect. 6.2. Heterogeneity in workflows of organizations can result in process termination, thus requiring process mediation. Process Mediation is dependent on accuracy, that can be achieved when comparison of communicating organizations workflows is performed. The workflows can be totally mismatch resulting in no communication or there may be some level of commonalities that will result in messages exchange.

5.2 Scalability

Scalability is another important aspects of any system, addressed by our proposed system with the help of *Interaction Ontology*. Initially, it is only based on HL7 Laboratory domain process artifacts, however, other domains (e.g. Patient Administration) process artifacts can easily be added to the ontology without requirement of any compilation. It can be observed from three perspectives that are formally validated in Sect. 6.3. Firstly, addition of new application roles to HLH WSMO Repository and its relationship with existing interactions. Secondly, addition of new interactions and its relationship with existing application roles. Change can easily be accommodated with in interaction ontology, providing scalable solution.

6 Proposed system validation using formal specifications (Z notations)

Accuracy and scalability of the proposed system is validated using formal methods in this section. The main contributions for semantic process interoperability in the proposed system can be categorized into two steps: HL7 Message Generation (information to be communicated by web services) and Process Mediation using HLH WSMO Repository. The processing of information from HLH Studio takes place in WSMX that controls the whole process flow. HLH Studio handles the message generation and HLH WSMO Repository hold the process mediation information and the formal representation of concepts and their relationship are described in Table 3.

The concepts in Table 3 starts from the clinical information which is the information that client wants to exchange, but requires standard format provided in the form of HL7_FORMAT. Therefore, relation among them is represented by CLINICAL_INFORMATION $\nrightarrow$ HL7_MESSAGE. All clinical information cannot be represented in HL7 format, so the symbol "$\nrightarrow$" indicates partial mapping function. HL7_MESSAGE consist of INTERACTION that is used for the exchange of message and relationship between them is partial injection represented by "$\rightarrowtail$". Each INTERACTION is associated with sender and receiver APPLICATION_ROLE. On the other hand each APPLICATION_ROLE can be associated with many INTERACTIONs, so their relationship is binary represented by "$\leftrightarrow$". The main process artifacts in HL7 are interaction and application roles, that are stored in *HLH WSMO Repository* and their semantics are stored in *Interaction Ontology*. We will be considering *HLH WSMO Repository* for validating the accuracy and scalability of the proposed system in the following sub-sections.

6.1 Formal representation of HLH WSMO repository

Interaction Ontology handles the semantic aspects of HL7 process artifacts with each other. Interactions and application roles are the process artifacts that plays important role in process interoperability and seamless communication. As *Interaction Ontology* is part of HLH WSMO Repository, therefore the initial state schema of HLHWSMORepository_ProcessArtifatcs is as follows:

HLHWSMORepository_ProcessArtifacts

$$
\begin{aligned}
& inter : \mathbb{F}\, INTERACTION \\
& appRole : \mathbb{F}\, APPLICATION\_ROLE \\
& sendWS : INTERACTION \rightarrow APPLICATION\_ROLE \\
& receiveWS : INTERACTION \rightarrow APPLICATION\_ROLE \\
\hline
& \forall x \mid x \in INTERACTION \bullet sendWS(x) \neq \emptyset \wedge receiveWS(x) \neq \emptyset \\
& \forall x \mid x \in INTERACTION \bullet sendWS(x) \wedge receiveWS(x) = \emptyset
\end{aligned}
$$

where *inter* and *appRole* are finite sets of interactions and application roles in HL7 standard respectively. The functions *sendWS* and *receiveWS* shows total functions relationship as a single interaction can have distinct application roles as sender and receiver. For Example:
inter = Order Fulfilment Request, Promise Activate, Result Status, . . .
appRole = Order Placer, Order Filler,. . .
sendWS = Order Fulfillment Request →Order Placer
receiveWS = Order Fulfillment Request →Order Filler

The process artifacts are made part of the process workflows of different organizations. Organizations conforms to variation of process artifacts based on their requirements. Schema for HLH WSMO Repository based on workflow compliance is as follows:

$$
\begin{array}{|l}
\hline
\textit{HLHWSMORepository\_WorkflowCompliance} \\
org1 : \mathbb{F}\ WORKFLOW \\
org2 : \mathbb{F}\ WORKFLOW \\
org1Compliance : WORKFLOW \leftrightarrow INTERACTION \\
org2Compliance : WORKFLOW \leftrightarrow INTERACTION \\
\hline
\exists\, w \bullet WORKFLOW \mid org1Compliance(w) \neq \#org2Compliance(w) \\
\vee \\
\forall\, w \mid\bullet WORKFLOW \mid org1Compliance(w) = \#org2Compliance(w) \\
\hline
\end{array}
$$

where org1 and org2 represents two different organizations workflows and are compliant to HL7 specifications. These may contain same or different number of interactions in their workflows represented by constraints in the schema.

Both the schemas discussed above can be represented as included schema in HLHWSMORepository schema:

$$
\begin{array}{|l}
\hline
\textit{HLHWSMORepository} \\
\textit{HLHWSMORepository\_ProcessArtifacts} \\
\textit{HLHWSMORepository\_WorkflowCompliance} \\
\hline
\end{array}
$$

In order to represent no change while reading the schemas, the following schema is used:

$$
\begin{array}{|l}
\hline
\Xi\, HLHWSMORepository \\
\Delta HLHWSMORepository \\
\hline
NoChange \equiv APPLICATION\_ROLE' = APPLICATION\_ROLE, \\
INTERACTION' = INTERACTION \\
\hline
\end{array}
$$

These schemas are further used for validating the accuracy and scalability aspects of the proposed systems in the following subsections.

6.2 Accuracy of proposed system

Accuracy aspect of the proposed system is defined in ProcessMediation schema as follows:

Operation: Process Mediation between organizations workflows

$$
\begin{array}{|l}
\hline
\textit{ProcessMediation} \\
\Xi\, HLHWSMORepository \\
report! : Report \\
workflow? : \mathrm{dom}\ org1Compliance \\
\hline
\exists\, w \bullet workflow \mid w \in \mathrm{dom}\ org2Compliance \bullet w = workflow? \\
\quad =: org1Compliance(workflow?) \cap org2Compliance(w) \neq \emptyset \\
\mathbf{report! = WorkflowsAreCompatible} \\
\hline
\end{array}
$$

This operation represent mediation process between two organizations having HL7 compliant systems. These organization's workflows can communicate interactions among themselves and are said to be compatible to each other to exchange set of valid HL7 messages.

6.3 Scalability of HLH WSMO repository for new HL7 specifications

Scalability of the system is measured for new specifications. Therefore change in *Interaction Ontology* occurs when new application roles and interactions are added to the system to support new emerging specifications.It can be represented by following formal method function:

$AddArtifatcsToHLHWSMORepository \mathrel{\widehat{=}} AddApplicationRoles \vee AddInteractions$
where *AddApplicationRoles* and *AddInteractions* are partial functions describing the formal addition of new application roles and interactions to HLHWSMORepository. The schema's of these operations are discusses as follows:
Operation: Adding Application Roles

$$
\begin{array}{l}
\textit{AddApplicationRoles} \\
\hline
\Delta \textit{HLHWSMORepository} \\
\textit{appRole}? : \textit{APPLICATION\_ROLE} \\
\hline
\exists\, \textit{inter} : \textit{INTERACTION} \mid \textit{inter} \in \operatorname{dom} \textit{sendWS} \\
\textit{sendWS}' = \textit{sendWS} \oplus \{\textit{inter} \mapsto \textit{appRole}?\} \\
\vee \\
\exists\, \textit{inter} : \textit{INTERACTION} \mid \textit{inter} \in \operatorname{dom} \textit{receiveWS} \\
\textit{receiveWS}' = \textit{receiveWS} \oplus \{\textit{inter} \mapsto \textit{appRole}?\} \\
\hline
\end{array}
$$

We need to add new application roles *appRole?* to HLHWSMORepository (changing its state to Δ HLH WSMO Repository) and create its relationship with existing interactions changing existing functions from *sendWS* and *receiveWS* to *sendWS'* and *receiveWS'* respectively. The operand ∨is used because an application role can be associated with at least one interaction.
Operation: Adding Interactions

$$
\begin{array}{l}
\textit{AddInteractions} \\
\hline
\Delta \textit{HLHWSMORepository} \\
\textit{inter}? : \textit{INTERACTION} \\
\hline
\exists\, \textit{appRole} : \textit{APPLICATION\_ROLE} \mid \textit{appRole} \in \operatorname{ran} \textit{sendWS} \\
\textit{sendWS}' = \textit{sendWS} \oplus \{\textit{appRole} \mapsto \textit{interaction}?\} \\
\wedge \\
\exists\, \textit{appRole} : \textit{INTERACTION} \mid \textit{appRole} \in \operatorname{ran} \textit{receiveWS} \\
\textit{receiveWS}' = \textit{receiveWS} \oplus \{\textit{appRole} \mapsto \textit{inter}?\} \\
\hline
\end{array}
$$

We need to add new interactions *inter?* to HLHWSMORepository and create its relationship with existing application roles changing *sendWS* and *receiveWS* to *sendWS'* and *receiveWS'* respectively. The operand $\wedge$ is used because an interaction should have distinct sender and receiver application roles.

7 Discussion

This section discuss the features of the proposed system that are associated with existing systems. Moreover, it also highlights features of the proposed system that leverage it from existing systems.

7.1 Common features with existing systems

The proposed system is similar to COCOON, Artemis and PPEPR projects as all these are web based healthcare projects based on semantic technologies. COCOON project uses WSMO infrastructure for service discovery as is used by the proposed system. The proposed system also uses WSMX discovery engine to discover appropriate services. Artemis project is making systems interoperable with each other. To achieve interoperability, standards needs to be followed and Artemis and the proposed system are using HL7 standard for bringing interoperability in the system. PPEPR project is also based on WSMO semantic technology like the proposed system. It mainly handles the heterogeneity issues between different healthcare environments. Our system also resolves heterogeneities semantically between healthcare systems having heterogeneous workflows.

7.2 Differences with existing systems

The major differences between the proposed system and discussed healthcare projects are:

- All the discussed healthcare systems are not based on healthcare standard based workflows. The proposed system is based on HL7 based workflows. Also workflows heterogeneities are not catered in the above mentioned projects.
- The proposed system is based on handling semantic process interoperability that is not handled by any of the projects discussed above. Although PPEPR project is helpful in describing the overall mechanism of workflows. These workflows are static in nature as they only handle the pre-specified interactions. The heterogeneity in workflows of different HL7 compliant healthcare organizations is not handled in PPEPR. Organizations that conform to different flow of interactions in their workflows, cannot effectively communicate until and unless issue of process heterogeneity is resolved. This aspect is not covered in PPEPR as it mainly covers HL7 V2 to HL7 V3 conversion. For the most part it focuses on integration of healthcare systems conforming to different version of HL7 whereas our approach enables process interoperability at a semantic level.

8 Conclusion and future work

The proposed system is based on state of the art semantic web technology WSMO that plays vital role in making healthcare standard achieve true semantic interoperability. It is a cost effective, flexible and interoperable solution with strong emphasis on the importance of semantic process interoperability. It focuses on the timely delivery of information for providing better health to patients. It emphasizes on the importance of semantic process interoperability with data interoperability. The behavioral aspects of HL7 V3 are catered with the help of Interaction Ontology which leads to process interoperable system. Semantic interoperability is not complete without semantic process interoperability. It also shows that WSMF is the most suitable framework for providing semantic web services that leads to bringing automation in the system. The behavioral aspects of HL7 V3 needs integration with standard terminologies in order to effectively use the standard and provide interoperability which is the main concern of healthcare domain nowadays.

The proposed system can play an important role in HL7 V2 and V3 converter as it can help in the conversion from one standard to another by taking in to account behavioral aspects as well. These standards have many differences in behavioral aspects that requires semantic mappings. For example, ADTÂ05 trigger event of V2 can be mapped to PRPA_TE412001UV02 (Inpatient Encounter Appointment scheduled) of V3. the proposed work can also be extended to services ontology for HL7 that can be used for a comprehensive SOA architecture.

Acknowledgments This research was supported by the MKE (The Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency)" (NIPA-2012-(H0301-12-2001)).

References

1. Semantic web health care and life sciences (hcls) interest group. http://www.w3.org/blog/hcls. (Last visited in May 2010)
2. Hl7 standards. http://www.hl7.org/implement/standards/index.cfm?ref=nav. (Last visited in March 2012)
3. Khan WA, Hussain M, Afzal M, Latif K, Ahmad HF, Khattak AM (2009) Towards semantic process interoperability. In: 10th International HL7 Interoperability Conference (IHIC), Kyoto, pp 88–95
4. Standards support interoperability safety and quality patient care. http://www.hitug.org/CA/article_read.asp?title=Standards+Support+Interoperability+Safety+and+Quality+Patient+Care&item=25. Last visited in March 2012
5. Krogstie J, Veres C, Sindre G (2006) Interoperability Through Integrating Semantic Web Technology, Web Services, and Workflow Modeling. Interoperability of Enterprise Software and Applications, pp 147–158
6. Fensel D, Bussler C (2002) The web service modeling framework wsmf. Electron Commer Res Appl 1(2):113–137
7. Miller T (2009) Siemens healthcare—workflow and solution division. http://www.siemens.com/investor/pool/en/investor_relations/financial_publiations/_speeches_and_presentations/cmd_healthcare/cmd_2009_miller_hws.pdf
8. Song X, Hwong B, Matos G, Rudorfer A, Nelson C, Han M, Girenkov A (2006) Understanding requirements for computer-aided healthcare workflows: experiences and challenges. In: International conference on software engineering (ICSE), Shanghai

9. Beeler GW, Huff S, Rishel W, Shakir A-M, Walker M, Mead C, Schadow G (1999) Message development framework. Technical report, Version 3.3, Health Level Seven, Inc
10. Hl7 version 3 standard: Infrastructure management: Transmission infrastructure. Technical report. Normative 2011 (2011)
11. Kerrigan M, Mocan A, Tanler M, Bliem W (2007) Creating semantic web services with the web service modeling toolkit (wsmt). In: Proceedings of workshop on making semantics work for business (MSWFB) at the 1st European semantic technology conference (ESTC), Vienna
12. Cabral L, Domingue J, Mottal E, Payne T, Hakimpour F (2004) Approaches to semantic web services: an overview and comparisons. In: Proceedings of the First European Semantic Web Symposium, ESWS, pp 225–239, Heraklion, Crete (2004)
13. Nixon L, Paslaru E (2004) State of the art of current semantic web services initiatives. Technical report, EU-IST Network of Excellence (NoE) IST-2004-507482 KWEB. http://knowledgeweb.semanticweb.org/semanticportal/deliverables/D2.4.ID1.pdf
14. Motta E, Domingue J, Cabral L, Gaspari M (2003) Irs-ii: a framework and infrastructure for semantic web services. Semant Web-ISWC 2003:306–318
15. Hakimpour F, Sell D, Cabral L, Domingue J, Motta E (2004) Semantic web service composition in irs-iii: the structured approach. In: CEC ’05: Proceedings of the seventh IEEE international conference on e-commerce technology (CEC’05), pp 484–487, Washington, DC
16. Patil AA, Oundhakar SA, Sheth AP, Verma K (2004) Meteor-s web service annotation framework. In Proceedings of the 13th International Conference on World Wide Web, pp 553–562
17. Meteor-s: Semantic web services and processes applying semantics in annotation, quality of service, discovery, composition, execution. http://lsdis.cs.uga.edu/proj/meteor/. (Last visited in Febuary 2011)
18. Battle S et al (2005) Semantic web services framework (swsf) overview. Technical report, W3C Member Submission. http://www.w3.org/Submission/SWSF/
19. de Bruijn J, Fensel D, Kifer M, Kopeck J, Lara R, Lausen H, Polleres A, Roman D, Scicluna J, Toma I (2005) Relationship of wsmo to other relevant technologies. Technical report, W3C Member Submission
20. Della Valle E, Cerizza D, Bicer V, Kabak Y, Laleci GB, Lausen H (2005) The need for semantic web service in the ehealth. In: W3C workshop on Frameworks for Semantics in Web Services
21. Dogac A, Laleci GB, Kirbas S, Kabak Y, Sinir SS, Yildiz A, Gurcan Y (2006) Artemis: deploying semantically enriched web services in the healthcare domain. Inf Syst 31(4–5):321–339
22. Carr CD, Moore SM (2003) Ihe: a model for driving adoption of standards. Comput Med Imaging Graph 27(2):137–146
23. Wozak F, Ammenwerth E, Hörbst A, Sögner P, Mair R, Schabetsberger T Ihe based interoperability for service oriented architectures
24. Wozak F, Ammenwerth E, Horbst A, Sogner P, Mair R, Schabetsberger T (2008) Ihe based interoperability-benefits and challenges. Stud Health Technol Inf 136:771
25. Sahay R, AKhtar W, Fox R (2008) Ppepr: Plug and play electronic patient records. In: Proceedings of the 2008 ACM Symposium on Applied Computing (SAC), Fortaleza, Ceara
26. Sahay R, Fox R, Zimmermann A, Polleres A, Hauswrith M (2011) A methodological approach for ontologising and aligning health level seven (hl7) applications. Availability, reliability and security for business, enterprise and health information systems, pp 102–117
27. Afzal M, Hussain M, Ahmad HF, Ali A (2008) Design and implementation of open source hl7 version 3 for e-health services. In Proceedings of 9th International HL7 Interoperability Conference (IHIC), Crete, pp 174–179
28. Umer S, Afzal M, Hussain M, Latif K, Ahmad HF (2011) Autonomous mapping of hl7 rim and relational database schema. Inf Syst Front :1–14
29. Services: Service oriented architecture and hl7 v3. http://www.hl7.org/v3ballot/html/welcome/environment/index.html. (Last visited in March 2012)