Tool-assisted Healthcare Knowledge to HL7 Message Translation

Priya Jayaratna and Kamran Sartipi
Department of Computing and Software
McMaster University
Hamilton, ON, L8S 4K1, Canada
{*jayarasp, sartipi*}*@mcmaster.ca*

***Abstract*—In the new network-centric healthcare IT environment, standardization of information representation, organization and dissemination is the first step towards achieving semantic interoperability among heterogeneous systems. In this paper, we discuss roadblocks encountered in a real-world project to integrate two disparate healthcare systems based on HL7 v3 standards. We propose a tool-assisted approach to support standard-compliant message workflow design and lay the foundation for a new tool to support our approach.**

I. Introduction

The ability of systems built upon heterogeneous information models to exchange vital clinical, financial and administrative information is pivotal to the success of healthcare organizations in providing quality services. Shifting towards an integrated healthcare environment through ventures such as Electronic Health Record (EHR) [1] requires to leverage the new messaging and terminology standards such as Health Level 7 (HL7) [2] and SNOMED CT[1] [3].

While the new HL7 version 3 (v3) has been hailed over its predecessors for being a ”true” standard offering precision and unambiguity, the worldwide healthcare community has so far been reluctant to adopt it mainly due to its overwhelming complexity. We have experienced these real challenges in our project to integrate a Clinical Decision Support System (namely Vascular Tracker, VT) developed by the COMPETE[2] group [4], with a Cardiac Rehab Center (CRC) in a different location based on HL7 v3 standards.

These standards are sufficiently comprehensive to cover the breadth and depth of the medical domain information. However, organization of the information models into a multi-level, domain-based hierarchy offers a challenging environment for non domain-expert IT personnel. Furthermore, creating semantic maps between legacy data and HL7 v3 messages currently requires a thorough understanding of HL7 information architecture as well as standard clinical terminology systems such as SNOMED CT [3] and LOINC[3] [5]. To the best of our knowledge, there is no open-source tool to support design and implementation of HL7 v3 compliant integration. As such, message workflow design typically involves wading through pages of HL7 documentation using primitive text search tools.

[1]SNOMED CT: Systematized NOmenclature of MEDicine - Clinical Terms
[2]COMPETE III: Computerization of Medical Practice for the Enhancement of Therapeutic Effectiveness
[3]LOINC: Logical Observation Identifiers Names and Codes

The real-world challenges that we faced during the aforementioned project served as our motivation to drive this research. Our overall goal is to develop solutions that reduce the overwhelming complexity of the HL7 v3 compliant integration projects, and consequently a wider adoption of the HL7 v3 standards.

In this paper, we propose a Tool-Assisted Message Mapping Process (TAMMP) to support the message workflow design phase of HL7 v3 compliant integration projects. We lay the foundation for a comprehensive and user-friendly tool to store, locate and explore HL7 v3 artifacts in electronic format using leading edge Semantic Web (SW) technologies [6].

II. Related work

There are a number of commercial support tools available for HL7 version 2. 7Scan [7] is a specialized browser and editor that finds, displays, edits and transmits text-based HL7 version 2 messages with ease. 7Edit [8] is a productivity tool for browsing, editing, searching, validating HL7 messages and communicating with systems that support HL7 format. 7Edit supports HL7 versions 2.1 up to 2.6. NeoTool’s NeoBrowse [9] offers a multi-view interface making HL7 2x messages easy to view and understand. Our research is geared towards developing an open-source tool that supports HL7 version 3 which is fundamentally different from version 2.

The HL7 v3 mapping process proposed in this paper is continuation of work carried out by Yarmand and Sartipi [10]. Their proposed model for message standardization is based on guidelines set forth by Canada Health Infoway[1]. Interaction selection and terminology mapping are offline operations unassisted by tools. In contrast, we propose a tool-assisted approach that is independent of Canadian national guidelines.

In other healthcare integration related research, Liu *et.al.* [11] discuss an HL7 v2 based integration project to establish interoperability between a hospital information system (HIS) and a Picture Achieving and Communication System (PACS) based on DICOM. They propose an information exchange gateway between DICOM and HL7 v2 based on a series of parsers, transaction processors and send/receive modules capable of processing, translating and transmitting data between the two systems. Mirth [12] is a far more advanced, full-fledged, open source healthcare messaging integration engine developed by WebReach, Inc., a health care IT consulting company based out of Irvine, California. Mirth is based on a unique client-server and Enterprise Service Bus (ESB) architecture and consists of connector, filter and transformer modules to send/receive, parse, transform messages from HL7 v2 to legacy formats. Mirth has been adopted by several healthcare organizations to facilitate middleware services in their standard-based integration efforts. The latest and the greatest research and development efforts in electronic health is in the arena of Integrated Electronic Health Record (iEHR). Currently Canada Health Infoway is spearheading projects to realize a SOA-based, shared Electronic Health Record system in Canada leveraging HL7 v3. EHR Infostructure (EHRi) [13], an elaborate framework supporting architectural requirements, tools and environment necessary to build a pan-Canadian EHR, has been developed by Infoway to drive the initiative.

Overall, there's an increasing trend towards standard-based integration of legacy systems leveraging emerging technologies such as SOA and ESB [11], [14], [15]. Our mission is to contribute towards legacy system interoperability by providing guidelines, well-defined processes and tool-support to improve complexity, return on investment (ROI) and turnaround time of HL7 v3 based integration projects.

III. Healthcare standards and technologies

This section introduces healthcare standards and various technologies that are applicable to this paper.

A. Health Level 7 (HL7)

HL7 v3 comprises of a pair of base specifications - an object-oriented information structure called the *Reference Information Model (RIM)* and a set of *vocabulary domains*. RIM and its derivatives describe structure of data in terms of classes, attributes, constraints and relationships whereas the vocabulary domain encapsulate domain concepts and terms. HL7 message refinement process describes how message types are derived from core RIM classes.

HL7 message refinement process. The strategy for development of version3 messages and related information structures is based upon the consistent application of constraints to HL7 RIM and the HL7 Vocabulary Domains and upon the extension of those specifications to create representations constrained to address a specific health care requirement. Constraints are applied on appearance, cardinality, type and vocabulary sets of base classes and attributes in a top down manner to recursively derive progressively specialized information structures.

Following are the list of information models and messaging structures derived from HL7 RIM. *Domain Message Information Model (D-MIM)* is a subset of the RIM that includes a fully expanded set of class clones, attributes and relationships that are used to create messages for any particular domain (e.g., accounting and billing, claims, and patient administration); *Refined Message Information Model (R-MIM)* is used to express the information content for one or more messages within a domain. Each R-MIM is a subset of the D-MIM and only contains the classes, attributes and associations that are required to compose those messages; *Hierarchical Message Description (HMD)* is a tabular representation of the sequence of elements (i.e., classes, attributes and associations) represented in an R-MIM. Each HMD produces a single base message template from which the specific message types are drawn; *Message Type* represents a unique set of constraints on message identification that are presented in different forms such as: grid, table, or spreadsheet.

B. SNOMED CT

SNOMED CT is a comprehensive multilingual, clinical terminology offering a consistent way of indexing, storing, retrieving and aggregating clinical data across specialties and sites of care. SNOMED CT is organized into a hierarchical ontology with each term attached to a concept code, descriptions and relationships with other concepts. The current SNOMED CT version contains close to 283,000 active concepts, 732,000 active terms and 923,000 active relationships, making it the most comprehensive standard terminology system in the world.SNOMED CT is recommended by HL7 Organization as a terminology standard for clinical data exchange.

C. LOINC

LOINC is a database of codes representing terms used primarily in the Laboratory and Observation areas of healthcare. LOINC was initiated in 1994 as a voluntary effort to meet the demand for electronic movement of clinical data from laboratories that produce the data to hospitals and physician's offices [5]. LOINC has been identified by the HL7 Standards Development Organization as a preferred code set for laboratory test names in transactions between health care facilities, laboratories, laboratory testing devices, and public health authorities. Unlike SNOMED, LOINC codes are not organized in any symmetrical or hierarchical manner, thus making the codes arbitrary.

D. Resource Description Framework (RDF)

RDF consists of entities and binary relationships or statements between those entities represented as *subject-predicate-object* triples. In graphical notation of RDF, the source of the relationship is called the *subject*, the labeled arc is the *predicate* (also called property), and the relationships destination is called the *object*. The RDF data model distinguishes between *resources*, which are Uniform Resource Identifiers (URIs) representing a unique concept, property or object, and *literals* which are just strings. The subject and the predicate of a statement are always resources, while the object can be a resource or a literal.

TAMMP tool uses RDF to represent and store metadata information about HL7 artifacts. Semantic Web technologies such as RDF offer a rich platform to implement efficient and accurate semantic search capabilities. By using RDF, any future changes to the HL7 information models and the artifact metamodel can be accommodated with minimum effort.

In recent years a number of Semantic Web (SW) languages such as Ontology Inference Layer (OIL), DARPA Agent Markup Language (DAML) and Web Ontology Language (OWL) [16] have been developed upon RDF. Even though they offer improved descriptiveness, RDF remains the lowest common denominator among all and offers sufficient expressivity and precision for our tool.

E. Sesame framework

Sesame is an open source Java framework for storing, querying and reasoning with RDF and RDF Schema. It can be used as a database for RDF and RDF Schema, or as a Java library for applications that need to work with RDF internally. Sesame consists of a Sesame library, Sesame server and Sesame repositories. The library can be deployed as a Java Servlet application on Apache Tomcat server. The repository can be in-memory or a relational database such as MySQL. Sesame supports an advanced inferencing and query language Sesame Query Language (SeRQL) [17] to query and find implicit information in RDF schema and data. In TAMMP, we use Sesame as the storage and search framework for RDF-encoded HL7 artifacts.

IV. Tool Assisted Message Mapping Process (TAMMP)

Designing HL7 message exchange for integrating COMPETE III's VT application to CRC was a difficult task due to absence of guidelines and tools. Our initial approach was to drill down the HL7 v3 information model hierarchy progressively, from domain to topic to D-MIM to R-MIM to HMD to message types, in an offline process, examining each artifact until the most appropriate message type was found. This top-down domain analysis approach is highly inefficient and carries a steep learning curve for a non-domain expert. TAMMP streamlines the HL7 messaging workflow design process into well-defined steps with the help of a semantic search tool. Figure 1 illustrates different steps involved in the TAMMP.

Artifact Preprocessing step at the top prepares HL7 artifacts and RDF instances for use by the tool. This step is described in detail in Section IV-D.

Step 1: ***Integration Requirements Analysis.*** This step involves examining information exchange requirements of the systems being integrated. System designers are encouraged to use storyboarding and use case analysis techniques to identify domain keywords, transactions and data fields within each scenario.

Step 2: ***Interaction Search.*** Transactions extracted in the previous step are semantically mapped to HL7 Interactions with the help of TAMMP tool's advanced semantic search feature.

HL7 Interactions. Because they provide critical contextual information in addition to payload schema, TAMMP approach to message workflow design is centered around identifying the matching HL7 Interaction(s) for each use case.

HL7 v3 Interaction is a single, one way information flow satisfying requirements of a single business transaction. An Interaction explicitly answers to the following questions:

1) What is a particular payload schema (*Message Type*)?
2) What causes a message to be sent (*Trigger Event*)?
3) How does a receiving system know when it should send a particular type of response message. (*Receiver Responsibilities*)?

The TAMMP tool retrieves Payload schema, Transmission Wrapper schema, Control Act schema, Trigger Event, Application Roles and Receiver Responsibilities associated with an interaction. "Payload schema" specifies the HL7 structure for message content; "Transmission Wrapper schema" specifies various administrative and data transmission related information to be communicated; "Control Act Wrapper" specifies the context of a message (e.g., whether the message is an order, a request, a response etc. and more specifically, it contains the trigger event); "Trigger Event" specifies the conditions under which a particular message type can be used; "Receiver Responsibilities" are interactions that specify how a HL7 compliant receiver system would respond to a message. If the sender requests a report, it would either receive the "requested report" or a "rejection" notification. If the sent information is intended to be persisted with the external system, the possible response would be the status of the operation (i.e, successful/ aborted/ queued etc.). "Application Roles" are only informative and merely suggest what application roles might participate in the interaction.

Information encapsulated in the interactions dictates how the integration program logic is designed. Sending system must have the logic in place to handle all possible synchronously or asynchronous responses.

Step 3: ***Vocabulary Mapping.*** While the previous steps ensure HL7 compliance for message schema, this step ensures that data fields communicated are interpreted accurately by the receiver. This is achieved by converting local terms in to standard terminology codes for transmission. To achieve this, TAMMP tool integrates with terminology systems SNOMED and LOINC.

Step 4: ***Schema Exploration.*** TAMMP offers a user friendly and web-based GUI for exploring the retrieved schema to ensure full HL7 compliance through correct instantiation. Explorer-style browsing capabilities allows for drilling down HL7 information hierarchy with ease. Within each schema, navigation to underlying information models, contained sub-schema called Common Message Elements (CMETs), elements, attributes and constraints is facilitated. The tool also links attributes to appropriate HL7 Vocabulary Domains for drawing proper codes and values.

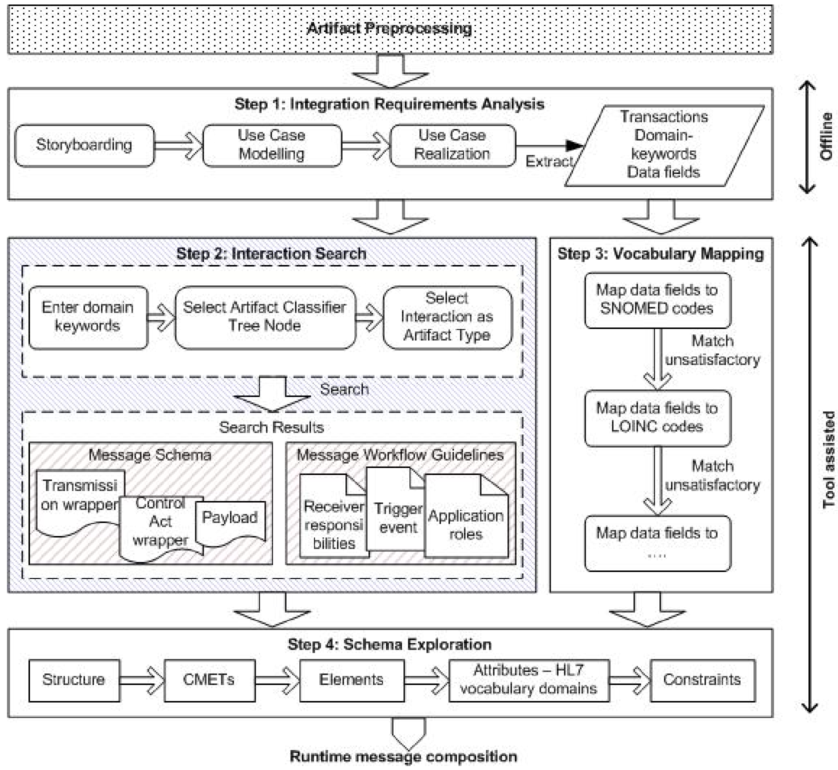

Fig. 1. Tool-Assisted Message Mapping Process - process of mapping healthcare transactions to HL7 messages

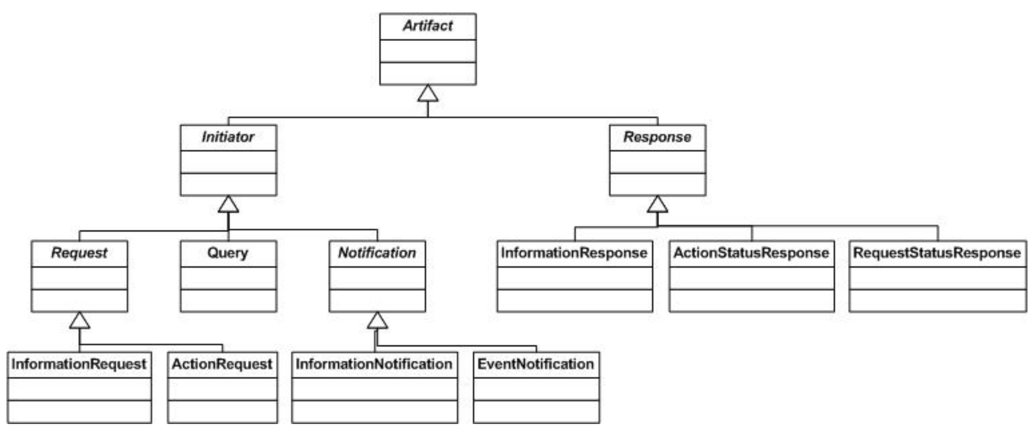

Fig. 2. HL7 v3 artifact classification model

A. Artifact metadata model

In order to support efficient and user friendly artifact search, an enhanced metadata model describing HL7 v3 artifacts is required. Since HL7 information model for each healthcare sub-domain has been developed by a different technical committee of the HL7 Organization, peer levels of models lack homogeneity. For example, Activate, Revise and Nullify message types of the Patient Billing topic of Account and Billing domain derive from the same R-MIM (Patient Billing Account Event), whereas the same message types for Person topic of Patient Administration domain derive from three different R-MIMs. Through extensive analysis of the HL7 v3 information hierarchy, we have developed an alternate classification model (Figure 2) for artifacts based on the actual transactions they support.

At the highest level, artifacts are classified as: *Initiator* or *Response*. Initiator artifacts support an information exchange initiation and can be further classified into *Request, Query* and *Notification*. Request may either be an *InformationRequest* or *ActionRequest* and Notification may be an *EventNotification* or *InformationNotification*.

Response artifacts can be further classified as: *Information-Response*, *ActionStatusResponse* or *RequestStatusResponse*. InformationResponse generally happens in response to a In-

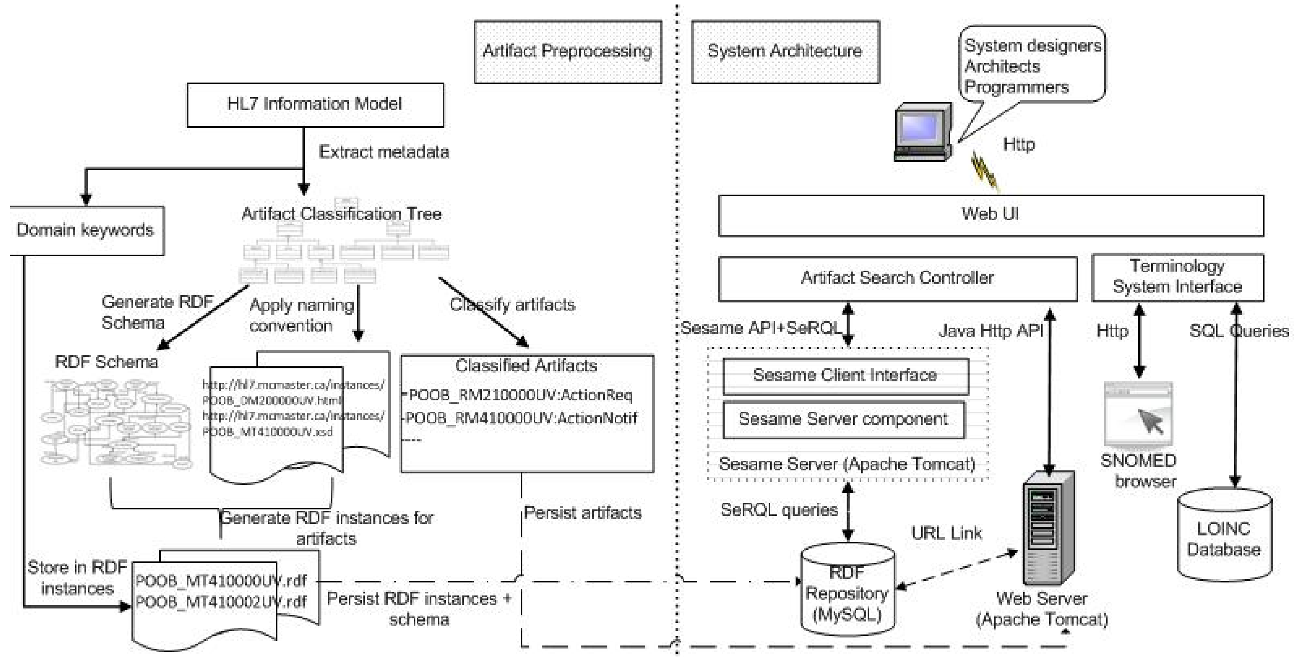

Fig. 3. TAMMP architecture. Left: steps for pre-processing HL7 v3 artifacts for storing in TAMMP repositories. Right: TAMMP system architecture.

formationRequest. An ActionRequest may sometimes require the sender be updated with an ActionStatusResponse (i.e., completed/aborted/queued etc.). RequestStatusResponse communicates the status of a request (i.e., accepted/rejected) to the sender.

The new metadata model encapsulates artifacts at R-MIM, HMD, Message Types and Interactions levels of the HL7 hierarchy. This scheme superimposes well with real world healthcare transactions and hence will help make the message workflow design with TAMMP efficient and accurate.

B. TAMMP tool

TAMMP is an open source, web-based tool that supports message workflow design activities associated with HL7 v3 based integration. The architecture of TAMMP tool is illustrated in Figure 3.

C. Architecture

TAMMP tool consists of Web UI, Web Server, Sesame Server, Artifact Search Controller and Terminology System Interface components. The Java Servelet based GUI is a user friendly environment for searching, browsing, navigating and exploring artifacts. Web Server is an Apache Tomcat server where HL7 artifacts such as XML schema, documentation, XML sample instances, information models and other representations are stored. Sesame Server consists of an RDF repository, Server component and Sesame API. RDF repository is a MySQL database of RDF instances with metadata pertaining to each artifact. Sesame server component handles connections and communications with the RDF repository to execute search and retrieve RDF instances. Artifact Search Controller accesses Sesame infrastructure with Sesame API to leverage its services. It also parses retrieved RDF instances to obtain URLs of artifacts referred to, and requests the web server for them. The Terminology System Interface supports searching for SNOMED and LOINC codes for local terms by integrating into existing SNOMED browser by BT [18] and a MySQL database of LOINC codes. Right portion of Figure 3 illustrates the high level architecture of TAMMP.

D. RDF-based search and retrieval

Our approach to implementing semantic search is to create an RDF instance with metadata for each HL7 artifact. The RDF instance will carry information such as the artifact classification tree node that best represents the artifact and its relationships with other artifacts. Each D-MIM has also been associated with appropriate keywords and phrases that describe information it represents. These keywords would also be stored in RDF instances as metadata. Left half of Figure 3 details the activities involved in offline artifact pre-processing stage. As a first step, HL7 information models will been analyzed to extract the artifact classification model discussed in Section IV-A and domain keywords. This metadata model is converted to an RDF schema by applying rules of RDF syntax and semantics specified by W3C. Since RDF requires all resources to be uniquely identifiable, we adopted an artifact naming convention based on their HL7 artifact ID which is unique. For example, *Observation Request* message schema will be named POOB_MT210000UV.xsd based on its HL7 artifact ID POOB_MT210000UV.xsd. Finally, RDF instances describing the metadata and relationships of each artifact is generated in conformance with the schema and by analyzing the HL7 information models. Artifacts are persisted in the Web Server and RDF instances are stored in the RDF repository for access by the application.

At runtime, the user accesses the search function and inputs domain keywords or phrases such as ”patient”, ”vital signs” or ”blood donation” that appear in use case descriptions derived in the *Integration Requirements Analysis* step. The user may also select the artifact classification tree node that best describes the use case. The Search Controller then generates SeRQL queries based on the search criteria and access the RDF repository via Sesame API. Depending on the strength of search criteria, more than one match per use case may be returned. Information in resulting RDF instances will be parsed and a hierarchy of related URLs pertaining to each RDF instance will be retrieved from the web server. These will be displayed in a user friendly, browseable format. The retrieved message schema can also be downloaded via the tool.

Figure 4 illustrates RDF graph of HL7 v3 artifact metadata model. A part of the XML serialization of the RDF schema is given below.

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [<!ENTITY xsd
"http://www.w3.org/2001/XMLSchema#">]>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xml:base="http://www.mcmaster.ca/hl7/schema"
<rdfs:Class rdf:ID="Artifact">
  <rdfs:Class rdf:ID="D-MIM">
    <rdfs:subClassOf rdf:resource="#Artifact"/>
</rdfs:Class>
<rdfs:Class rdf:ID="R-MIM">
  <rdfs:subClassOf rdf:resource="#Artifact"/>
</rdfs:Class>
<rdfs:Class rdf:ID="Interaction">
  <rdfs:subClassOf rdf:resource="#Artifact"/>
</rdfs:Class>
...
<rdfs:Class rdf:ID="Category"/>
<rdfs:Class rdf:ID="Request">
  <rdfs:subClassOf rdf:resource="#Category"/>
</rdfs:Class>
<rdfs:Class rdf:ID="InformationRequest">
  <rdfs:subClassOf rdf:resource="#Request"/>
</rdfs:Class>
<rdfs:Class rdf:ID="ActionRequest">
  <rdfs:subClassOf rdf:resource="#Request"/>
</rdfs:Class>
<rdfs:Class rdf:ID="Query">
  <rdfs:subClassOf rdf:resource="#Category"/>
</rdfs:Class>
...
<rdfs:Class rdf:ID="Response">
  <rdfs:subClassOf rdf:resource="#Category"/>
</rdfs:Class>
<rdfs:Class rdf:ID="ActionStatusResponse">
  <rdfs:subClassOf rdf:resource="#Response"/>
</rdfs:Class>
...
<rdf:Property rdf:ID="artifactClass">
  <rdfs:range rdf:resource="#Category"/>
  <rdfs:domain rdf:resource="#Artifact"/>
</rdf:Property>
<rdf:Property rdf:ID="keywords">
  <rdfs:domain rdf:resource="#Artifact"/>
</rdf:Property>
</rdf:RDF>
```

The RDF instance for Interation “*Request to record subject observation*” (Artifact ID POOB_IN000001UV) that is persisted in the RDF store is as follows:

```
1. <?xml version="1.0"?>
2. <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
3.  syntax-ns#"xmlns:hl7="http://hl7.mcmaster.ca/hl7/schema#">
4.   <hl7:Interaction rdf:about="http://hl7.mcmaster.ca/
5.     instances/POOB_IN000001UV.html">
6.     <hl7:artifactClass rdf:resource=
7.       "http://hl7.mcmaster.ca/
8.       schema/ActionRequest"/>
9.     <hl7:keywords>observation,
10.       clinical observation,vital signs,
11.       height,weight,blood pressure
12.     </hl7:keywords>
13.   </hl7:Interaction
14. </rdf:RDF>
```

Line 1, `<?xml version="1.0"?>`, is the XML declaration which indicates the version of XML used. Line 2 begins an rdf:RDF element. This indicates that the following XML content (starting here and ending with the `</rdf:RDF>` in line 14 is intended to represent RDF. On the same line, there is an XML namespace declaration, represented as an xmlns attribute of the rdf:RDF start-tag. This declaration specifies that all tags in this content prefixed with rdf: are part of the namespace identified by the URI reference http://www.w3.org/1999/02/22-rdf-syntax-ns#. URI references beginning with this namespace are used for terms from the RDF vocabulary. Line 3 specifies the XML namespace declaration for the prefix hl7:. This specifies that the namespace URI reference http://www.mcmaster.ca/hl7/schema# is to be associated with the hl7: prefix.

Line 4 indicates that the current RDF describes an instance of class Interaction. The URI of the resource described in the RDF is given in the ”about” attribute in lines 4 and 5. Lines 6 to 12 specify some of the properties of this resource. Lines 6, 7 and 8 indicate that value of property artifactClass for the resource is http://hl7.mcmaster.ca/ instances/ActionRequest. Lines 9, 10 and 11 specify that the value of property keywords is ”observation,clinical observation,vital signs,height,weight,blood pressure”.

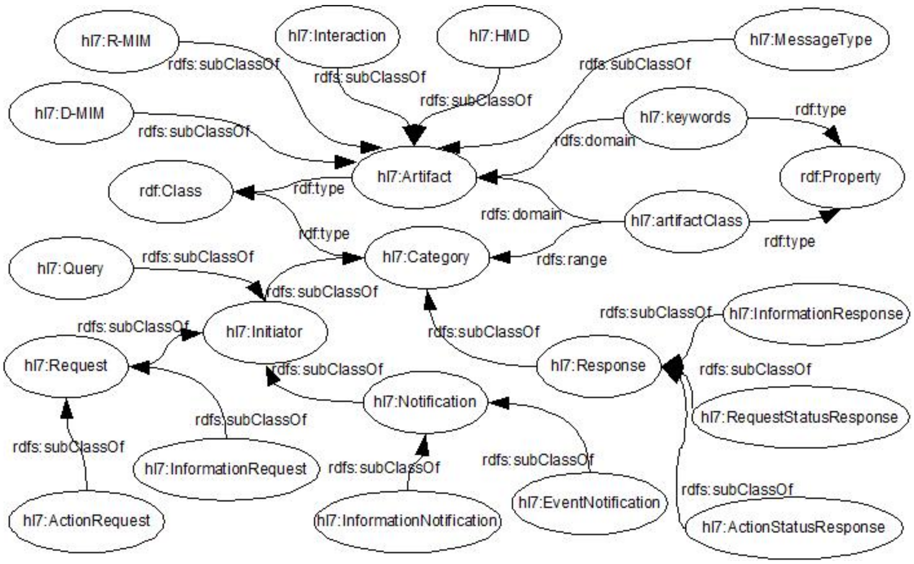

Fig. 4. RDF Graph for HL7 artifact metadata model

V. Conclusion

In this paper, we discussed various obstacles encountered during healthcare integration projects using HL7 v3. We have proposed a tool-assisted message workflow design process with well defined steps, aimed at reducing complexity of associated tasks. The founding concepts, architecture and technologies for a design support tool that aids message selection, terminology mapping and exploration of HL7 artifacts have also been defined. In the process we have demonstrated how Semantic Web technologies can be leveraged to offer advanced metadata search features.

At this time TAMMP tool does not offer an environment for automated message instantiation and schema editing. For future research, we will continue to improve the tool to provide such features to render model to HL7 message translation as seamless as possible.

Increasingly, governments of many countries including Canada are recognizing the importance of the role of Information Systems in improving the quality of public health services. This is evident by unveiling of various new ventures such as the EHR [1] project. While IT companies and healthcare institutions engage in such collaborations, the research community has a vital role to play in solving various technological issues that continue to be bottlenecks.

References

[1] "Canada Health Infoway website," http://www.infoway-inforoute.ca/en/home/home.aspx.

[2] "Health Level 7 website," http://www.hl7.org.

[3] "SNOMED CT website." http://www.ihtsdo.org/snomed-ct/.

[4] "COMPETE III Vascular Tracker website," http://www.compe te-study.com/.

[5] "LOINC website." http://www.loinc.org.

[6] D. Fensel, "The Semantic Web and its Languages," *IEEE Intelligent Systems*, pp. 67–73, nov-dec 2000.

[7] "7Scan HL7 browser," http://www.7scan.com/.

[8] "7Edit," http://www.7edit.com/home/index.php.

[9] NeoTool, "NeoBrowse HL7 browser," http://www.neotool.com/products/neobrowse.

[10] M. Yarmand and K. Sartipi, "Standard-based Data and Service Interoperability in eHealth Systems," in *IEEE ICPC 2008, Amsterdam, Netherlands*, 2008.

[11] B. Liu, X. Li, Z. Liu, Q. Yuan, and X. Yin, "Design and Implementation of information exchange between HIS and PACS Based on HL7 standard," in *Proc. of the 5th International Conference on Information Technology and Application in Biomedicine, China*. IEEE Computer Society, may 2008, pp. 552–555.

[12] G. Bortis, "Experiences with Mirth: An Open Source Health Care Integration Engine," in *Proc. of ICSE08, Leipzig, Germany*. ACM, may 2008, pp. 649–652.

[13] "Infoway EHR Infostructure," http://www.infoway-inforoute.ca/en/WhatWeDo/Infostructure.aspx.

[14] T.H.Yang, "A scalable multi-tier architecture for the National Thaiwan University hospital information system based on HL7 standard," in *CBMS06, Leipzig, Germany*. IEEE, 2006, pp. 99–104.

[15] D. Berler and S. Pavlopoulos, "Design of interoperability framework in a regional healthcare system," in *Engineering in Medicine and Biology Society*, 2004, pp. 3093–3096.

[16] S. Arroyo, R. Lara, Y. Ding, M. Stollberg, and D. Fensel, "Semantic web languages - strengths and weaknesses," http://members.deri.at/ michaels/publications/sw-languages-IADIS04.pdf.

[17] "Sesame RDF Query Language," http://www.openrdf.org/doc/sesame/users/ch06.html.

[18] BT, "SNOMED browser," http://www.jdet.com/.

[19] W3C, "Semantic web," http://www.w3.org/2001/sw/.

[20] "W3C RDF Primer," http://www.w3.org/TR/2004/ REC-rdf-primer-20040210/.

[21] R. Guha, R. McCool, and E. Miller, "Semantic search." ACM, Budapest, Hungary, MAY 2003, pp. 700–709.

[22] X. Ping, L. Ko, and R. Shang, Eds., *Dynamic Messages Creation Method for HL7 based Healthcare Information System*. IEEE, 2007.

[23] "Sesame," http://www.openrdf.org/index.jsp.

[24] Oracle, "Healthcare Transaction Base," http://www.oracle.com/industries/healthcare/htb.html.

[25] "Representing UML in RDF," http://infolab.stanford.edu/ melnik/rdf/uml/.

[26] "Web Ontology Language," http://www.w3.org/2004/OWL/.