An Agent-based Knowledge Management Framework for Electronic Health Record Interoperability

Fang Cao
McMaster University/Department of Computing and Software, Hamilton, Canada
Email: caof@mcmaster.ca

Norm Archer[2] and Skip Poehlman[3]
[2]McMaster University/DeGroote School of Business, Hamilton, Canada
[3]McMaster University/Department of Computing and Software, Hamilton, Canada
Email: {archer, skip}@mcmaster.ca

***Abstract* - In recent years, the dramatic increase in the use of information technology for healthcare has resulted in much innovative research on eHealth applications. But it has been widely acknowledged that unlocking the real value in clinical records is highly dependent upon health information standards that allow interoperability between various clinical systems, supporting the easy exchange of records among stakeholders in the patients circle of care. This paper proposes a software agent based virtual integration framework to integrate multiple electronic health record (EHR) systems from distributed (possibly heterogeneous) databases, featuring three properties. First, a loose coupling between EHR formats and software engineering of the application allows the agent based framework to be flexible for on-line reconfiguration and deployment. Second, the framework is designed with a knowledge base that supports both medical practitioners and consumers, by managing healthcare information at a higher knowledge-based level. Third, the framework integrates distributed databases as well as adaptive user interfaces to support personalized health information systems, which can be used by a range of users with differing requirements. We believe that this agent-based technical framework also demonstrates a new direction for handling other eHealth interoperability issues such as the development of personal health record systems, as well as providing a technical foundation for developing clinical decision support systems.**

***Index Terms*: eHealth, Interoperability, Electronic Health Records, Agents, Framework, Integration**

I. INTRODUCTION

Healthcare at all levels, whether it is local, regional, national and/or international, is a vast open environment typically characterized by shared and distributed decision making and management of care. Such an open environment usually requires the communication of complex and diverse forms of information between a variety of clinical and other settings, as well as the coordination between groups of health care professionals with very different skills and roles. Healthcare software systems must operate effectively in this environment, in order to meet the information needs of patients and health care providers. Practitioners in health care environments require that the information is both timely and error-free, such that recommendations or decisions offered by the software systems are secure and trustworthy [1, 2]. The goal of this document is to explore effective approaches to developing a technical framework as the foundation to support personalized smart eHealth management and future clinical decision support systems (CDSS) for patients and medical practitioners, as well as healthcare domain experts without adhering to any one specific standard data definition. To achieve this goal, a multi-agent system seems an appropriate means to utilize knowledge retrieval and analysis of distributed and possibly heterogeneous databases, and to present automated medical knowledge most suitable for the selected user profile.

A. Motivation and Objectives

One of the emerging trends in today's eHealth is that health care practitioners and patients should be empowered with all the medical information available about patients from multiple sources. However, it is a challenging task to develop a software system to provide effective health information management and at the same time, maintain an adaptive user-friendly interface. A number of research proposals have been proposed [3,4,5] and it has been acknowledged that the overall solution needs to solve four major problems:

- The difference between data and knowledge needs to be addressed. In today's eHealth environment, there are a large number of separate systems and processes generating patient data that are stored in large databases. However, most of these data are in specific medical terminology intended for specialized medical practitioners. In order to customize this health information, these data need to be retrieved and translated into knowledge suitable for consumers, whether they are patients or medical practitioners. Compared to conventional data-based medical applications, knowledge-based systems are more suitable for patients and average citizens, creating a better environment for empowering patients and

doi:10.4304/jetwi.1.2.119-128

promoting "autonomous citizens" [6].

- The intended system should be based on a flexible architecture such that new functions and services can be added without modifying the entire system, allowing it to adapt and evolve dynamically with new data and technology.
- The type of information sent to a specific user should be the information the user needs. Therefore, personalizing information format and presentation is necessary for each user; using parameters that describe his/her own specific profile.
- Multiple distributed database systems present a daunting task for the intended system user to query and analyze data efficiently. The solution should include a framework, which supports an interoperability mechanism to identify and provide necessary measures to incorporate the differences in database models.

The framework we propose is based on a multi-agent platform. This uses a knowledge-based approach for information retrieval, with agent-oriented software that enables dynamic on-line reconfiguration, and adheres to a self adaptation interface for personalized information presentation. We believe that a successful implementation of such a framework can lead to virtual system integration by recognizing the lack of systematic considerations in eHealth interoperability research, including: 1) existing standards for EHR interoperability are difficult to reuse. They are either incomplete in terms of functionality or lacking the specification of the precise meaning of the underlying data [7]; and 2) eHealth applications that utilize EHRs (Electronic Health Records) today are closely coupled with specific EHR standards [7,8], posing a difficult problem in communicating with other systems that use different EHR standards.

The objective of this work is to propose a knowledge based framework to solve these problems by abstracting medical concepts and terminologies and by using a structured hierarchy to establish entities and their relationships.

II. Current State of Art

In this section, we look at the state-of-the-art in the relevant fields of study in our work - specifically eHealth system development and EHR Interoperability.

A. *eHealth System Development*

During the 1990s, as the Internet exploded into public consciousness, a number of e-terms began to appear and proliferate. The terms were useful: e-mail brought new possibilities for people to communicate rapidly and share experiences; e-commerce proposed new ways to conduct business and financial transactions through the Internet. At the same time, the introduction of eHealth represented the promise of information and communication technologies to improve health care systems [9]. At the present time, the term "eHealth" is widely used by many individuals, academic institutions, professional bodies, and funding organizations. It has become an accepted neologism despite the lack of an agreed-upon clear or precise definition. The precise meaning of eHealth varies with the context in which the term was used. For our purposes, we will use a definition adapted from [10]:

'E-health is an emerging field of medical informatics, referring to the organization and delivery of health services and information using the Internet and related technologies. In a broader sense, the term characterizes not only a technical development, but also a new way of working, an attitude, and a commitment for networked, global thinking, to improve health care locally, regionally, and worldwide by using information and communication technology.'

Aside from the increasing use of information technology in healthcare today, many research projects have been conducted to study how to improve the efficiency of eHealth delivery and management. In [11], it is suggested that an emerging trend in eHealth today is that the healthcare business is driven by non-professionals, namely patients (or in business terms, consumers) that with their interests drive new services even in the healthcare field, mostly to empower themselves through access to information and knowledge. In fact, further discussions in [11] indicate that this emerging trend of patient empowerment is one of the major characteristics that distinguish eHealth from telemedicine ('rapid access to shared and remote medical expertise by means of telecommunications and information technologies, no matter where the patient or relevant information is located'). This has resulted in the development of personal health systems. A typical setting in the use case of a personal health system would include at least one consumer, a software system based on a mature technical framework, and the healthcare institutions which include actors such as hospitals or private physicians, health plan insurers, pharmacies, and other related business systems. Most interactions occur between either consumers or healthcare providers and the platform which provides necessary services and data. Successful development of such a platform must consider two aspects. First of all, the relevant data and knowledge must be readily available at the point of care. At first glance, this is not a difficult problem since most healthcare institutions now use computer systems for record keeping. However, most of the data is meant to be used directly by doctors and other practitioners, and would not be easy to understand for average patients. Secondly, the system being used must be designed to accommodate future changes as well as different user roles. A good practice in developing software is to follow an incremental process which divides the entire problem into sub-problems and then solves one of them at a time. Given the huge amount of information about patients, scattered throughout multiple sources, it is not possible to include information from all medical areas that one system can handle from the beginning. A better approach would be to develop a prototype of a system which handles a specific medical area, allowing more functionality to be added in the future without completely rewriting the existing system. On the other hand, since there is a big difference in terms of the expertise of medical practitioners and the average consumer, the system should provide an adaptive interface which supports the characteristics of a specific user profile, and display the most relevant information to that specific user in a suitable environment.

B. Electronic Medical Records and Interoperability

Medical information systems today store clinical information about patients in many kinds of proprietary formats. To address interoperability problems between different formats, several EHR standards have been developed that structure the clinical content for the purpose of exchange. This section presents a brief introduction to the most relevant EHRs, examines their level of interoperability, and assesses them in terms of content structure, multimedia support, and access services.

Electronic Healthcare Records have been an active research field in medical informatics for many years. There have been various definitions of EHRs, and we will use the definition [12] "digitally stored health care information about an individual's lifetime with the purpose of supporting continuity of care, education and research, and ensuring confidentiality at all times." An EHR includes information such as observations, laboratory tests, diagnostic imaging reports, treatments, therapies, patient identifying information, legal permissions, and allergies. Currently, this information is stored in all kinds of proprietary formats through a multitude of medical information systems available on the market and used by different organizations. This results in a severe interoperability problem in the healthcare informatics domain.

To address the EHR interoperability problem, several standards have been proposed and continue to evolve. These standards include Health Level 7 (HL7) Clinical Document Architecture (CDA) [13], openEHR [14], and CEN EHRcom [15]. These standards are all intended to structure clinical information content and provide markup for the purpose of exchange. Another industry initiative is Integrating the Healthcare Enterprise (IHE) [16] which has specified the Cross-Enterprise Document Sharing (XDS) integration profile [17]. The idea behind IHE XDS is to use ebXML registry/repository architecture to facilitate the sharing of healthcare documents. Despite the popularity of these interoperability standards, their problems are obvious. First of all, standards such as HL7 CDA only specify the medical contents of a message and therefore are not a complete interoperability framework. Secondly, they do not distinguish knowledge from data, which may affect the ability of the average healthcare consumer to understand the information. Even in IHE XDS, the meaning of meta-data of all shared documents is not explicitly specified [18, 19]. Thirdly, there is a lack of design methodology from the software perspective. This poses difficulties in promoting program reuse and dynamic configuration. Our strategy is to allow the separation of EHR content development from system design, so that a change in EHR content causes minimal change in system configuration and deployment.

Another important issue in integrating electronic medical records is semantic interoperability. The medical field does not have a unified terminology set, which causes a problem for semantic interoperability. This is reflected by the fact that the messaging standards used in eHealth today may use different terminologies for the same concept. Any incorrect interpretation of these concepts could result in misdiagnosis of the patient's illness or even death. Therefore, semantic interoperability is critical in any EHR integration framework. Traditional terminology systems are based on the enumerative classification of concepts and terms. These schemes are constructed entirely by experts enumerating all the possible concepts that are to be represented within the scheme. This approach lacks reusability and extensibility, and is only appropriate when the context is limited to summary descriptions and interpreted by skilled humans. An alternative approach is to capture the general principles and knowledge that underlay the use of medical terminology within a formal model, which can be used to generate only sensible medical concepts and associated terms. Outside of medicine, descriptive logics are being increasingly used for knowledge representation, indexing, and data management. Therefore, it is considered a fitting theoretic foundation to build a formal knowledge model. One example is SNOMED CT (Systematized Nomenclature of Medicine -- Clinical Terms) [20], which is a systematically organized, and computer process-able collection of medical terminology covering most areas of clinical information. In SNOMED CT, logical terms such as "terms" and "concepts" are used to model different levels of entities. It also uses a compositional structure which means the underlying knowledge model consists of a limited number of elementary concepts, a set of attributes, and a series of rules to specify how individual elements may be combined to form complex concepts. For example, the term "pain" is an atomic concept, and "back pain" represents composition of two atomic concepts "back" and "pain". Although SNOMED CT actually considers the term "back pain" as a single concept, the composition rule allows for other concepts to be added forming new concepts such as "chronic back pain". Theoretically, one can add as many adjectives to describe complex concepts in SNOMED CT. Another ontological approach is provided by OpenGALEN[21], which is an open-source project aimed at providing semantic integration through its terminology server. Similar to SNOMED CT, OpenGALEN constructs medical terminologies based on description logic models. The heart of OpenGALEN is a formal conceptual information model, namely the Common Reference Information Model. This conceptual model stores the clinical content such as medical concepts and can be used as a framework for specific knowledge about protocols or decision support. The GALEN Common Reference Model is built with GRAIL (Galen Representation and Integration Language). GRAIL is a concept modeling language and a formal system for modeling medical concepts in a form suitable for computer reasoning. Similar to SNOMED CT, the GALEN Common Reference Model is compositional in the way that GRAIL can declaratively combine atomic concepts to form complex concepts. The use of the formal language GRAIL not only bridges the gap between coding categories for medical information and day-to-day use of clinical language, but also has the advantage of generating explicit and medically sensible clinical information. Finally, the GALEN terminology server is the software solution which implements the GRAIL language and stores the reference information model. It is also the device that provides the terminology integration service to client applications.

III. METHODOLOGY

With the goal of designing a system based on flexibility and adaptability, and the need to use existing information and applications, the necessity of extending an existing framework becomes important. Therefore, the best strategy is to reuse and extend the functionality of an existing framework. This requires a clear understanding of the architectural foundation of our intended system.

The key aspects of the agent based framework are data repository modeling and mechanisms for agent communication. This can be abstracted by the entities shown in Figure 1. The database in which the EHR information is stored and processed constitutes the basic building blocks of our data repository. The various information agents are communicating with the database and the knowledge base where the EHR information is further processed and stored as knowledge queries. This is done by the data wrapper agent, which has sufficient knowledge of underlying database structures and is responsible for transforming data tables into knowledge representations. The knowledge representation in this case contains the following components:

- *Meta-data used for abstract information on knowledge and other related information.* The meta-data works like a multi-level index for relational databases, and is mainly used for fast searches. For example, meta-data for a diabetes knowledge base could have information such as blood sugar level and insulin usage or possible trends such as climbing sugar levels indicative of approaching diabetes or marker tracking (oxalate levels or sodium levels, etc.) in response to treatment therapies or changing food nutrition/exercise programs. Any queries that contain these query parameters could be used for cross-referencing the diabetes knowledge base for possible relationships, even if the original query was not intended for diabetic diagnosis.
- *Common medical terminology base used for medical terminology semantic interoperability.* The medical field does not have a unified terminology set, which causes a problem for semantic interoperability. For example, the terms heart attack, myocardial infarction, and MI may mean the same thing to a cardiologist, but they are all different to a computer. Therefore, a unified and comprehensive medical terminology systems must be included as part of the information infrastructure. Given the complexity of medical domain, an enumerative approach of a terminology base is impractical. A better solution is to represent medical terminologies as knowledge using a logic-based formalism. This is typically done by designing a common reference model formulated in specialized description logic. The concepts and terms in a common reference model are usually organized in one or more hierarchies, while the model checking algorithms can be customized to query the concepts and instance described.
- *User profile for different user targets.* Our framework targets different user groups and returns information intelligently, based on the user's profile and past knowledge patterns. For example, a result desired by a physician and a patient is usually different. Physicians may ask for as much information as possible, therefore helping them understand more about a patient situation and prevent further possible complications, whereas patients may just need to know if they actually have the sickness they suspected. A user profile contains basic parameters of the user's expert level and possible preferences. The user profile also analyzes the user's past history in order to get an update on the user's preferences.
- *Query definition for fast construction of dynamic queries based on user inputs.* The query definition contains all necessary database syntactic constructs and any of their possible extensions for the query search engine.

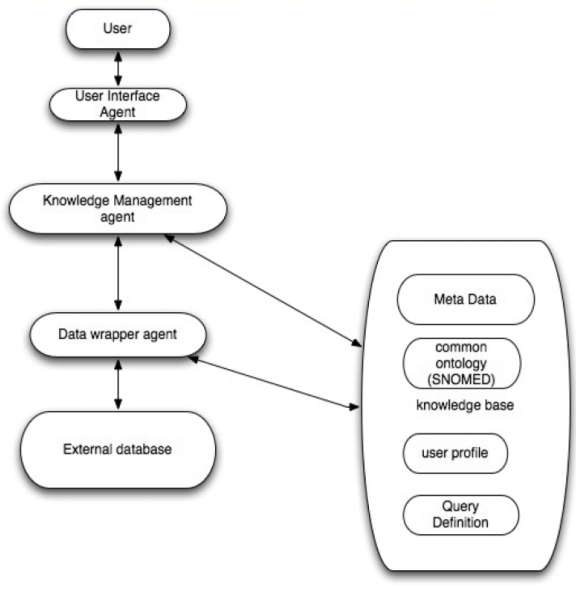

Figure 1. Overall System Architecture

A. Hierarchical Ontology Development

An ontology defines a common vocabulary for researchers who need to share information in a domain. It includes machine-interpretable definitions of basic concepts in the domain, and relations among them. The main reason for using an ontology in domain-specific data modeling is that it enables a reuse of domain knowledge and allows a separation of domain knowledge from operational knowledge. In the case of EHR interoperability, the ontology allows us to focus on the modeling of health-related information through the functionalities of the agents and other aspects in the framework. Data representation in the knowledge base can be modeled by a hierarchical ontology. For the purpose of easing transition from an EHR standard to the underlying messaging, we use HL7 RIM as the foundation of our ontology, as shown in Figure 2, edited in Protégé[22]. There are specific reasons to do so. First of all, HL7 RIM is an excellent referencing point for an ontology since it has a similar information structure and covers all the entities that are modeled in an HL7 V3 message. Secondly, HL7 RIM is a high-level data model that has been put through extensive validation processes by domain experts. Therefore, it provides certain facts about entities within the ontology hierarchy. Third, an ontology based on HL7 RIM helps the transition to HL7 messaging by reusing its data types and semantics. The ontology is stored as a dynamically linked library (DLL) so it is only loaded when necessary. An advantage of using a DLL is its dynamic updating so that any change made to the ontology can be reflected in the code to update the data modeling hierarchy.

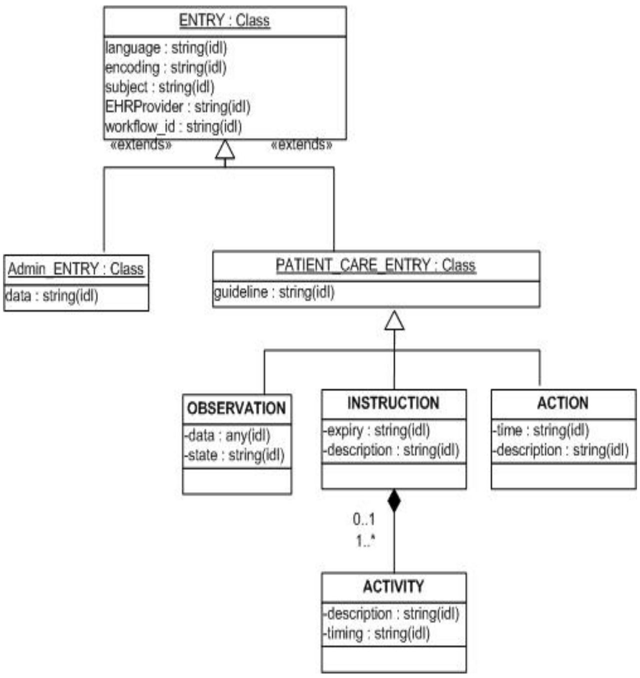

Figure 2. Example of HL7 RIM - Based Ontology

The development of an ontology-based knowledge base is a complex process. The procedures that need to be undertaken could vary based on the current context. The process usually requires entity modeling, domain facts, and inference rules. As mentioned earlier, HL7 RIM provides a good reference for entity modeling and domain facts in healthcare. However, this only gives us an empty knowledge system since the abstract inference rules have not been added to become executable rules.

Since we begin development from a small ontology, we are using an incremental approach for the ontology engineering process. A set of representative facts can be used to validate the ontology and, if successfully validated, we can add more domain facts to build more complex rules. At this point, the abstract inference rules we use are induced from the class-entity relationship diagram in HL7 RIM. For example, basic semantic rules such as isComponentOf can be induced from the subclass type in the UML class diagram in Figure 2. From basic rules such as isComponentOf, we can induce other rules such as acts ON so that behavior acts on a subclass entity are automatically transmitted to a parent class entity. For example, a disorder which acts on a heart valve can be automatically induced to

DisOrder acts ON (valve IsComponentOf Heart)

Abstract inference rules such as those above can be transformed into OWL statements using Protégé, and implemented straightforwardly in a Java-based domain modeling tool such as JADE [23] or JESS [24]. Furthermore, JESS integration in Protégé makes it possible to reuse the ontology in the knowledge rules.

B. Ontology Mediation

When an HL7 message arrives that is encoded in an EHR standard format, we need to mediate the message with the ontology. What this means is that the knowledge management agent must analyze the various parts of HL7 message, map it to the ontology, and make appropriate adjustments. The process of handling the possible overlaps and mismatches between ontologies is called ontology mediation. There are different approaches to ontology mediation such as ontology mapping, ontology alignment and merging, etc [25]. For our purposes, we start from a relatively small scale and use a specific pattern for specification and exchange of ontology differences.

This pattern is used to accommodate differences between attributes in different classes used in separate ontologies. One can use this to map one concept to another concept, or an attribute value in a concept to another concept. For example, assume we have two incoming EHR messages in two different ontologies that describe patient demographic information. Ontology O1 has a concept Patient with attribute Gender. The other ontology O2 has no Patient concept but instead has two concepts Man and Woman. For the attribute Gender in O1, the possible values are Male and Female. Now we need to create two mapping rules to handle the mismatches in the style of modeling: one for mapping Male to Man, the other for mapping Female to Woman. The mapping becomes (Patient Female attributeCondition(Gender, Female)). This example illustrates the mapping from a concept, namely Man and Woman, to attribute values in another concept, namely Male and Female in Patient. This pattern can be generalized to become a general solution of class mapping such that a general mapping method for the above example could be:

Mapping(Patient or (Man or Woman))

The mapping pattern can be used to develop a mapping language used for ontology mediation. For example, this language can be transformed into mapping languages such as OWL DL [26] and WSML-Light [27]. The possible difficulty is that the original vocabularies and symbols used to describe rules and relations may not be allowed in the target language. This could be solved by adding extension mechanisms that allow arbitrary logic relations in the transformed language.

C. *A Service Oriented Framework with Agent Support*

A service-oriented architecture (SOA) is a software architectural concept where applications are partitioned into discrete units of functionality called services. Each service implements a set of related business rules or function points. The basic building blocks of SOA are shown in Figure 3. The services are provided to the client

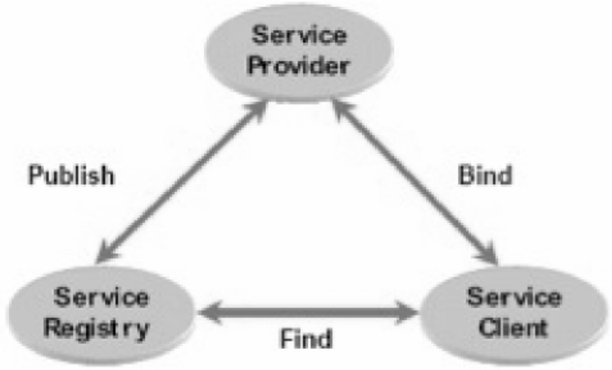

Figure 3. Basic Service Oriented Architecture (Adapted from [28])

applications or consumers. The advantage of adopting SOA architecture in the software design is the flexibility for change. SOA allows for the reuse of applications or applications components as well as promising interoperability between heterogeneous applications and technologies.

Services are offered by service providers that procure the service implementations, supply their service descriptions, and provide related technical and business support. Since services may be offered by different enterprises and communicate over the Internet, they provide a distributed computing infrastructure for cross-enterprise application integration and collaboration. Clients of services can be other solutions or applications within an enterprise or clients outside the enterprise, whether these are external applications, processes or customers/users. To satisfy these requirements, services should be technologically neutral in that the invocation mechanisms (protocols, descriptions and discovery mechanisms) should comply with widely accepted standards. Services should also be loosely coupled as they must not require knowledge or any internal structures or conventions (context) on the client or service side.

The shift towards the service-oriented computing paradigm not only involves a new way of conceptualizing complex distributed applications, but also calls for more abstract software development and deployment technologies. Agent-oriented software engineering delivers such an abstraction.

SOA is an architectural approach that has the objective of achieving loose coupling among interacting software agents. A service is defined as a unit of work done by a service provider and achieves desired results for a given consumer, where both provider and consumer are roles played by software agents on behalf of their owners. The results of a service are usually a change of state for the consumer, or for the provider or for both. A high-level interaction between agents in a service-oriented framework is shown in Figure 4. The interaction between agent components in a service-oriented architecture must satisfy two constraints. First of all, the services should present a small set of interfaces to all participating software agents. Secondly, the

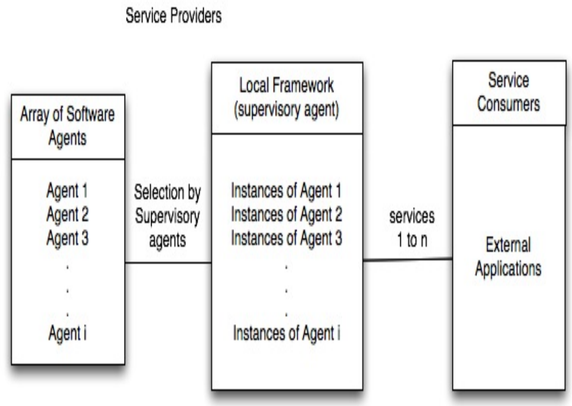

Figure 4. Abstract Interaction Between Agents and Services

agent communication mechanism must exchange a set of descriptive messages constrained by an extensive schema delivered through the interfaces. In our scenario, the message layer implemented by the agent communication language should consist of underlying clinical content encoding as well as language constructs that provide the message exchange mechanism. The agent communication language must also be extensible to allow new versions of services to be introduced without interrupting existing services. As we will explain later, the agent interaction is based on a flexible communication language which encapsulates underlying clinical content supported by a medical information standard such as an ontology.

To be qualified as an agent, a system is often required to have properties such as autonomy, social ability, reactivity, and proactivity [29]. The key feature which makes it possible to implement systems with the above properties is that programming is done at a very abstract level; more precisely, using a terminology introduced by Newell, at the knowledge level [30]. In the agent-oriented vision, software is built not by providing low level imperative lines of code to be followed sequentially, but rather by defining high-level goals to be achieved. In order for an agent to achieve its goals it must have a number of capabilities; most notably, an agent must have reasoning, communicating, and acting abilities. We will focus on two main properties of agents in our framework: reasoning and communicating.

Reasoning: Given a set of goals, knowledge about the world's behavior and information on the current status of the world, the agent must be able to reason to achieve the goals that have been delegated to it. In our framework, knowing a transactional process in the form of a ontology description, a security or transactional requirement or a rule in the form of a ontology specification allows the agent to model the world's behavior and therefore to plan and act accordingly. Figure 5 depicts an example of this process. Since our ontology is hierarchical and is only loaded dynamically, each ontology is handled by a single type of agent, which only has that specific knowledge. Therefore, it is the responsibility of the knowledge management agent to spawn a new agent that the domain area is required to handle the overall situation. The knowledge

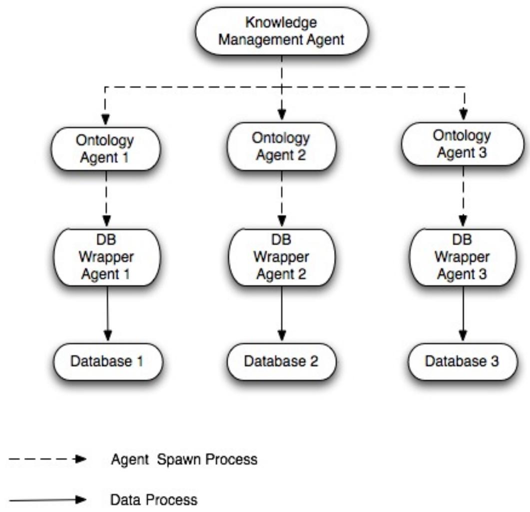

Figure 5. Agent Spawning Model

management agent acts as a central broker and provides that service by generating the correct type of agents. Meanwhile it loads the required library that conforms to the ontology.

Communicating: Agents must be able to communicate with one another and with services in order to cooperate, reach agreements, and negotiate business interactions. This happens via standardized languages such as those offered by XML-based web service descriptions. We use KQML (Knowledge Query and Manipulation Language) [31,32,33] to handle communications between agents. The overall communication mechanism in our framework is shown in Figure 6. The communication component is encapsulated by the use of KQML among agents, which provides a service component that is responsible for the interaction. KQML offers an abstraction of an information agent (provider or consumer) at a higher level. In particular, KQML assumes a model of an agent as a knowledge-based system (KBS). Such systems can usually be modeled as having two virtual knowledge bases – one representing the agent's information store (i.e., beliefs) and the other representing its intentions (i.e., goals) [34].

Furthermore, agent proactivity is represented by acting on their behalf based on their goals and beliefs in the environment they populate. In the adaptive user interface scenario, acting is based on reasoning and possibly communicating with the knowledge component that stores the user profile. For example, user search histories are stored in the knowledge base so that each user's next search is compared to previous search parameters. The adaptive user interface reasons with the user profile and its past history to determine if a novice or expert interface is required. If a decision cannot be made, the adaptive interface agent will ask the user what type of interface he/she prefers, and then load the necessary graphical interface components to render the screen. This may not be the perfect solution at first sight, but it is likely that the user parameters such as occupation and expert level information will lead to a best possible scenario.

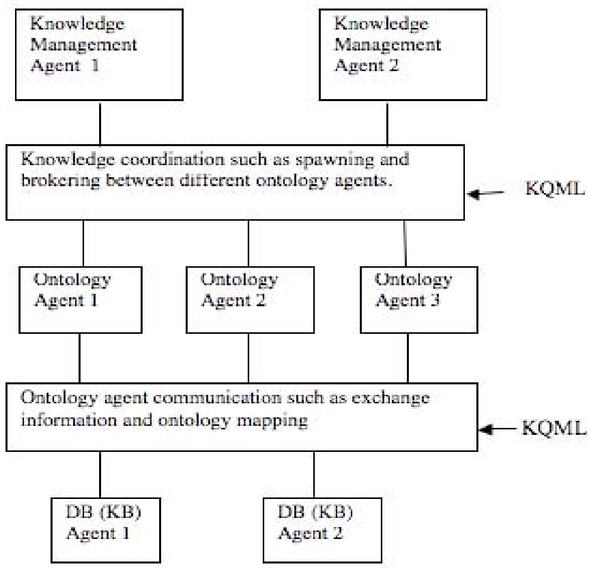

Figure 6. Abstract Communication Model

D. *Knowledge-based Access to Distributed Databases*

Knowledge-based systems (KBS) are productive tools of Artificial Intelligence (AI) working in a narrow domain to impact quality, effectiveness, and knowledge-oriented approach in decision-making process. According to [35], KBS possess characteristics such as:

- Providing a high intelligence level. The difference between knowledge and data is that knowledge provides a higher level of conceptual intelligence, whereas data are often raw and not self-explanatory.
- Offering a vast knowledge base. One of the main components of KBS is the knowledge base, in which domain knowledge, meta-knowledge (knowledge about knowledge), factual data, procedural rules, business heuristics, etc. are available. In the case of EHR knowledge

modeling, the knowledge base could include an ontology modeling hierarchy, semantic interoperability framework, database heuristics, logic inference rules, etc.

- Improving management activities: Explanation, reasoning, and self-learning are important aspects of KBS. These characteristics help improve the accountability and scope of the system. The intelligent decision-making capability shown by a KBS also provides justification for the decisions made by the system. Figure 7 illustrates a typical structure of KBS and how different levels of data are viewed in the data pyramid in KBS.

As most business applications deal with several databases of a homogeneous nature, they can interact easily. However, in eHealth applications, two problems pose difficult challenges in achieving interoperability in accessing and manipulating data. First of all, the existence and use of multiple EHR presents a problem for effective data sharing and management. Typical

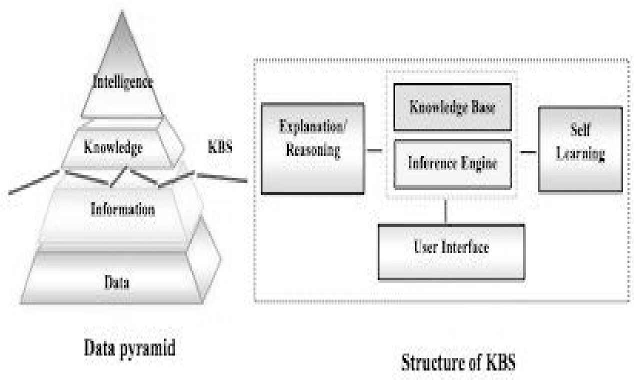

Figure 7. Knowledge Base System Architecture (Adapted from [35])

development of an EHR system is usually done separately and without knowledge of interoperability with other standards. Secondly, the heterogeneous nature of databases does not support just one type of data modeling topology. For a thorough consideration of achieving interoperability, data heterogeneity must be considered. For example, if we assume that only a relational database is used, an agent could be deployed to integrate queries to the database and make the system flexible. However, accommodating different types of databases is a much more complex problem. The possibility of using an agent oriented system has been suggested, to address the heterogeneous nature of complex environments such as eHealth. For example, in [36], an agent-based architecture based on speech-act theory is proposed to address heterogeneity issues from various data sources. A similar approach is used in [37] where a multi agent system is designed for a geographical information system and the agent interaction is based on a mediator model. In our scenario, the data wrapper agent is responsible for performing schema transformation and query translation, and essentially provides a programming environment explicitly for the exploitation and reuse of uniform data models. The data wrapper agent will be communicating with the knowledge management agent, which controls the storage of meta-data and schema mapping information.

IV. Conclusions and Future Work

Over the years, the existence and use of multiple EHRs has presented problems for effective data sharing and management, since these EHRs were developed separately and without a need for interoperability with other standards. They also do not promote software reuse, and prevent flexible adaptive software maintenance.

This paper presents a conceptual agent-based framework to address electronic health record interoperability. The distributed and often heterogeneous nature of health information exchange implies the possibility of adopting a service-oriented approach where the services are implemented by various agents communicating via KQML. This knowledge-based methodology provides a concrete and more precise model of medical information exchange, especially for the benefit of healthcare consumers. The knowledge base is modeled primarily at a high level using a multi-level ontology where different medical domain information is stored in dynamically linked libraries. This provides the benefit of efficient network bandwidth usage and flexible system reconfiguration as only the necessary libraries are loaded at the right time. Overall, this framework provides a basis for personal health system development which promotes software reuse and allows dynamic reconfiguration without disrupting other services in the environment.

We think this framework is the first step to truly achieve seamless integration of electronic medical information exchange and provide a better quality of healthcare. A prototype of such a system using a small ontology example is being developed; however, it is unlikely that an ontology for complete healthcare information will ever be truly achieved. Compromises may need to be made during agent reasoning and communication that are usually based on incomplete information. Determining how close we will get to a desired final interoperability result is still subject to careful testing. To evaluate our final result, we intend to use a scenario-based analysis to determine if our framework satisfies our initial research objective. The reason that we intend to focus on framework validation is that this research is a feasibility study to investigate how multiple EHR sources can be placed systematically into a knowledge accessible framework. Specific test cases will be developed at the later stage of the research to examine whether our implementation meets our goal.

This informal presentation of our framework lacks formal grounding, but formal modeling tools such as Petri Nets could help to provide a thorough verification of our framework. In addition, if the proposed complex ontology is to be fully developed, there must be cooperation among different domain experts and groups.

ACKNOWLEDGEMENT

This research was supported by grants to Dr. Norm Archer and Dr. Skip Poehlman from the Natural Sciences and Engineering Research Council of Canada.

REFERENCES

[1] J. L. Nealon, A. Moreno, "Agent-based Applications in Healthcare. Applications of Software Agent Technology in the Healthcare Domain", In Weitstein Series in Software Agent Technologies. Birkhauser Verlag, Basel, 2003.

[2] J. Nealon, "Agent Applied in Healthcare" AI Communications, 18(3):171–173, 2000.

[3] Kyusuk Chung, Ross M. Mullner, "Current Issues in Health Care Informatics", Journal of Medical Systems, 30(1), February 2006.

[4] Shankararaman Lobato, E. V.PIRA, "A Personalized Information Retrieval Agent", In IASTED International Conference on Artificial Intelligence and Soft Computing,, 1999.

[5] A.Isern Moreno, "Accessing Distributed Healthcare Services Through Smart Agents", In Proceedings of the 4th IEEE International Workshop on Enterprise Networking and Computing in the Health Care Industry, Nancy, France, healthCom 2002.

[6] M.Cabrera, J-C Burgelman, M.Boden, O.da.Costa, C.Rodriguez, "eHealth in 2010: Realising a Knowledge-based Approach to Healthcare in the EU Challenges for the Ambient Care System", Technical Report EUR 21486 URL:ftp://ftp.jrc.es/pub/EURdoc/21486/ExeSumm.pdf Date Accessed: March 2009

[7] Eichelberg, Aden, Riesmeier, Dogac, and Laleci., "A Survey and Analysis of Electronic Healthcare Record Standards" CSURV: Computing Surveys, 37, 2005.

[8] C. P. Waegemann, editor. Toward an Electronic Health Record Europe '97: Conference — October 1997, London, Toward and Electronic Health Record Europe 1997, Newton, MA, USA, 1997. Medical Record Institute.

[9] Alvarez RC, "The Promise of e-health-a Canadian Perspective", eHealth Int, 17(1), Sept 2002.Eichelberg, Aden, Riesmeier, Dogac, and Laleci., "A Survey and Analysis of Electronic Healthcare Record Standards" CSURV: Computing Surveys, 37, 2005.

[10] G Eysenbach, "What is e-health?" J Med Internet Res, 3(2), 2001.

[11] G Eysenbach, "What is e-health? Part 2", J Med Internet Res, 2001.

[12] I. Iakovidis, 'Towards Personal Health Record: Current situation, obstacles and trends in implementation of electronic healthcare records in europe" International Journal of Medical Informatics, 52(128):105–117, 1998.

[13] Calvin Beebe, Paul V. Biron Sandra Lee Boyer, Daniel Essin, Elliot Kimber, et al. "The HL7 Clinical Document Architecture, J Am Med Inform Assoc. 2001 Nov-Dec;8(6):552-69.

[14] openEHR Community, "OpenEHR. Future Proof and Flexible", URL: http://www.openehr. org. Date Accessed: Feb 2008

[15] CEN/ISSS eHealth Focus Group, "Report from the CEN/ISSS eHealth Standardization Focus Group: Current and Future Standardization Issues in the eHealth Domain: Achieving Interoperability", Final version, March 2005. URL:http://www.i2-health.org/ehealth-iop/eHealthStandardizationExecutive summaryFinalversion2005-02-21.pdf Date Accessed: Dec 2007

[16] IHE, "Integrating the Healthcare Enterprise". URL: http://www.ihe.net.

[17] Integrating Health Enterprises, "IHE IT Infrastructure Technical Framework Supplement 2004-2005 Cross-Reference Document Sharing (XDS)" Trial Implementation Version, unpublished, Mar 2004, Date Accessed: Apr 2005

[18] Thomas Aden, Asuman Dogac, Gokce Laleci, Marco Eichelberg., "Enhancing IHE XDS for Federated Clinical Affinity Domain Support" IEEE Transactions on Information Technology in Biomedicine, 11(2):213–221, 2007

[19] Alper Okcan, Asuman Dogac, Veli Bicer, "Collaborative Business Process Support in IHE XDS through ebXML" Business Processes. In Proceedings of the 22nd International Conference on Data Engineering, ICDE 2006, Atlanta, GA, USA, page 91. IEEE Computer Society, 2006.

[20] International Health Terminology Standards Development Organisation, "SNOMED CT", URL: http://www.ihtsdo.org/snomed-ct/, Date Accessed: Jan 2008

[21] OpenGALEN Project, "Mission, Background, and Content". URL: http://www.opengalen.org/index.html, Date Accessed: Jan 2007

[22] Daniel L. Rubin, Holger Knublauch, Ray W. Fergerson, Olivier Dameron, and Mark A. Musen. "Protégé-OWL: Creating ontology-driven reasoning applications with the web ontology language", AMIA Annu Symp Proc. 2005; page 1179.

[23] A. Poggi, G. Rimassa, F. Bellifemine, G. Caire. "JADE a white paper", unpublished, 2003. URL: http://jade.tilab.com/papers/2003/WhitePaperJADEEXP.pdf Date Accessed: Jul 2007

[24] E. J. Friedman-Hill, "Jess, the Java expert system shell", unpublished, May 2007, URL: http://www.jessrules.com/jess/docs/52/ Date Accessed: Jul 2007

[25] Paolo Bouquet, Fausto Giunchiglia, Frank van Harmelen, Luciano Serafini, Heiner Stuckenschmidt, "Contextualizing Ontologies" Journal of Web Semantics, 1(4):325–343, 2004.

[26] Jeff Z. Pan, Edward Thomas, "Approximating OWL-DL ontologies'', In AAAI, pages 1434–1439. AAAI Press, 2007.

[27] David Booth, Canyang Kevin Liu. "Web services description language (WSDL) version 2.0 part 0: Primer". World Wide Web Consortium, Recommendation REC-wsdl20-primer-20070626, June 2007.

[28] Mike P. Papazoglou, Marco Aiello, "Service-Oriented Computing and Software Agents", 2004 Kluwer Academic Publishers.

[29] Norm Archer, Junlian Xiang. "Intelligent Agent Systems in Healthcare", World Congress on the Management of eBusiness, 2006

[30] A. Newell. "The knowledge level AI", 18:87–127,1982.

[31] Labrou Y. Mayfield, J Finin, "KQML as an agent communication language." In J. M. Brad- shaw (Ed.), Software agents (pp. 291-316). CA: AAAI Press., 1997.

[32] Kazuhiro Fuchi, Toshio Yokoi, "KQML:an information and knowledge exchange protocol", Knowledge Buidling and Knowledge Sharing, 1994

[33] Rich Fritzson, Tim Finin, Don Mckay, Robin McEntire, "The KQML Information and Knowledge Exchange Protocol", In Third International Conference on Information and Knowledge Management, 1994.

[34] Hoon Jin, In-Cheol Kim, "A BDI Agent Approach to

Healthcare Web Services", IFBME Proceedings, World Congress on Medical Physics and Biomedical Engineering 2006, August 27 – September 1, 2006 COEX Seoul, Korea "Imaging the Future Medicine", Springer Berlin Heidelberg, 2007

[35] Tuthill, G.S., Levy S.T., Knowledge-based systems: A manager's perspective. TAB Professional and Reference Books.

[36] Lizhen Liu, Hantao Song, Ling Bai, "Agent-based integration of heterogeneous database system", Proceedings of the Ninth International Conference on Computer Supported Cooperative Work in Design, Volume 2, Issue 24-26, Page(s) 1034 - 1037, 2005

[37] O.L.M de Faris, Santos, N.. "An agent-based geographical information system", International Conference on Web Technologies and Internet Commerce. Volume 1, Issue , 28-30 Nov. 2005 Page(s):998 – 1005, 2005

Fang Cao is currently a Ph.D. Candidate in Department of Computing & Software, McMaster University, Hamilton, Canada. He received his Master Degree in Applied Science in Computer Science at McMaster University in 2006, and Bachelor degree of Science at University of Ottawa in 2004. His research area is software engineering methodology for improving electronic health record interoperability.

Dr**. Norm Archer** is a Professor Emeritus in the DeGroote School of Business, McMaster University, Canada, and a Special Advisor to the McMaster eBusiness Research Centre. He is an Associate Member of the Computing and Software Department. Norm and his graduate students and colleagues are intensively involved in studies of the adoption and use of electronic health records in Canada, including issues of interoperability, electronic medical record system adoption by physicians, and the impact of personal health record systems on health self management by consumers. During his career, he has published over 100 refereed journal articles and conference papers.

Dr**. Skip Poehlman** is an Associate Professor in the Computing and Software (CAS) Department. He teaches in both the Computer Science and the Software Engineering Programmes offered in the Department at both baccalaureate and graduate (Master's and Ph.D.) levels. As head of the Applied Computersystems Research Group (ACsG), he has graduated eight Ph.D. students and some 30 Master's level students. Currently, three Ph.D. candidates and one M.Sc. student labour in CS research areas including, Knowledge-based System Applications & Graphical User Interface (GUI) development, and BioEngineering involving eHealth issues and diagnostic imaging analysis. Collaboration research efforts with the McMaster e-Commerce Research Centre, (MeRC, including eHealth) and McMaster's Department of Medicine are on-going. Other support originates from Federal government grants. A number of peer-reviewed papers have been published in AI journals, Distributed Systems journals, IEEE Transactions and Computer-Human Interaction journals as well as related conferences. Skip is a member of the ACM, American Association of Artificial Intelligence (AAAI) and Canadian Information Processing Society (CIPS) as well as a Senior Member of the IEEE. Academically, also at McMaster University, he maintains associate memberships in other Engineering Departments such as Engineering Physics and Mechanical Engineering.