Sharing e-Health information through ontological layering.

Pavandeep Kataria
Radmila Juric

School of Electronics and Computer Science

Sharing e-Health Information through Ontological Layering

Pavandeep Kataria
University of Westminster
P.Kataria1@westminster.ac.uk

Radmila Juric
University of Westminster
Juricr@westminster.ac.uk

Abstract

e-Health information, including patient clinical and demographic data, is very often dispersed across various environments, which either generate them or retrieve them from different sources. Healthcare professionals often need related e-health information in order to obtain a more comprehensive picture of a patient's health status. There are many obstacles to retrieving information and data from heterogeneous sources. In this paper we show that our ontological layering helps in (a) classifying requests imposed by healthcare professionals when retrieving e-health information from heterogeneous sources and (b) resolving semantic heterogeneities across repositories and composing an adequate answer to issued requests. We use a layered software architectural model based on Generic ontology for Context-aware, Interoperable and Data sharing (Go-CID) software applications, applicable to e-Health environments. Ontological layering and reasoning have been demonstrated with semantic web technologies.

1. Introduction

Modern e-health systems extend the traditional delivery of healthcare in primary care and hospitals towards personalized self-care, home care or any other healthcare service which can be delivered at any time and any place. Their operational environments are based on collaboration between patients, who require personalized health services and healthcare practitioners who deliver them. However, the emphasis is on sharing and exchanging the expertise amongst healthcare specialists, and more importantly on sharing knowledge, information and any other clinical and demographic data across heterogeneous e-health environments. This paper focuses on the problem of data sharing from various systems in order to manage patient health. We aim to assess the feasibility of sharing information across heterogeneous data sources and achieving various levels of interoperability in modern e-health systems.

We have been motivated in our work by e-health research [1,2], our government's initiative [3], and the European Union declarations [4]. The motto is "to educate and inform health care professionals, managers and consumers; to stimulate innovation in care delivery and health system management" [5]. More precisely, interoperability and interconnection between various e-health systems, patient and professional mobility; and international cooperation [4] have become essential in the delivery of modern healthcare. We intend to disseminate to healthcare professionals our experiences of building e-health systems, through the sharing of e-health data, in order to connect them with patients and make informed decisions based on up-to-date information available across e-health systems.

In this paper we propose a solution for sharing e-Health data, based on ontologies, reasoning upon them and their layering. We use a software architectural model for **G**eneric **o**ntology for **C**ontext aware, **I**nteroperable and **D**ata sharing (Go- CID) software applications [6, 7]. The model assists in the retrieval of heterogeneous data, enables their semantic interoperability and sharing. Our scenario is in an e-health operational environment, where a healthcare professional requests the retrieval of heterogeneous data about a particular patient in order to obtain a comprehensive picture of the patient's health. However, it is important to note that:

a) we focus on semantic heterogeneities between data structures in highly structured environments, which are common in e-health systems. We leave problems of synchronizing the results of heterogeneous retrievals, with sensor's derived data and multimedia data streams, outside the scope of this paper;
b) we focus on semantic technologies and leave old fashioned solutions of mediation and wrapping outside our scope. Our long term goal has been to assess how successful semantic technology can be in addressing the complex problem of data sharing and interoperability in e-health environments.
c) we focus on ontological solutions in terms of building layers which can manage semantic heterogeneities. The layers do not necessarily support SQL-like software applications, i.e .they are open to any other means of data retrievals.

Section 2 gives the Scenario of an e-health environment, which issues a request to obtain a comprehensive picture of a patient's health status. In section 3 we apply the Scenario to Go-CID. We explain ontological layering, i.e, the creation of local and derived ontological concepts, their relations and reasoning upon them. Related works on ontologies in the field of interoperability and data sharing, including e-health, are given in section 4. We conclude and list our future works in section 5.

2. The Scenario

We consider a heterogeneous e-health environment, where a healthcare professional may require "*a comprehensive picture of a patient's health status*". Any information which is generated about the patient should be obtained. An example of information he/she may need is in Table 1:

Table 1: Semantics of Request obtaining: *"a comprehensive picture of patient's health status"*

i) The **Electronic Patient Record** (**EPR**) stored within the patient's **General Practitioner's** (**GP) system**;
ii) Patient's **medical histories** created in various **hospitals**, **clinics**, **health centers**, and stored in any format: from database records of hospital information systems to documents generated from medical records as results of patients consultations with healthcare professionals (i.e. the patient might have been treated at various hospitals/health centers).
iii) Results of various tests (**test-results**), which could have been carried out in any **clinic** and outside the healthcare practitioner's location. For all three retrievals (i)-(iii) healthcare professionals use **unique** and/or **common** identifiers to retrieve a correct record.

Obtaining "*a comprehensive picture of a patient's health status*" would require the retrieval of a variety of data and information, which may range from the selection of relational database records to searching for unstructured documents on the web. The request itemized in i)-iv) from Table 1 would require that we know exactly: which data/information might be available about the patient and where such information may be stored. In order to answer this question and decide which exact data repositories are to be retrieved, we need to illustrate the request from Table 1. Therefore Table 2 is a specific example of the request: *"Dr. Smith would like to have a comprehensive picture of Mr. Jones's health status as a diabetic"* (we call it "`subject Jones`"), i.e. Table 2 is a specific instance of Table 1. The request from Table 2 is named request $\mathbf{R_j}$.

Table 2: Semantics of Request Rj for Go-CID ("Dr. Smith would like to have a comprehensive picture of Mr. Jones's health status as a diabetic")

1. Stored in the *medical summary* within database DB_{GP}, of Mr. Jones's **EPR** created during his *latest visit* to his **GP**. Dr. Smith is interested in the *latest health complaints* and Mr. Jones might want to see decisions on treatments and medications that have been made by his **GP**. (Dr. Smith can retrieve Mr. Jones's *medical summary* and the *latest visit / health complaints record* by typing in Mr. Jones's s*urname* and his *Patient_No)*
2. Stored in *medical record* from **medical histories** within database DB_{loc} in Dr. Smith's **hospital**, where Mr. Jones was treated earlier (Dr. Smith can retrieve Mr. Jones's *medical record* by typing in Mr. Jones's l*ast name* and *Patient-ID)*
3. Stored in the *test carried out* records and *latest health complaint* from database $DB_{X\text{-}Ray}$ from **clinic** (such as *X/Ray* clinic) under Mr. Jones's *surname and Patient_ID* (Dr. Smith can retrieve Mr. Jones's *test carried out* and the *latest health complaint* record by typing in Mr. Jones's s*urname* and his *Patient_ID).* The *results of tests carried out* are related to the *latest health complaint*.
4. Stored in documents on web site "*Get moving and fight diabetes*" (the contents of the web site are stored in DB_{web}) from the local interest group, which supports diabetics in their everyday life. Dr. Smith is interested in *sport/activities* applicable to Mr. Jones's condition and in any other information he may have found relevant to Mr. Jones's *activities* within such a group.

Table 2 contains a detailed semantics of the request $\mathbf{R_j}$ in terms of specifying which kind of information is needed to carry out the request on "`subject Jones`". However, grey shaded words are the results of assertions on the e-healthcare environment taxonomy; generated from the semantics stored in Tables 1 and 2. If we wanted to find out which repositories are to be retrieved when carrying out the request on *"*`subject Jones`*"* then the following taxonomy and assertions are to be used:

- FUNCTIONAL
- **- RETRIEVAL**
  - **-DATABASE/INFORMATION REPOSITORIES**
    - **- LOCAL**

- **HOSPITAL**
  - *DBloc*
- **NON-LOCAL**
  - **GP SYSTEM**
    - *DBgp*
  - **CLINIC**
    - *DBx-ray*
  - **HEALTH CENTER**
  - **SOCIAL SYSTEM**
  - **CARE SYSTEM**
  - **WEB**
    - *DBweb (content of URI)*
- **PATIENT HEALTH STATUS**
  - **PERCEPTION 1 "COMPREHENSIVE"**
    - **ELECTRONIC PATIENT RECORDS**
      - *medicalSummary*
      - *latestHealthComplaint*
      - *latestVisit*
    - **MEDICAL HISTORY**
      - *medicalRecord*
      - **TEST RESULTS**
        - *x-rayImage*
      - **RELEVANT INFORMATION**
        - *sports-activities*
  - **PATIENT RETRIEVAL POINTS**
    - **UNIQUE IDENTIFIERS**
      - *patientID*
      - *patientNO*
    - **COMMON IDENTIFIERS**
      - *patientName*
      - *patientLastName*
      - *patientSurname*
      - *patientDOB*

The semantics from Table 1 give the basic categorization of the taxonomy above (bold-faced words). The semantics from Table 2, (*"subject Jones"*) gives taxonomical items (italic words) that belong to the items behind the basic categorisation. Basic categorisation and taxonomical items are self` explanatory. Note: the PERCEPTION category in the taxonomy models the healthcare professional's perception of the e-healthcare environment. For example: a functional requirement- "I want the retrieval of Patient Medical History" may be interwoven with a non-functional requirement: "I want the retrieval Patient's Medical History to be displayed in one window, by the most recent year first" if a professional specifies the exact manner in which he/she wanted the information [8].

We give one example of the assertion: it determines that database *DBloc* should be retrieved when carrying out the request on *"subject Jones"*. The combination of the *medicalSummary, latestHealthComplaint* and *latestVisit* taxonomical items from **ELECTRONIC PATIENT RECORDS** class from the **PERCEPTION 1 "COMPREHENSIVE"** sub-class under the **PATIENT HEALTH STATUS** super-class, would imply that the **FUNCTIONAL** super-class of **LOCAL** and **HOSPITAL** classes from the **RETRIEVAL** sub-class in which taxonomical item *DBloc* is placed, will be used for such retrieval. This assertion would suggest that in order to carry out the request on *"subject Jones"*, *DBloc* is one of the repositories which must be retrieved. Alternative assertions could be made about any other repository, i.e. *DBloc, DBgp, DBx-ray, DBweb* are also additional repositories to be retrieved if we want to carry out the request on *"subject Jones"*.

To summarize: the request issued by Dr. Smith, itemized in $\mathbf{R_j}$, specifies the exact functionality of the application which runs in Dr. Smith's operational environment, and which ultimately gives "*a comprehensive picture of a Mr. Jones's health status*". By performing assertions upon Dr. Smith's e-health environment taxonomy, we generate another request $\mathbf{R_k}$, which specifies that *DBloc*, *DBgp, DBx-ray, DBweb* are repositories to be retrieved if we want to carry out the request on *"subject Jones"*.

3. The Outline of Go-CID

We describe a Software Architecture (SA), which accommodates Go-CID software applications, and makes provisions for managing the retrieval of heterogeneous information across a variety of sources. Go-CID was introduced in [7] and its improved version was compared to an SA for retrievals of heterogonous medical sources in [6].

SA from Figure 1 is a layered model, based on software architectural styles principles [9, 10] and exploited in other models, which use architectural layering to address interoperability in software systems [11, 12]. A set of repositories $\{DB_1 \ldots DB_n\}$ at the Persistence Layer are heterogeneous sources of data /information, generated by various parties, users and applications. We use the symbol of a database here, but that does not mean that they are traditional databases. We may have multimedia data streams of medical images, data generated by user's inputs and data which has already existed in structured or semi-structured formats (relational/web data etc.)

An application $\mathbf{App_i}$, from a set of software applications $\{App_1 \ldots App_n\}$ placed in the Application Layer, issues requests $\{\mathbf{R_j} | \ i=1 \ldots n\}$, similar to request for "*subject Jones*" from the Scenario. The functionality of $\mathbf{App_i}$, includes the retrieval of data from repositories $\{DB_1 \ldots DB_n\}$.

It is important to note that $\mathbf{App_i}$ is perceptive to a variety of requests $\{R_1 \ldots R_n\}$, which may be generated by users, and which are stored in a separate

Requirements Layer. For example the user, Dr. Smith, is issuing the request on "`subject Jones`". Therefore, "`subject Jones`" has an impact on an **App$_i$**, because it may decide which repositories are relevant for **App$_i$** and which functionality within **App$_i$** is needed for carrying out the request on "`subject Jones`".

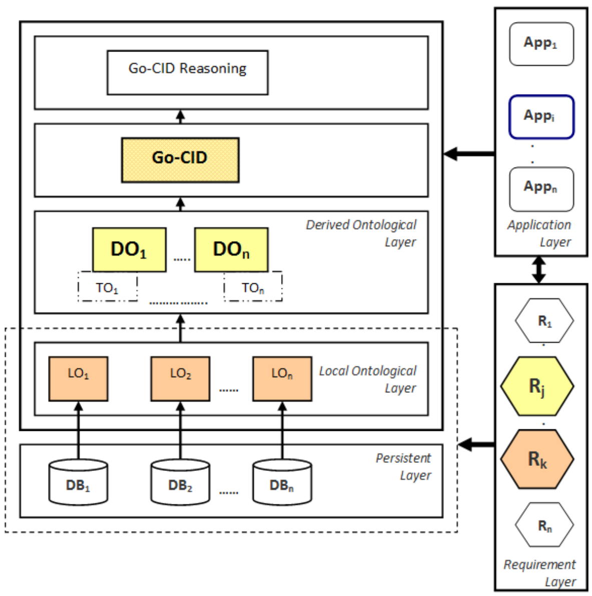

Figure 1: Go-CID Software Architecture

The Go-CID architectural layers store different ontologies. The ontological layering starts with a set of local ontologies {LO_1 … LO_n} generated directly from repositories {DB_1… DB_n}. **LO$_i$** represent the underlying schemas of **DB$_i$** in an ontological format, i.e. the semantics from database schemas are transferred into ontological concepts of **LO$_i$** .

If the "`subject Jones`" request is **R$_j$**, then **R$_j$** automatically generates another request **R$_k$**, which stores the semantics about the environment where **App$_i$** resides, i.e. **R$_k$** describes the environment in terms of specifying "which local ontologies from {LO_1 … LO_n} are relevant to request **R$_j$** and ultimately needed by application **App$_i$**."

Therefore, we start with a set of local ontologies chosen through the semantics stored in **R$_k$** and we move upwards through the Go-CID layers by exploiting the semantics stored in **R$_j$**. The backbone of ontological layering is in 'Ontology Mapping', as defined in [13]. We use their definition that "mapping ontologies O_1 and O_2, means that for each entity (concept C, relation R, or instance I) in ontology O_1, we try to find a corresponding match, which has the same meaning, in ontology O_2".

3.1. Ontology Mapping in Go-CID

Ontology mapping in Go-CID happens for different purposes at different layers, as explained in 1)-3) below:

1) We create target ontologies {TO_i | i=1 … n} as a consequence of *alignments* of local ontologies {LO_i | i=1 … n}. The alignment is triggered by the fact that underlying data repositories {DB_1… DB_n} (i.e. their local ontologies {LO_1 … LO_n}) will contain 'similar' and 'semantically-related' concepts.

Note that our definition of 'similar' and 'semantically-related' concepts overlaps with works of [14], [15] and [16]. Concepts are 'semantically-related' if they are a set of synonyms or hyponyms which are based on:

- overlapping ontological classes available at the conceptual level, or
- unique concepts which denote the sub-category of a more general ontological class that is not overlapping with existing concepts, but contains semantics which are related to them.

Concepts are 'similar' if they share conflicting data values because of their conflicting data representation such as: different structural units and different label/names. Examples are (i) synonyms, homonyms, misspelled names at the table, identifier and attribute levels, and (ii) schema conflicts which include union incompatibility, isomorphism, aggregation and generalization/ specialization. We use their terminology of "synonyms and hyponyms" to denote that there are semantic conflicts across heterogeneous databases. The semantic conflicts show that there might be synonyms at e.g. the attribute levels where PATIENT.LAST_NAME in one database probably describes the same semantic concept as PATIENT.SURNAME in another database. Their hyponyms correspond to traditional schema conflicts in databases in terms of having isomorphism, when the database concepts with the same semantics are described through a different number and type of attributes in different schemas. The same applies to generalization/specialization conflicts. Some of these semantic conflicts have been elaborated in [17] and [18].

Therefore the semantics in request **R$_k$**, picks the correct local ontologies which are to be 'aligned' according to the semantics in request **R$_j$**. The alignment ensures that:

- we understand the semantics of 'similar' ontological concepts in {LO_1 … LO_n} and
- we *infer* the correct set of matches between *a pair* of ontological data instances belonging to 'similar' ontological concepts from {LO_1 … LO_n}. They

will ultimately secure the successful execution of request $\mathbf{R_j}$.

2) We create derived ontologies {DO_i | i=1 … n} as a result of the *integration* of target ontologies $\mathbf{TO_m}$ and $\mathbf{TO_n}$ from {TO_i | i=1 … n}. This integration depends on the specificity of request $\mathbf{R_j}$ (`"subject Jones"`). This ensures that:

- we understand the semantics of *aligned* ontological data instances from $\mathbf{TO_m}$ and $\mathbf{TO}_n$,
- we *assert* the correct set of matches between *a pair* of ontological classes belonging to 'semantically-related' ontological concepts from $\mathbf{TO_m}$ and $\mathbf{TO}_n$,
- we *model* the relationships between *aligned* ontological data instances which belong $\mathbf{TO_m}$ and $\mathbf{TO_n}$.

3) We *merge* $\mathbf{DO_i}$ ontologies from {DO_i | i=1 … n} into a final ontological model **Go-CID**. This merge ensures that:

- we understand the complete set of semantics from aligned ontological data instances in the {TO_i | i=1 … n} and the complete set of semantics in derived ontological relationships from {DO_i | i=1 … n} and
- we generate the correct set of ontological classes which accommodate derived ontological relationships from {DO_i | i=1 … n} and aligned ontological data instances from {TO_i | i=1 … n}.

To summarise: Go-CID ontology has resulted from local ontologies' alignments into target ontologies, integration of target ontologies into derived ontologies and the merging of derived ontologies into the final Go-CID for a particular request $\mathbf{R_j}$ (`"subject Jones"`). However, the details of alignment, integration and merging can be given only if we use the semantics stored in a particular request $\mathbf{R_j}$ which is given in section 4.

4. Sharing e-Health Information though Go-CID

4.1. An instance of Go-CID

In this section we apply the Scenario to illustrate the Go-CID layering as described in 1)-3) from section 3. Thus, Fig. 2 illustrates the application of the scenario to Go-CID, and is an instance of Fig 1. We have 4 local ontologies, which create 3 target ontologies, which are in turn are integrated into two derived ontologies. Derived ontologies are merged into Go-CID.

Request $\mathbf{R_j}$, from Table 2 gives $\mathbf{R_k}$, which is the result of taxonomies/assertions explained in the Scenario. In other words chosen repositories shaded grey in Table 2**:** $\mathbf{DB_{loc}}$, $\mathbf{DB_{GP}}$, $\mathbf{DB_{X\text{-}R}}$ and $\mathbf{DB_{web}}$ comprise $\mathbf{R_k}$. Local ontologies $\mathbf{LO_{loc}}$, $\mathbf{LO_{GP}}$, $\mathbf{LO_{X\text{-}R}}$ and $\mathbf{LO_{web}}$ mirror the semantics contained in $\mathbf{DB_{loc}}$, $\mathbf{DB_{GP}}$, $\mathbf{DB_{X\text{-}R}}$ **and** $\mathbf{DB_{web}}$. They are created automatically, thus we detail the target/derived/Go-CID ontologies in the next three subsections.

4.1.1. Creating Target Ontologies

We *infer* the correct set of matches between *a pair* of ontological data instances that are 'similar' ontological concepts from $\mathbf{LO_{loc}}$, $\mathbf{LO_{GP}}$, $\mathbf{LO_{X\text{-}R}}$ and $\mathbf{LO_{web}}$. We use the semantics stored in request $\mathbf{R_j}$ as shown in Table 2 to trigger the alignments of ontologies in terms of finding:

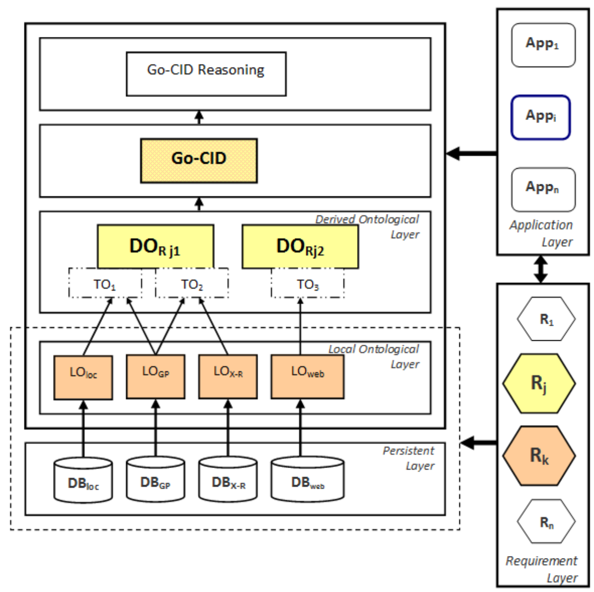

Figure 2: Illustrating Go-CID (Applying the scenario)

- Matches between individual instances for Mr Jones's 'name', his 'PatientID/NO' and his 'Medical History/Summary' across the LOloc, LOGP, and LOX-R local ontologies. These matches are triggered by naming conflicts which exist across LOs and which are carried forward from their underlying databases. The individual instances chosen to be aligned are dictated by the semantics from $\mathbf{R_j}$ (Table 2).
- A match is inferred using the object property: *loloc_patientID* connecting individual instance: *patientID* belonging to the LOLOC_PATIENT class in $\mathbf{LO_{loc}}$ and the object property: *logp_patientNO* connecting individual instance: *patientNO* to the LOGP_PATIENT class in $\mathbf{LO_{GP}}$,
- A match is inferred using the object property: *loloc_patient_lastname* connecting individual instance: *patient_lastname* belonging to the

LOLOC_PATIENT class in $\mathbf{LO_{loc}}$ and the object property: *loxr_patient_surname* connecting individual instance: *patient_surname* to the LOXR_PATIENT class in $\mathbf{LO_{XR}}$,

- A match is inferred using the object property: *loloc_medical_history* connecting individual instance: *medical_history* belonging to the LOLOC_MEDICAL_HISTORY in $\mathbf{LO_{loc}}$ and the object property: *logp_medical_summary* connecting individual instance: *medical_summary* to the LOGP_MEDICAL_SUMMARY in $\mathbf{LO_{GP}}$.

The alignment process includes the creation of a ‘bridging axiom’ between ‘similar’ ontological data instances. The ‘bridging axiom’ describes how one ontological data instance can be mapped into another whilst being stored separately (on an ‘ad hoc basis’) from the local ontologies they belong to. An example of mapping in the alignment process would be the use of the Semantic Web Rule Language (SWRL) [17] rules to infer ‘similar’ matches between ontological data instances, using comparison operations in predicate logic, i.e. whether a datatype property in a local ontology is EQUAL or NOT EQUAL to a datatype property in the local ontology. Matches between $\mathbf{LO_{loc}}$, $\mathbf{LO_{GP}}$, $\mathbf{LO_{X\text{-}R}}$ and $\mathbf{LO_{web}}$ are used for the alignment of these ontologies, which stores results in the generation of target ontologies $\mathbf{TO_i}$:

- aligned individual instances: *patientID* and *patientNO* give an inferred class: TO1_PATIENT from $\mathbf{TO_1}$,
- aligned individual instances: *patient_lastname* and *patient_surname* give an give an inferred class: TO2_PATIENT from $\mathbf{TO_2}$,
- aligned individual instances: *medical_history* and *medical_summary* give an inferred class: TO3_ MEDICAL_HISTORY_SUMMARY from $\mathbf{TO_3}$.

4.1.2. Creating Derived Ontologies

We *assert* the correct set of matches between *a pair* of ontological classes belonging to ontological concepts from target ontologies and local ontologies to perform ontology integration. Therefore, we *assert* the correct set of matches from $\mathbf{TO_1}$ and $\mathbf{TO_2}$ in terms of finding overlapping concepts, i.e. a match is inferred between the ontological class: TO1_PATIENT from the target ontology $\mathbf{TO_1}$ and the ontological class: TO2_PATIENT from $\mathbf{TO_2}$.

We also *assert* the correct set of matches between target ontology $\mathbf{TO_3}$ and local ontology $\mathbf{LO_{X\text{-}R}}$ in terms of finding unique concepts, i.e. a relation is modeled between the object property: *loxr_health_complaint* from the local ontology $\mathbf{LO_{X\text{-}R}}$ and the ontological class: TO3_MEDICAL_HISTORY_SUMMARY from the target ontology $\mathbf{TO_3}$.

The integration process includes the creation of a ‘taxonomical axiom’ between ‘semantically-related’ ontological classes. The ‘taxonomical axiom’ asserts how one ontological class is related to another from the target and local ontologies they belong to. An example of relating two target ontologies in the integration process would be the use of SWRL rules to assert matches (we can also name them ‘semantic’ matches) between ontological classes through their associated object properties. SWRL rules in the ontology integration process are different to the SWRL rules in the alignment process because:

a) they use the result sets of SWRL rules from the alignment process, i.e. the inferred individual instances (including the ontological classes that they belong to) are dealt with, and
b) they use SWRL rules at an object property level, i.e. relations are asserted at both target and local ontological levels.

Thus, matches between $\mathbf{TO_1}$, $\mathbf{TO_2}$, $\mathbf{TO_3}$ and $\mathbf{LO_{X\text{-}R}}$ are used for the integration of these ontologies, which results in the generation of $\mathbf{DO_i}$:

- aligned ontological class: TO1_PATIENT from $\mathbf{TO_1}$ uses the object property: *patient_has* to create a relationship between TO1_PATIENT and the integrated ontological class: DORJ1_PATIENT from derived ontology $\mathbf{DO_{Rj1}}$. TO2_PATIENT from $\mathbf{TO_2}$ also uses object property: *patient_has* to create a relationship between TO2_PATIENT, and the integrated ontological class: DORJ1_PATIENT from derived ontology $\mathbf{DO_{Rj1}}$
- aligned ontological class: TO3_MEDICAL_HISTORY_SUMMARY from $\mathbf{TO_3}$ uses object property: *patient_details* to create a relationship between TO3_MEDICAL_HISTORY_SUMMARY and integrated ontological class: DORJ2_PATIEN from $\mathbf{DO_{Rj2}}$.
- object property: *loxr_latest_health_complaint* from $\mathbf{LO_{X\text{-}R}}$ is used to create a relationship between $\mathbf{LO_{X\text{-}R}}$ and DORJ2_PATIENT from $\mathbf{DO_{Rj2}}$ (this integration skips alignment).

4.1.3. Creating Go-CID Layer

To provide “*a comprehensive picture of a Mr. Jones’s health status*”, the derived ontologies $\mathbf{DO_{Rj1}}$ and $\mathbf{DO_{Rj2}}$ are merged into **Go-CID,** i.e. the complete set of ontological classes and properties from $\mathbf{DO_{Rj1}}$ and $\mathbf{DO_{Rj2}}$ are merged together. Reasoning rules are used to create a single, coherent ontology **Go-CID.** The merged ontology will produce a translation of all the source ontologies (local and derived ontologies) into **Go-CID**, i.e. **Go-CID** should include information from all original sources without changing their underlying databases..

4.1.4. Examples of SWRL Rules

Due to shortage of space, we show a few SWRL rules to illustrate the alignment, integration and merging of our ontologies. Examples of SWRL rules used during the alignment process are in Tables 2, 3,4.

Table 3: Alignment Process: SWRL Rule 1

LOLOC_PATIENT (?p) ∧ loloc_patientID (?p, ?r) ∧ LOGP_PATIENT (?s) ∧ logp_patientNO (?s, ?t) • TO1_PATIENT (?r) ∧TO1_PATIENT (?t)

SWRL Rule 1 in Table 3 shows that the object properties: 'loloc_patientID' and 'logp_patientNO' are used to determine a match by transferring individual instances: 'patientID' from the LOLOC_PATIENT in $\mathbf{LO_{loc}}$ and 'patientNO' from the LOGP_PATIENT class in $\mathbf{LO_{GP}}$ to the ontological class: TO1_PATIENT class T$\mathbf{O_1}$. Therefore we infer individuals into the TO1_PATIENT class.

Table 4: Alignment Process: SWRL Rule 2

LOLOC_PATIENT (?a) ∧ loloc_patient_lastname (?a, ?b) ∧ LOXR_PATIENT (?c) ∧ loxr_patient_surname (?c, ?d) • TO2_PATIENT (?b) ∧TO2_PATIENT (?d)

SWRL Rule 2 in Table 4 shows that the object properties: 'loloc_patient_lastname' and 'loxr_patient_surname' are used to determine a match by transferring individual instances: 'patient_lastname' from the LOLOC_PATIENT in $\mathbf{LO_{loc}}$ and 'patient_surname' from the LOXR_PATIENT class in $\mathbf{LO_{XR}}$ to the ontological class: TO2_PATIENT class T$\mathbf{O_2}$. Therefore we infer individuals into the TO2_PATIENT class.

Examples of SWRL rules used during the integration process can be seen in Tables 4 and 5.

Table 5: Integration Process: SWRL Rule 3

TO1_PATIENT(?c) ∧ patient_has (?c,?f) ∧ TO2_PATIENT(?f) → DORJ1_PATIENT: patient_details (?c,?f)

SWRL Rule 3 in Table 5 shows that an asserted relationship between ontological class TO1_PATIENT and TO2_PATIENT uses the object property: patient_has as an overlapping concept to assert a relation between ontological class: DORJ1_PATIENT and object property: patient_details stored in $\mathbf{DO_{Rj1}}$.

Table 6: Integration Process: SWRL Rule 4

TO3_MEDICAL_HISTORY_SUMMARY (?i) ∧ loxr_latest_health_complaint (?j) → DORJ2_MEDICAL_RECORD : references_patient (?i,?j)

The integration SWRL rule 4 in Table 6 shows that: an asserted relationship between ontological class TO3_MEDICAL_HISTORY_SUMMARY and object property: loxr_latest_health_complaint implies a relation between ontological class: DORJ1_PATIENT and object property: patient_details stored in $\mathbf{DO_{Rj2}}$.

5. Related Work

Ontologies are used across a variety of business and scientific communities as a way to share, reuse and process domain knowledge, which is central to many applications such as information management, electronic commerce, semantic web services, scientific knowledge portals etc. Using ontologies to address interoperability is proving to be successful in providing a broader range of information and contexts through shared semantics and syntax. Behaving like reference trees, semantic and syntactic conflicts can be masked through mapping techniques to resolve heterogeneities [18, 19, 20, 21, 22, 23]. We allow ontologies to behave like enriched data-centric models that provide a means to deal explicitly with semantic data interoperability challenges. The examples of using ontologies in such a manner have been elaborated in various works [24, 25, 26, 27, 28, 29, 30, 31] thus enabling a common understanding of the structure of the information /data across heterogeneous environments.

It has been difficult to find a similar work which deal with retrievals of heterogeneous sources through applications (i) perceptive to the environments where they happen to reside and (ii) with users who may have an impact on the way applications are run and on the choice of sources applications may need. This is true for many problem domains, including healthcare. However these applications from (i) and (ii) are heterogeneous in their nature and by resolving the issue of their semantic conflicts, we may be able to address better the problem of interoperability. In

5.1-5.2 we overview works which use ontologies when dealing separately with data sharing and interoperability. We could not find works which address retrievals of heterogeneous data sources using ontological layering. There are works which deal separately with the problem of resolving semantic conflicts. They do not use ontological layering and are concerned with the semantic heterogeneities of relational database schemas.

5.1. Ontologies and data sharing

There are many examples where ontological models are used as a way of providing a common shared repository of knowledge, thus allowing the sharing of data/knowledge across different computing environments. However, most of them propose data integration systems that provide a common knowledge base, through a single ontological model used as a core component, to enable data and knowledge sharing from different domains.

BioMediator ontology [32] provides a common interface to Web-accessible sources of biologic information. It is driven by information stored in a Protégé knowledge-base and provides data integration over multiple structured /semi-structured biologic data sources. Queries are run on the knowledge-base to run retrievals across semantically/syntactically heterogeneous data.

The European project for Standardized Transparent Representations in order to Extend Legal Accessibility [33] proposes a shared data oriented platform based on an ontology, which allows public administrations to develop legal knowledge management solutions. They use a Legal Knowledge Interchange Format (LKIF) as the core modeling language for the ontology. LKIF is built upon existing XML-based standards, including RDF and OWL. The LKIF is stored as a Protégé 2000 knowledge base, supported by its Application Programmer Interfaces (APIs), thus, allowing data sharing amongst other legal knowledge-bases.

The ontology from [34] is used for organizing and sharing large sets of data, produced by Reactive Oxygen Species (ROS) signaling networks in plants. The ontological model formalizes the data sets according to a shared set of agreed concepts. The ability to reason, upon the semantics within such concepts, is derived through the syntax of the terminology.

5.2. Ontologies and Interoperability

In spite of a few attempts to introduce semantic interoperability and semantic technologies into health services it is too early to see ontological models, which deal with heterogeneities in the healthcare domain. The three examples below give a picture of three different ontological solutions, which are used for alleviating database interoperability in general. Sung and McLeod propose a schema matching framework for the identification of a set of correct matches between relational schema attributes. An ontology is used to replicate the relational schemas so that the semantics behind each attribute can be extracted through the form of ontological classes and instances. Similar matching semantics from the relational schemas are derived using an ontology driven semantic-matching technique to identify common quantifying parent classes of similar semantics of underlying data attributes.

An ontology approach for the reuse of relational sources in the context of semantic-based access to information from [36] is used for i) extracting the semantics hidden in relational sources by wrapping them into ontological concepts within an ontology and ii) understand the methodology for semantic extension of such ontologies through reasoning upon ontological classes derived. Extracting hidden semantics contained in relational sources are captured through the use of associating views over data source to elements of the extracted ontology. Heuristics rules are applied based on ideas of standard relational schema design and normalization to reduce data loss and to preserve the semantics of constraints in the database.

The Relational DataBase Ontology (RDBO) from [37] is used for resolving semantic conflicts between RDBSs automatically while allowing the individual RDBSs to evolve. RDBO is based on ontological classes that make up the semantic descriptions of the individual RDBSs. Each ontological class conforms to a set of vocabularies, structures, and restrictions that are commonly agreed upon by participating RDBSs. A reasoning engine is used to validate and infer additional semantic relationships from the existing relationships. To resolve semantic conflicts, terms defined in different databases ontologies are compared to each other semantically using semantic weights and the reasoning engine.

6. Conclusions

In this paper we demonstrate how a SA for Go-CID software application assists in the retrieval of heterogeneous e-health information. We use ontological layering, i.e. alignment, integration and merging of ontologies through reasoning rules, in order to resolve semantic heterogeneities of e-health information. The ontological models were implemented in Protégé 2000 Ontology Editor, version 3.4. The Pellet Reasoner was used to test the consistency of the ontology's classes /properties/ instances. The SWRL rules were implemented through the SWRLTab plugin in Protégé, and executed using the Jess Rule Engine (Jess Engine).

We proved with this and previous works that the Go-CID ontological layering in e-health environments is re-usable, i.e. the SA for Go-CID applications is not domain specific. We have used it for the semantic management of requirements in pervasive systems and for creating situation aware software applications, which depend on a multitude of devices, user preferences and the delivery of semantic services in modern computing (6,7,8).

We also proved that Go-CID layering is dictated by the semantic of requests imposed on heterogeneous e-health systems. The request on "subject Jones" dictates ontological alignment, integration and merging. The originality is in both: classifying the content of requests (as in (a) from the abstract) and creating ontological layers according to the semantics in them and semantic conflicts across the retrieved sources (as in (b) from the abstract).

The contribution of this paper is twofold:

We deal with semantic interoperability by carrying forward semantic conflicts from one or more databases and deducing logic/inferring knowledge about conflicts in order to resolve them (section 3.1). This is done through ontological layering (sections 4.1.1 and 4.1.2), as a result of ontological alignment, integration, and merging, based on reasoning through SWRL rules. This is where our layering is unique: it resolves semantic conflicts through reasoning and without affecting original repositories, i.e. without imposing any change on them.

Our ontological layering is capable of addressing heterogeneities without using a variety of algorithms and key-words matching [35] based on a "weighting" to infer knowledge about these environments. We exploit semantics of underlying databases through ontological reasoning which enables us to resolve different semantic conflicts at a different ontological layers. This automatically excludes SQL like joint queries and traditional approaches to retrievals of heterogeneous data repositories.

We are improving our reasoning mechanism within ontological layering by a) evaluating the way we infer new data types (i.e. are SWRL built-in operators enough?); b) testing our assertions upon semantic matches (i.e. what is the best combination of rule sets or object properties?). The work on automating assertions upon the e-healthcare environment taxonomy, given in section 2, is a part of our separate research on semantic management of requirements in pervasive healthcare, which is being used for demonstrating Go-CID. Our urgent task is to evaluate the performance of software applications built upon Go-CID and its accessing mechanisms to ontological concepts from applications. We also plan to use our Go-CID SA as a framework for creating a tool similar to Conflict Resolution Environment for Autonomous Mediation, as an aid for resolving semantic conflicts through ontological layering.

7. References

[1] Ahern, D.K., "Challenges and opportunities of eHealth research", American Journal of Preventive Medicine, 2007, Vol. 32, Suppl. 5, pp. 75-82.

[2] Armstrong, N., Hearnshaw, H., Powel, J., and Dale, J., "Stakeholder Perspectives in the Development of a Virtual Clinic for Diabetes Care: A Qualitative Study", Journal of Medical Internet Research, 2007, Vol. 9 No. 3, e23.

[3] "NHS e-Health Research should focus on Patients", from E-Health Insider, published on 14-Nov-2005, Available at: http://www.e-health-insider.com/News/1537/nhs_ehealth_research_should_focus_on_patients.

[4] "eHealth in a Europe without frontiers: Building New Initiatives - Working Together", The Portorož Conference Declaration, 7 May 2008: eHealth 2008, available at: http://www.ehealth2008.si/index.php?id=26&mid=25.

[5] Silber, D., "The Case for eHealth", Book Series: Studies in Health Technology and Informatics, IOS Press, Vol. 100/2004, pp.3-27.

[6] Ganguly, S., Kataria, P.,Juric, R., Ertas, A., and Tanik, M. M., "Sharing Information and Data across Heterogeneous e-Health Systems", Tele-Medicine and e-Health Journal, 2009, Vol. 15, Issue 5, pp.454-464.

[7] Kataria, P., Juric, R., and Madani, K., "Go-CID: Generic Ontology for Context Aware, Interoperable and Data Sharing Applications", proceedings of the 11th IASTED Conference on Software Engineering Applications, Cambridge, MA, US, November 19-21 2007, pp. 439-444.

[8] Koay, N., Kataria, P., Juric, R., Terstyansky, G., and Oberndorf, P., "Ontological Support for Managing Non-Functional Requirements in Pervasive Healthcare" proceedings of the 42nd Int. Conference on System Science, Hawaii, Big Island, US, January 5-8 2009, CD ROM.

[9] Bass, L., Clements, P., and Kazman, P., Software Architecture in Practice Addison and Wesley, 2nd Edition, SEI Series in Software Engineering, 2003.

[10] Clements, P., Kazman, R., and Klien, M., Evaluating Software Architectures: Methods and Case Studies, Addison and Wesley, SEI Series in Software Engineering, 2001.

[11] Juric, R., Kuljis, J., and Paul, R., "Software Architecture to Support Interoperability in Multiple Database Systems", proceedings of the 22nd IASTED Conference on Software Engineering, Innsbruck, Austria, 11-14 February 2004, pp. 71-77.

[12] Juric, R., Kuljis, J., and Paul, R., “Contextualising Components when Addressing the DB Interoperability”, proceedings of the 8th IASTED Conference on Software Engineering Applications, Boston, MA, USA, 9-11 November 2004, pp. 690-695.
[13] Marc, E., and Steffen, S., “QOM- Quick Mapping”, proceedings of the 3rd Int. Semantic Web Conference, Florida, USA, 20-23 October 2003, pp.683-697.
[14] Noy, N., “Semantic integration: A Survey of Ontology Based Approaches”, proceedings of the Int. Conference on Management of Data (ACM SIGMOD), Paris, France,13-18 June 2004, Vol. 33, Issue 4, pp. 65-70.
[15] Saleem, A., “Semantic Web Vision: survey of ontology mapping systems and evaluation progress”, Masters Thesis No:MCS-2006:13, Blekinge Institute of Technology, Sweden, 2006. Available at: http://www.bth.se/fou/cuppsats.nsf/all/a72fe543e31b0cb7c125722c007edcc8/$file/MCS-2006-13.pdf.
[16] Choi, N., Song, I., and Han, H., “A survey on Ontology Mapping”, SIGMOD Record, 2006, Vol.35, No.3, pp.34-41.
[17] Park, J., and Ram, S., “Semantic Conflict Resolution Ontology (SCROL): An Ontology for Detecting and Resolving Data and Schema-level Semantic Conflicts”, IEEE Transactions on Knowledge and Data Engineering, February 2004, Vol. 16, No. 2, pp.189-202.
[18] Sheth, A. P., “Changing Focus on Interoperability in Information Systems: From System, Syntax, Structure to Semantics”, in Interoperating Geographic Information Systems (M F Goodchild, M J Egenhofer, R Fegeas and C A. Kottman (eds), 1998, Kluwer, pp 5-10.
[19] Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosof, B., and Dean, M., “SWRL: Semantic Web Rule Language- Combining OWL and RuleML Overview and Guide”, W3C Member Submission 2004. Available at: http://www.w3.org/Submission/2004/SUBM-SWRL-20040521/.
[20] Hitzler, P., Krotzsch, M., Parsia, B., Patel-Schneider, P. F., and Rudolph, S., “OWL 2 Web Ontology Language Primer”, W3C Working Draft 2009. Available at: http://www.w3.org/TR/2009/WD-owl2-primer-20090421/.
[21] Wongthongtham, P., Chang, E., and Sommerville, I., “Software Engineering Ontology for Software Engineering Knowledge Management in Multi-site Software Development Environment”, Journal of Systems Architecture, 2006, Vol.52, Issue.11, pp.640-653.
[22] Tetlow, P., Pan, J. Z., Oberle, D., Wallace, E., Uschold, M., Kendall, E., “Ontology Driven Architectures and Potential Uses of the Semantic Web in Systems and Software Engineering”, W3C Editors Draft 2006. Available at: http://www.w3.org/2001/sw/BestPractices/SE/ODA/.
[23] McGuinness, D. L., and Harmelen, F. V., “OWL Web Ontology Overview and Guide”, W3C Recommendation 2004. Available at: http://www.w3.org/TR/owl-features/.
[24] Carroll, J.J., and Klyne, G., “Resource Description Framework”, W3C Recommendation 2004. Available at:http://www.w3.org/TR/rdf-concepts/.
[25] Gruber, T. R., “Toward principles for the design of ontologies used for knowledge sharing”, Journal of Human and Computer Studies, 1995, Vol. 43, Issue 5-6, pp.907-928.
[26] Alani, H., Kim, S., Millard, D.E., Weal, M.J, Hall, W., Lewis, P.H., and Hadbolt, N.R., “Automatic Ontology-Based Knowledge Extraction”, IEEE Intelligent Systems, 2003, Vol.18, Issue1, pp. 14-21.
[27] Razmerita, L., Angenhrn, A. and Maedche, A., “Ontology-based User Modeling for Knowledge Management Systems”, proceedings of 9th Int. Conference on User Modeling, Johnstown, PA, US, June 22-26 2003, pp.213- 217.
[28] Bicer, V., Laleci, G. B., Dogac, A., and Kabak, Y., “Providing Semantic Interoperability in Healthcare Domain through Ontology Mapping”, presented at “eChallenges” Ljubljana, Slovenia, 19-21 October 2005, available at http://www.srdc.metu.edu.tr/webpage/publications/2005/2005healthcareSemanticInteroperability.doc
[29] Dou, D., Lependu, P., Kim, S., and Qi, P., “Integrating Databases into the Semantic Web through an Ontology-based Framework” proceedings of the 22nd Int. Conference on Data Engineering Workshops, Atlanta, Georgia, USA, 3-8 April 2006, pp.54.
[30] Sidhu, A.S., Chang, E., and Sidhu, B.S., “ Protein Ontology: Vocabulary for Protein Data”, proceedings of the 3rd Int. Conference on Information Technology and Applications, Sydney, Australia, 4-7 July 2005, Vol. 2, pp.465-469.
[31] Peiguang, L., Xu, R., Lv, C., and Zhang, N., “An Ontology-based and Distributed Service for EG” proceedings of the Int. Conference on Internet Computing in Science and Engineering, Monte Carlo Resort, Las Vegas, Nevada, USA, 14-17 July 2008, pp.485-491.
[32] Mork, P., Donelson, L., Tarczy-Hornoch, P., Dolan, C., Mitchell, J.A., Barrier, M., and Mei, H., “The BioMediator System as a Data Integration Tool to Answer Diverse Biologic Queries”, Studies in Health Technology and Informatics, May 2004, Vol. 107 (Pt 2), pp.768-772.
[33] Boer, A., Winkels R., Vitali, F. “MetaLex XML and the Legal Knowledge Interchange Format” in P. Casanovas et al. (Eds.): Computable Models of the Law, LNAI 4884, 2008, pp. 21–41.
[34] Strizh, I.G., “Ontologies for data and knowledge sharing in biology: plant ROS signaling as a case study”, in Wiley InterScience, 2006, Vol. 28, Issue 2, pp. 199-210.
[35] Sung, S. and McLeod, D., “Ontology-Driven Semantic Matches between Database Schemas”, proceedings of the 22nd IEEE Int. Conference on Data Engineering, Atlanta, US, April 3-8, 2006, pp.6.
[36] Lubyte, L., “Reusing Relational Sources for Semantic Information Access”, proceedings of the ACM first Ph.D. workshop in Conference on Information and Knowledge Management, Lisbon, Portugal, 22 July 2007, pp.9-16.
[37] Trinh, Q., Barker, K., and Alhajj, R., “Semantic Interoperability Between Relational Database Systems”, proceedings of the 11th Int. Database Engineering and Applications Symposium, Banff, Canada, 6-8 September 2007, pp.208- 215.