Heterogeneity and Context in Semantic-Web-Enabled HCLS Systems*

Antoine Zimmermann, Ratnesh Sahay, Ronan Fox, and Axel Polleres

Digital Enterprise Research Institute
National University of Ireland, Galway, Ireland
firstname.lastname@deri.org

Abstract. The need for semantics preserving integration of complex data has been widely recognized in the healthcare domain. While standards such as Health Level Seven (HL7) have been developed in this direction, they have mostly been applied in limited, controlled environments, still being used incoherently across countries, organizations, or hospitals. In a more mobile and global society, data and knowledge are going to be commonly exchanged between various systems at Web scale. Specialists in this domain have increasingly argued in favor of using Semantic Web technologies for modeling healthcare data in a well formalized way. This paper provides a reality check in how far current Semantic Web standards can tackle interoperability issues arising in such systems driven by the modeling of concrete use cases on exchanging clinical data and practices. Recognizing the insufficiency of standard OWL to model our scenario, we survey theoretical approaches to extend OWL by modularity and context towards handling heterogeneity in Semantic-Web-enabled health care and life sciences (HCLS) systems. We come to the conclusion that none of these approaches addresses all of our use case heterogeneity aspects in its entirety. We finally sketch paths on how better approaches could be devised by combining several existing techniques.

1 Introduction

Health Care and Life Sciences (HCLS) has been one of the primary field of application for knowledge representation and reasoning systems. Several attempts have been made to standardize formal knowledge about medical data even before the development of Semantic Web technologies [1]. Nowadays, HCLS specialists increasingly argue in favor of Semantic Web technologies for representing medical and clinical knowledge [2]. However, current Semantic Web technologies alone are still too limited to provide a unified framework for all the varieties of applications and subdomains of this field. It must be remembered that researchers and practitioners in this domain have been facing—and somewhat dealing with—hetereogeneity problems for decades already, whereas Semantic Web technologies are a comparably new solution they started looking at. These problems arise from different formats used, different regional regulations or legislations.

* This work is partly funded by Science Foundation Ireland (SFI) project Lion-2 (SFI/08/CE/I1380).

R. Meersman, T. Dillon, P. Herrero (Eds.): OTM 2009, Part II, LNCS 5871, pp. 1165–1182, 2009.
© Springer-Verlag Berlin Heidelberg 2009

Existing standards are now used in, e.g., hospitals for representing information models, clinical repositories, ontologies for terminologies and vocabularies, patient records, local policies, and so on. Although standards have improved the management of very complex intertwined data, they show their limits when integrating and exchanging data between different systems. Interoperability problems are likely to occur in such a vast field as HCLS. High level ontologies that are shared by all systems cannot describe all possible subdomains that may be needed by local clinics. Therefore, they must extend common knowledge with domain ontologies, as well as define internal policies, rules and vocabularies that are specific to their context. Even globally shared knowledge may be used differently according to the context.

Interestingly, the very same problems have been identified on the Semantic Web, where the notion of context and heterogeneity have not been sufficiently tackled in existing standards (viz., RDF and OWL) [3]. Our goal in this paper is to propose a survey of formal approaches (logical formalisms as well as other theoretical framework) and assess their adequacy towards managing context and heterogeneity over distributed HCLS systems. We show the implications of each approach on a concrete example that realistically represents a plausible scenario in this field. While other related formal comparisons exist in the literature [4, 5, 6], we are not aware of an existing survey that presents and analyse as many formal approaches and apparently no other such survey concretely illustrates the consequences of using such or such technique.

Our contribution, in addition to the analyzing of the state of the art, is an original classification of the approaches according to requirements identified from the study of concrete HCLS systems and goal of HCLS task force.[1] Formalisms are classified according to the following requirements:

- ability to identify and use context ("context-awareness");
- ability to relate heterogeneous knowledge, either within or between contexts (in particular via more or less expressive *ontology alignments*);
- robustness wrt heterogeneity (i.e., ability to tolerate incompatibilities or inconsistencies within or between contexts);
- ability to modularize ontologies (reusing multiple ontologies or parts of ontologies);
- ability to model internal policies or profiles distinctively.

We will show that there is a continuum between strongly and loosely connected knowledge. Strong connection (e.g., ontology axioms) easily leads to many inconsistencies but enhances cross-context inferences, while loose connection avoids inconsistencies but decreases knowledge transfer between contexts. We finally show how to combine several approaches to satisfy more requirements.

The features described above are based on our experience in projects, (1) Plug and Play Electronic Patient Records (PPEPR[2]) and (2) A Roadmap for Interoperability of eHealth Systems (RIDE[3]). PPEPR [7] is an ontology-based

[1] `http://esw.w3.org/topic/HCLS/ClinicalObservationsInteroperability`
[2] `http://www.ppepr.org/`
[3] `http://www.srdc.metu.edu.tr/webpage/projects/ride/`

integration platform which connects disparate healthcare systems and RIDE is a European roadmap project with special emphasis on semantic interoperability of eHealth systems.

The paper is organized as follows. Sect. 2 presents background knowledge about existing HCLS standards as well as a discussion on how Semantic Web technologies have been used so far to better ensure interoperability of globally distributed systems. Sect. 3 describes a concrete scenario where data have to be exchanged between two hospitals. We use this to highlight the reasons for the requirements we identify, as well as a motivation for the classification we define. Sect. 4 presents the formal approaches by showing application and consequences on the concrete scenario. We put emphasis on the advantage and disadvantage of each approach. In Sect. 5, we summarize the analysis in a table, putting in evidence what features are offered by each approach. We make a higher level comparison of the approaches and discuss open issues and sketch possible paths toward better a solution by combining several approaches. Sect. 6 concludes the article.

2 Health Care and Life Sciences (HCLS)

Healthcare is a complex domain and any data integration system which connects healthcare institutes must facilitate heterogeneous systems at two levels (1) information model specific data, and (2) domain and/or institute specific terminologies / vocabularies. These two levels must interoperate to aggregate and exchange medical records from disparate healthcare systems. In this section we describe these two levels and explain how regional clinical practices influence the modelling of clinical data.

Healthcare Information Model. Information model allows modeling of domain and/or institute specific message requirements. Health Level Seven (HL7[4]) standard (version 3) develops information model specific data standards for storing and exchanging information in the healthcare industry. HL7 is the most widely used healthcare standard and shares many semantic equivalences with other influential standards such as openEHR[5] and CEN13606[6].

The HL7 (version 3) information modeling process recognizes three interrelated types of information models: Reference Information Model (RIM), Domain Message Information Model (DMIM), and Refined Message Information Model (RMIM). The RIM is a unified model which covers all domains in healthcare and defines data structures, data types and vocabularies, as well as the relationships between them. DMIM is a refined subset of the RIM and is used to create messages for a particular domain (e.g., Lab, Hospital Admin). RMIM is a subset of a DMIM and is used to express the information content for a message or set of messages (e.g., Lab-Test Order Message). All three interrelated models use the

[4] `http://www.hl7.org/`
[5] `http://www.openehr.org/`
[6] `http://www.cen.eu/`

same notation and have the same basic structure but differ from each other in their information content, scope, and intended use.

In the example scenario presented in section 3, Galway University Hospital (GUH) uses the ontological representation of RIM and creates local ontology using DMIM/RMIM models.

Terminology and Vocabulary. Healthcare and life sciences (HCLS) terminologies and vocabularies (e.g., SNOMED [8], LOINC[7]) describe medical concepts. When these concepts are placed in a clinical record they become associated with an observation (e.g., lab test), time (e.g., effective time), policy/profile (e.g., drug policy, patient profile), and relationships with other records. These associations influence the interpretation of the concepts [9]. For example, (1) a clinical *observation* concept (e.g., blood sugar test) has an effective time during which it is valid, and (2) a diabetic concept placed in a "family history" segment of a patient record does not imply that the individual patient is a diabetic patient.

Standard compliant patient records are constructed from the combination of messages derived from information model and several terminologies/vocabularies refereing to message attributes. On the other side, many healthcare institutes does not use the standard specific information model, rather their messages are constructed using general models. The presence of different healthcare standards, large scale applicability, and limitations of syntactic integration solutions, motivated the healthcare specialists to apply Semantic Web technologies to resolve heterogeneity in formal and consistent ways.

Semantic Web For HCLS Data Integration. HCLS information models, terminologies, and vocabularies can be expressed as a set of RDF(S) / OWL propositions. RDF is best for expressing medical or patient data and OWL allows more expressive propositions to be expressed, like those that represent general knowledge rather than specific patient data elements. The relationships between heterogeneous healthcare data and knowledge can be formally expressed in OWL constructs. The reasoner underlying expressive Web language like OWL can be used to entail consistent sets of inferred knowledge about the healthcare Web resources.

W3C HCLS[8] Interest Group and various research projects have taken initiatives for ontological representation of healthcare information models and to integrate them with HCLS terminologies / vocabularies [2, 10]. This integration is crucial to effectively exchange patient records between disparate healthcare systems.

3 Example Scenario: Lab-Test Order

We present the case of Sean, a patient who experienced two medical incidents in different hospitals. This example shows how heterogeneity and context issues arise in electronic patient records of different healthcare systems.

[7] http://loinc.org/

[8] http://www.w3.org/2001/sw/hcls/

```
DataProperty: guh:orderDate
  domain: rim:Act
DataProperty: guh:time
  domain: rim:Act
DataProperty: guh:first_name
  domain: rim:EntityIdentification
ObjectProperty: rim:playedRoleIn
  domain: rim:Person
  range: rim:Role
ObjectProperty: guh:hasId
  domain: rim:Entity
  range: rim:EntityIdentification

DisjointClasses:
  guh:hasMedication some rxnorm:Avandia
  guh:hasMedication some rxnorm:Insulin

Class: rim:RolePatient
  SubClassOf: rim:Role
Class: guh:IrishPPSId
  SubClassOf: rim:EntityIdentification
Class: guh:LabTestOrder
  SubClassOf: rim:Act
Class: guh:HemoglobinTest
  SubClassOf: rim:Act

Individual: loinc:_4548−4
  Anntotation: rdfs:label
    "Hemoglobin A1c/Hemoglobin.total"
  TypeOf: loinc:Code
```

```
DataProperty: dc:date
  range: rdfs:Literal
DataProperty: foaf:firstName
  domain: foaf:Person
  range: rdfs:Literal
DataProperty: goet:identification
  domain: foaf:Agent
  range: xsd:string
ObjectProperty: goet:hasAuthor
  domain: galen:Observation
  range: foaf:Agent
ObjectProperty: goet:hasTreatment

Class: rxnorm:Avandia
  SubClassOf: galen:Drug
Class: rxnorm:Insulin
  SubClassOf: galen:Drug
Class: galen:Patient
  SubClassOf: galen:Human
Class: goet:OrderLabObservation
  SubClassOf: galen:OrderAct
Class: snomed:_43396009
  Annotation: rdfs:label
    "Hemoglobin A1c measurement"
```

Fig. 1. Extract of GUH (left) and GOET (Right) ontologies. Correspondences are informally represented by dotted lines.

Background. In Galway, Ireland community clinic, Dr. Paul Smith is a primary care physician using SmithSys as his Practice Management System. Galway University Hospital (GUH) has a pathological laboratory, directed by Dr. John Colleen. In Göttingen, Germany, Dr. Gustav Roth is a physician at the City hospital (GOET). GOET is directed by Dr. David Hahn. The information systems GUH and GOET can receive orders and transmit result reports electronically.

Let us now—somewhat anticipating—assume that these hospitals are using Semantic-Web-enabled systems. They model medical, clinical and patient knowledge using OWL, reusing well established ontologies such as SNOMED, Galen[9]. They extend them to represent local domain knowledge, internal policies, etc. The data, e.g., the patient records, are represented in RDF. This presents a realistic situation since GUH has already been involved in a project where Semantic Web technologies were used [7].

Fig. 1 shows a snippet of the ontologies used in GUH (left) and GOET (right). GUH is using ontologies that are strongly influenced by HCLS current standards (e.g., RIMOWL[10]) while GOET is using more Semantic Web oriented ontologies

[9] http://www.co-ode.org/galen/

[10] The OWL version of RIM. http://esw.w3.org/topic/HCLS/ClinicalObservationsInteroperability/HL7CDA2OWL.html

```
:sean a rim:Person ;
  rim:playedRoleIn [a rim:RolePatient] ;
  guh:hasID :678970W .
:678970W a guh:IrishPPSId ;
  rdfs:label "Sean Murphy's Irish PPS number" ;
  guh:first_name "Sean" ;
  guh:last_name "Murphy" ;
  guh:hasMedication [a rxnorm:Insulin] .
:paul a rim:Person ;
  rim:playedRoleIn [a rim:RolePhysician] ;
  guh:hasID :68374532 .
:68374532 a guh:PID ;
  rdfs:label "Paul Smith's professional Id" ;
:6560-89 a guh:LabTestOrder ;
  guh:orderedBy :paul ;
  guh:hasPatient :sean ;
  guh:orderDate "2007-07-16" ;
  guh:time "13:07:07" ;
  guh:orders :Test743890 .
:Test743890 a guh:HemoglobinTest ;
  rim:effectiveTime "M0717173034" ;
  guh:specimen :s7854698 ;
  guh:hasCode loinc:_4548-4 .
```

```
:345678IE a galen:Patient ;
  goet:identification "Irish Driving licence" ;
  foaf:family_name "Murphy" ;
  foaf:firstName "Sean" ;
  goet:hasTreatment [a rxnorm:Avandia] .
:6837-4532 a galen:Doctor ;
  goet:identification "Professional Id" ;
  foaf:family_name "Roth" ;
  foaf:firstName "Gustav" .
:7779-78 a goet:OrderLabObservation ;
  goet:hasAuthor :6837-4532 ;
  goet:hasPatient :345678IE ;
  galen:orders :Test777767 .
:Test777767 a galen:BloodSugarTest ;
  dc:date "2008-12-22T00:30:00" ;
  goet:hasMeasurement [a snomed:_408254005] ;
  goet:hasSpecimen :s89756 .
:s89756 a galen:Specimen .
```

Fig. 2. Extract of Lab-Test orders at GUH (left) and GOET (right)

(e.g., FOAF) which better ensures interoperability with other Linked Data (e.g., other administrative systems may take advantage of FOAF data).

Patient Case History 1. Dr. Smith has a patient Sean Murphy (identified by Irish PPS number: 678970W), a type 2 diabetic on Insulin therapy. Dr. Smith fills out the electronic order form in his office system for a Glycosylated hemoglobin (HbA1c) test and sends it to GUH. GUH receives the electronic order and performs the requested HbA1c test. Dr. Colleen authorizes the results of the HbA1c test and sends it to SmithSys. GUH drug policy stipulates that type 2 diabetics can be treated with either Avandia or Insulin, but not both.

Patient Case History 2. On Christmas holidays Sean visits Göttingen but has a major car accident and doctors need to operate quickly. Sean is identified using his international driving licence number 345678IE. GUH Lab and GOET, can share patient information but, since they are using different domain ontologies, manual effort is required for information integration. Due to time constraints GOET performs all tests locally. Sean provides an informal description of his medical history and Dr. Roth orders a HbA1c test to examine his current blood sugar level before performing surgery. GOET drug policy on type 2 diabetics does not suggest any restrictions for Avandia and Insulin treatments, and thus Dr. Roth decides to prescribe Avandia in complement of Sean's insulin treatment.

Fig. 2 show the snippets of Lab-Test orders represented in RDF Turtle[11]. Instances on both sides use global ontologies (RIM, LOINC, Galen, RxNorm, foaf, SNOMED) and local (guh, goet) ontologies.

[11] `http://www.w3.org/TeamSubmission/turtle/`

Issues related to heterogeneity and context. While GUH and GOET are modelling the same domain of knowledge in a common format (RDF and OWL), their systems are not directly interoperable due to variations in the way they model it. These systems are defining medical knowledge according to two different contexts in a heterogeneous way. There are two levels of heterogeneity:

- *intra-contextual heterogeneity*, which occurs within a context. As seen in the example, several vocabularies are used for describing different types of information (e.g., RxNorm for drugs, Loinc for types of tests). Multiple terminologies must be integrated in a modular way. Local systems also have to deal with knowledge of different nature, e.g., axioms about a domain of discourse (e.g., an hemoglobine test is a kind of act) and internal policy (a patient cannot be treated with both Insulin and Avandia).
- *inter-contextual heterogeneity* occurs between contexts. If the terminologies are different, the systems cannot interoperate unless some relations are asserted between the domain ontologies. Correspondences between local ontologies are informally presented in Fig. 1 using dotted lines. Besides, corresponding concepts of distinct ontologies can be modeled in very different ways. E.g., GUH here uses an object property for patient's identification, while GOET is using a datatype. Thus, systems should be able to tolerate such heterogeneity. Finally, we can see that identifying context is crucial, notably to see which policy belongs to which context.

Our aim in this paper is not to establish a universal notion of context. A natural choice to identify a context on the Web is to use URIs. Our focus is, how to aggregate and exchange contextual information identified by URIs. The next section does a reality check that evaluates how far HCLS systems could benefit from Semantic technologies to achieve seamless integration on the Web. Based on the limitations identified in a purely standard-based approach we survey other formal approaches that are relevant within the scope of this paper.

4 Handling Context and Heterogeneity

In the past researchers have attempted to handle context and heterogeneity when tried to aggregate and exchange structured knowledge from disparate information systems [11]. HCLS standard begin to incorporate contextual information within their frameworks [8], therefore data integration systems must use that contextual information. In this section we investigate the support of five features—Context Awareness, Modularity, Profile/Policy Management, Correspondence Expressiveness, Robustness to Heterogeneity—in standard and other relevant knowledge base approaches.

4.1 Standard Approach: OWL

Although we have already raised some doubts in the informal discussion above on the suitability of current Semantic Web technologies wrt heterogeneity and context, we can explore what OWL can offer to overcome these problems. This language partially addresses modularity and expressing correspondences between various ontologies.

OWL Solution. In the example scenario, several well identified terminologies are reused. OWL provides an import feature thanks to the property `owl:imports` which helps to modularly integrate several domain ontologies. By stating that GUH ontology imports RIM and LOINC, the axioms of these ontologies are automatically made part of the GUH ontology.

In terms of correspondence expressivity, OWL supports mappings constructs by stating axioms involving terms of different ontologies, thanks to the following OWL keywords: *subClass*, *sameAs*, *equivalentClass*, *equivalentProperty*, *differentFrom*, *disjointWith*, and *propertyDisjointWith*.

In the example scenario, concepts and properties of the two hospitals are modeled differently, but correspondences can be identified and expressed in OWL. In List. 1, we present the axioms that should be added to make the two systems interoperate[12]. Notice that these mappings relating terms in two different contexts cannot be distinguished from mappings between terms of imported ontologies within one context.

```
Class: ( foaf:Person ) SubClassOf: ( rim:Entity )
Class: ( galen:OrderAct ) SubClassOf: ( rim:Act )
Class: ( rim:playedRoleIn some rim:RolePatient ) EquivalentTo: ( galen:Patient )
Class: ( guh:LabTestOrder ) EquivalentTo: ( goet:OrderLabObservation )
Class: ( guh:HemoglobinTest and (rim:measures some loinc:_4548−4) )
    EquivalentTo: ( galen:BloodSugarTest and (goet:hasCode some {snomed:_43396009}) )

EquivalentProperties: ( guh:first_name ) ( foaf:firstName )
EquivalentProperties: ( guh:hasMedication ) ( goet:hasTreatment )

Instance: ( guh:sean ) sameAs: ( goet:345678IE )
```

Listing 1. Extract of OWL supported mapping definitions

This approach is the only possible way of dealing with heterogeneity which fully complies with established Semantic Web standards. It has been argued that it improves interoperability of HCLS systems [2], when compared to previous standards in this field, such as HL7. However, these standards are clearly not enough to solve the important issues presented in Sect. 3.

Limitations. First, while a form of modularity is offered by OWL, its import statement can only handle the reuse of full ontologies without being able to specify subparts of them. This is particularly problematic with large ontologies like SNOMED, which have to be fully integrated, even when only a small subdomain is needed.

Second, not all relevant mappings can be expressed in OWL. For example, (1) the *ObjectProperty* `guh:hasId` and the *DatatypeProperty* `goet:identification` are designed for similar purpose (identify the person) but OWL semantics does not allow to map between *ObjectProperty* and *DatatypeProperty*; (2) OWL does

[12] As in Fig. 1, we are using the OWL Manchester syntax [12].

not support operations on attributes, e.g., the concatenation of two *Datatype-Properties* (e.g, `guh:orderDate`, `guh:time`) into a single *DatatypeProperty* (e.g., `dc:date`). Other examples include unit or currency conversion.

Third, OWL does not include any feature for distinguishing between universal facts (e.g., a patient is a person) and local policy or profile (e.g., people should not be treated with both Insulin and Avandia). Additionally, OWL does not permit identifying the context of an axiom or term. The implication of these two limitations is that policies have to be represented as DL axioms, and these axioms are affecting all contexts identically. In the scenario, according to GUH, Sean is treated with Insulin. When he goes to GOET, the record indicates that he has been given Avandia. Thanks to the aforementioned mappings, the terms hasMedication and hasTreatment can be interpreted interchangeably, so that GOET can understand GUH record automatically. But it leads to a contradiction with the GUH policy because Sean has now both treatments. Yet, it should not be the case because GOET does not have this policy, and therefore should not detect an inconsistency. Note that undesired interactions can be reduced by using subsumption instead of equivalence in mappings, but the problem remains.

Fourth, OWL is not tolerant to diverging modeling of a knowledge domain. Different points of view can equally well describe a domain of interest, while being partially incompatible. Interpreting all axioms and assertions as equaly true, in all contexts, may easily lead to inconsistency or nonsensical entailments.

4.2 Distributed Description Logic

Distributed Description Logics (DDL) [13] is a formalism which was developped to formalize contextual reasoning with description logics ontologies. Indices $i \in I$ are used to determine from which context an ontology or an axiom comes from. Given, for instance, an axiom $C \sqsubseteq D$ from an ontology O_i, DDL uses the prefixed notation $i : C \sqsubseteq D$ to highlight the context of the axiom. Moreover, cross-context formulas can be defined to relate different terminologies. These particular formulas are called bridge rules and written either $i:C \xrightarrow{\sqsubseteq} j:D$ or $i:C \xrightarrow{\sqsupseteq} j:D$ where i and j are two different contexts, and C and D are terms from the contextual ontologies O_i and O_j respectively. A bridge rule $i:C \xrightarrow{\sqsubseteq} j:D$ (resp. $i:C \xrightarrow{\sqsupseteq} j:D$) should be understood as follows: from the point of view of O_j (i.e., in the context j), C is a subclass (resp. superclass) of D.

In terms of model-theoretic semantics, this is formalized by assigning a distinct description logics interpretation $\mathcal{I}_i$ to each contextual ontology O_i, instead of having one single global interpretation. Thus, there is as many domains of interpretation as there are contexts. Additionally, cross-context relations are explicited by so-called domain relations, that is set-theoretic binary relations between each pairs of contexts (formally, $r_{ij} \subseteq \Delta^{\mathcal{I}_i} \times \Delta^{\mathcal{I}_j}$). Two contextual interpretations $\mathcal{I}_i$ and $\mathcal{I}_j$ satisfy a bridge rule $i:C \xrightarrow{\sqsubseteq} j:D$ (resp. $i:C \xrightarrow{\sqsupseteq} j:D$) iff $r_{ij}(C^{\mathcal{I}_i}) \subseteq D^{\mathcal{I}_j}$[13] (resp. $r_{ij}(C^{\mathcal{I}_i}) \supseteq D^{\mathcal{I}_j}$).

[13] For a set S, $r_{ij}(S) = \{x \in \Delta^{\mathcal{I}_j} \mid \exists y \in S, \langle x, y\rangle \in r_{ij}\}$.

The advantage of this approach is the identification of context, a better robustness wrt heterogeneity, improved modularity. However, it still misses some of the requirements we identified.

Solution in DDL. In the scenario, ontologies would be related thanks to C-OWL [14] bridge rules, which instantiates DDL for the description logic of OWL. A P2P reasoning system called Drago [15] implements a fragment of C-OWL and could be used in each hospital. Each peer manages its own context by reasoning with its internal ontology and "incoming" bridge rules. Messages are sent to neighbour peers according to a distributed algorithm involving bridge rules in order to take advantage of knowledge from other contexts.

In our healthcare use case, GUH and GOET may implement a Drago reasoner. GOET expresses the correspondences by way of bridge rules, as shown with a few examples in List. 2.

guh:(*rxnorm*:*Insulin*) $\xrightarrow{\equiv}$ **goet**:(*rxnorm*:*Insulin*)
guh:(*rxnorm*:*Avandia*) $\xrightarrow{\equiv}$ **goet**:(*rxnorm*:*Avandia*)
guh:(*rim*:*playedRoleIn* some *rim*:*RolePatient*) $\xrightarrow{\equiv}$ **goet**:(**galen**:**Patient**)
guh:(*guh*:*hasMedication*) $\xrightarrow{\equiv}$ **goet**:(**goet**:**hasTreatment**)
guh:(*guh*:sean) $\xmapsto{\equiv}$ **goet**:(**goet**:345678IE)

Listing 2. Extract of DDL bridge rules

Because of the semantics of bridge rules, no inconsistency can be derived in this case. So DDL reduces the problem of diverging policies. In fact, DDL dicreases interactions between different ontologies, which in turn decrease the chance of inconsistency.

Limitations. Bridge rules are not able to represent mappings between object and datatype properties, nor can they express operations on datatypes. Besides, C-OWL uses the same import mechanism as OWL. Additionally, the non-standard semantics of DDL may be counter intuitive, sometimes. Neither disjointness nor cardinality constraints are "transferred" from an ontology to the other via bridge rules. That is, if Insulin and Avandia are disjoint in GUH, and there are the bridge rules above, it cannot be inferred that Insulin and Avandia are disjoint in GOET. However, a variant of DDL has been defined to treat this specific problem [16]. Finally, the problem of policy is not completely solved. By adding the bridge rules:

guh:(*guh*:*hasMedication* some *rxnorm*:*Insulin*) $\xrightarrow{\equiv}$ **goet**:(**goet**:**hasTreatment** some *rxnorm*:*Insulin*)
guh:(not *guh*:*hasMedication* some *rxnorm*:*Avandia*) $\xrightarrow{\equiv}$ **goet**:(not **goet**:**hasTreatment** some *rxnorm*: *Avandia*)

Listing 3. Other possible bridge rules

GOET system can infer that a patient must not be treated with both Avandia and Insulin, which is what we tried to avoid.

4.3 Other Contextual Reasoning Formalisms

Contextual reasoning formalisms are characterized by a non-standard semantics where several ontologies are assigned distinct interpretations. Apart from DDL, this family of formalisms includes $\mathcal{E}$-connections, Package-based Description Logics and Integrated Distributed Description Logics.

$\mathcal{E}$-connections. $\mathcal{E}$-connections is another formalism for reasoning with heterogeneous ontologies [17]. Instead of expressing correspondences between ontologies, ontologies are connected by using special terms (called *links*) which can be used in an ontology assertion in conjunction with terms from another ontology. The semantics of links is very similar to the semantics of roles in description logics, except that instead of relating things from the same domain of interpretation, they relate two different domains. In principle, $\mathcal{E}$-connections serve to relate ontologies about very different domains of interest. For example, an ontology of laboratories in GUH could be connected to an ontology of medical staff used in GOET. To do this, one can define the link ⟨hasDirector⟩ and use it in GUH ontology as follows:

$\mathit{guh}\!:\!\mathsf{Laboratory} \sqsubseteq \exists\langle\ \mathsf{hasDirector}\ \rangle \mathbf{goet}\!:\!\mathsf{StaffMember}$

Thus, $\mathcal{E}$-connections are particularly useful for ontology design by modularly reuse and connect existing blocks. However, our paper focus on relating existing ontology systems on overlapping domains. So, although $\mathcal{E}$-connections is a relevant formalism for the management of heterogeneity, its applicability to the type of scenario we are interested in is weak.

Package-based Description Logics. Package-based Description Logics (P-DL [18]) is a formalism that was essentially designed for the modularity of Web ontologies. It essentially improves the import mechanism of OWL by allowing finer granularity of imports, namely by naming imported terms instead of entire imported ontologies. In P-DL, while each ontology is interpreted over a distinct domain of interpretation, the imported terms must be interpreted identically in both the importing and the imported ontologies. Therefore, this formalism does not tolerate variations in the modeling of terms shared among different ontologies. Non-equivalence correspondences must be represented as axioms in a module that imports all the related terms, which leads to the same limitations as in OWL.

Integrated Distributed Description Logics. Integrated Distributed Description Logics (IDDL [19]) use a different paradigm than the other contextual frameworks. While DDL, $\mathcal{E}$-connections and P-DL assert relations between ontologies from the point of view of one ontology, IDDL asserts correspondences from a "third party"'s point of view. This difference in the semantics implies that correspondences can be manipulated and reasoned with independently of the ontologies, allowing operations like inversing or composing correspondences.

A reasoning procedure for this formalism has been defined [20], where a central system detaining the correspondences can determine global consistency by communicating with local reasoners of arbitrary complexity. This formalism is useful for federated reasoning systems, while the interactions between local ontologies are rather weak. By separating local reasoning and global reasoning it better prevents interactions between contexts, thus being quite robust to heterogeneity. A policy of the form $C \sqsubseteq \neg D$ would only influence another ontology if a disjointness is asserted at the alignment level, e.g., $\mathrm{guh}:C \overset{\perp}{\longleftrightarrow} \mathrm{goet}:D'$.

The drawbacks are the possible missing inferences at the local level, and this approach does not take policy into account either. Correspondences are not more expressive than in DDL.

4.4 Handling and Reasoning with Inconsistencies

Robustness to heterogeneity is an important aspect in healthcare integration scenarios. One of the most problematic consequence of heterogeneity is the occurrence of undesired inconsistencies. Therefore, we deem useful investigating formal approaches for handling inconsistencies. There are two main ways to deal with inconsistent ontologies. One is to simply accept the inconsistency and to apply a non-standard reasoning method to obtain meaningful answers in the presence of inconsistencies. An alternative approach is to resolve the error, i.e., ontology repair, whenever an inconsistency is encountered.

Repairing or revising inconsistent ontology is, in principle, a possible solution for handling inconsistency. However, one major pragmatic issue we observe that healthcare institutes may not expose and/or allow repair of their knowledge bases due to various legal constraints. Also, in a typical Semantic Web setting, importing ontologies from other sources makes it impossible to repair them, and if the scale of the combined ontologies is too large like HCLS ontologies then repair might appear ineffective. Other work focus on revising mappings only [21], but they are meant to be used at alignment discovery time, which we are not discussing in this paper.

Reasoning with inconsistencies is also possible without revision of the ontology. One effective way of tolerating inconsistencies consist of using paraconsistent logics [22]. Paraconsistent logics use "weaker" inference system that entails less formulas than in classical logics. This way, reasoning can be done in the presence of inconsistency. A paraconsistent extension of OWL was proposed in [23]. Alternatively, defeasible argumentation [24] and its implementation Defeasible Logic Programs (DeLP [25]) have been introduced to reason and resolve inconsistencies. In this case, the TBox is separated into 2 subsets, one being *strict*, which means that it must always be used in reasoning, the other being *defeatable*, which means that an argumentation process may defeat them and nullify them for a particular reasoning task.

While we want to tolerate inconsistency when reasoning with ontology defined in other context, it is not desirable, in a HCLS system to tolerate local inconsistencies. The system should have a strict logical framework when it only treats local data, that are existing in a unique and well understood context.

Unfortunately, the approaches mentionned here are not able to disinguish local knowledge and external knowledge. They do not either allow specification of the types of mappings we need, and are not capable of treating policies.

4.5 Other Formal Approaches

Database-Style Integrity Constraints for OWL. This approach is motivated from data-centric problems in DL/OWL based applications. Authors [26] have established the relationship between role of Integrity Constraints (IC) in databases, i.e., (1) data reasoning (e.g., in checking the integrity of a database) and schema reasoning (e.g., in computing query subsumption), and (2) DL/OWL knowledge bases (e.g., schema (*TBox*) reasoning and data (*ABox*) reasoning). In this approach an additional *TBox* is introduced to model constraint axioms, in result knowledge contains two *TBoxes* and an *ABox*. In *TBox* reasoning, constraints behave like normal *TBox* axioms and for *ABox* reasoning they are interpreted as constraints in relational databases. This approach is very relevant in solving profile/policy issues of our example scenario. For example, to avoid inconsistency due to hospital specific drug policy, axiom:

$$\exists \mathrm{guh:hasMedication.rxnorm:Avandia} \sqsubseteq \neg \mathrm{guh:hasMedication.rxnorm:Insulin}$$

can be placed in *TBox* for constraints and when *TBox* reasoning is performed only standard axioms could be taken into account. In case of *ABox* reasoning constraints axioms can act as Integrity Constraints. To some extent, it helps formalizing policies but since it does not identify the context of these constraints, their utility for this purpose is limited. Moreover, as a standard OWL, robustness to heterogeneity is poor.

4.6 Modular Web Rule Bases

Although this approach is not based on current Semantic Web standards, it is relevant to this survey. The framework proposed in [27] makes the distinction between global knowledge, local knowledge and internal knowledge. The framework is based on a rule-based language rather than description logics and provides and approach to express and reason with modularity on top of Semantic Web. In this framework each predicate in rule-base is constrained with uses and scope, which in turn determine the reasoning process. The framework also treats different forms of negation (e.g., weak, strong) to include Open-World Assumption (OWA) and Closed-World Assumption (CWA). This rule-based framework provides model-theoretic compatible semantics and allow certain predicates to be monotonic and reasoning is possible during inconsistency of knowledge bases. This framework address few issues of our example scenario like (1) Rules can express DL-Axioms and can be exchanged with certain restrictions (e.g., private, global, local). For example drug policy rule of our example scenario:

$$\mathcal{F} \leftarrow \mathit{hasMedication}(?x,?y),\ \mathit{Avandia}(?y),\ \mathit{hasMedication}(?x,?z), \mathit{Insulin}(?z)$$

can be expressed and treated appropriately. However, one major problem we observe that how DL based ontologies (as majority of HCLS ontologies are DL ontologies) and rules can work together. The integration of DL with rules is still an open research problem [28]. Moreover, this framework is not concerned about the heterogeneity of the knowledge model, and do not provide expressive way of relating contextual ontologies.

4.7 Query-Based Data Translation

Query-based approach translates data from one knowledge source to another, and is close to the problem of expressing complex correspondences that we address in this paper. In this approach mappings between ontologies are first expressed in expressive alignment language [29] and then grounded and executed to a combined query language and engine, SPARQL++ and PSPARQL, called PSPARQL++ [30]. List. 4 show how (a) two ontology entities (guh:orderDate, guh:time) could be concatenated to a single entity (dc:date) and (b) a conversion is possible between object property and datatype property by using proposed cast-function that converts *xsd:string* to RDF resource. Expressive correspondences between ontology instances can be constructed in "SPARQL CONSTRUCT" to create additional dataset and query upon them.

```
(a) CONSTRUCT { ?X dc:date fn:concat(?Date,"T",?Time). }
    WHERE { ?X guh:orderDate ?Date . ?X guh:time ?Time . }

(b) CONSTRUCT { ?X guh:hasId rdf:Resource(fn:encode−for−uri(?Id)) . }
    WHERE { ?X goet:identification ?Id . }
```

Listing 4. Mappings Expressed in SPARQL CONSTRUCT

This approach allows one to express complex correspondences like concatenating attributes or even datatype to object properties and one can avoid some undesired interactions between knowledge of various sources. However, one major limitation is that the query result depends on how the correspondences are written and the knowledge in the domain ontologies are largely unexploited. Similarly, complex correspondences can be expressed in Rule Interchange Format (RIF), which offers a rich set of built-in functions (e.g., *string* manipulations), as well as a formal semantics[14] for interoperating with RDF and OWL knowledge bases. However, RIF is yet to be approved as W3C's standard recommendation and so far is still a work in progress, which is why we do not further focus on this approach here.

5 Summary and Solution Sketch

Table 1 summarizes the formal approaches for their ability to deal with heterogeneous knowledge bases. First row marks the identified features and first column show the formal approaches. Standard Semantic Web languages (RDF(S)/OWL)

[14] `http://www.w3.org/TR/rif-rdf-owl/`

Table 1. Formal Approaches towards Heterogeneity and Context

	C.A.[15]	M.[15]	P.& P.M.[15]	C.E.[15]	R.H.[15]
DL/OWL	No	Very limited	No	Good	Very weak
DDL/C-OWL	Yes	Yes	No	Very good	Good
P-DL	Yes	Yes	Very limited	Very limited	Weak
DDL Revisited	Yes	Yes	No	as DDL	Medium
$\mathcal{E}$-connections	Yes	Yes	No	see text	Excellent
IDDL	Yes	Yes	No	Good	Very good
DeLP/Paraconsistent	No	No	Limited	as DL	Good
Query-based	Yes/No?	No	No	Very good	Very good
Modular Rule bases	Yes	Yes	Limited	Limited	Weak
OWL/IC	No	Very limited	Good	Good	Weak

do not support the representation and reasoning of contextual knowledge. In result, local policy/profile management is out side the scope of Standard Semantic Web languages.

There are essentially 2 groups of formal approaches: one is trying to deal with the notions of context, heterogeneity and correspondences between ontologies: those are the so-called modular ontology languages DDL, DDL revisited, $\mathcal{E}$-connection, P-DL, IDDL and modular rule bases. These formalisms can be ordered in their ability to tolerate heterogeneity: P-DL $<$ modular rule bases $<$ DDL revisited $<$ DDL $<$ IDDL $<$ $\mathcal{E}$-connection Note that this ordering is based on our personal analysis rather than a formal, proved property. The more tolerant to heterogeneity, the less knowledge is transfered from one context to the others.

On the other hand, there are formalisms that very well handle heterogeneity by treating inconsistency specifically. Those are DeLP and paraconsistent logics. These approaches are not addressing the problem of contextual knowledge, nor the one of policy management. Modular rule bases and OWL/IC both offer a way to treat local policies, one by altering the mode of reasoning of a subset of the vocabulary, the other by separating the TBox in two for TBox reasoning and data integrity checking. Interestingly, OWL/IC is built on top, and compatible with, OWL TBox reasoning. Finally, query transformation offers an algorithmic way of translating instances between two terminologies without relying on complex reasoning process.

Tab. 1 shows that the advantages of certain approaches are exactly the drawbacks of other. Considering these remarks, we will sketch paths to a solution that better encompass the requirements by combining several approaches.

5.1 Towards a Framework for Handling Context and Heterogeneity

Not all approaches are incompatible. We believe that an improved solution to the problem of heterogeneity and context relies on the combination of several

[15] C.A.: Context-awareness; M.: Modularity; P.& P.M.: Profile and policy management; C.E.: Correspondence expressiveness; R.H.: Robustness to heterogeneity.

approaches. While a semantics must be chosen to allow context aware reasoning, they can be extended with other non-monotonic approaches. There are many possible directions, too many to compare them all, so we only show possible paths that we think are best suited for the scenarios we consider.

Adding constraints to context-aware formalisms. We present here a possible approach for both reasoning with heterogeneous context while still taking into account internal policies. This approach is inspired by [26], where they define a separate T-Box for integrity constraints. This way, we define a local T-Box as a pair $\langle D, P\rangle$, where D describes a domain ontology, while P represents the internal policy. This can be easily used on top of context-aware formalisms like DDL, P-DL or IDDL. However, the reasoning process have to be slightly modified. For each context, a distinct policy-extended T-Box is assigned, and possibly correspondences, or bridge rules, or imports are added. To simplify the discussion, let us consider that the DDL semantics is used. We define a policy-enabled distributed entailment $\models_P$ over distributed ontologies Ω as follows. For a given local OWL axiom $\alpha = i : C \sqsubseteq D$ in the terminology of ontology $\langle D_i, P_i\rangle$, $\Omega \models_P \alpha$ iff α is entailed by the distributed ontology composed of $D_i \cup P_i$ as well as the ontologies $\{D_j\}_{j\neq i}$ and the bridge rules.

In other words, only the policy axioms of the ontology which is asking for an entailment is used. In our scenario, it means that if GUH is reasoning, it will take its drug policy into account but not the one of GOET, while GOET would not consider the Avandia-Insulin counter indication of GUH. The very same approach can be easily adapted to P-DL or IDDL.

Modular Web rules with context-awareness. While we emphasize more on the previous approach because its applicability is straightforward, we also envisage less obvious and exploratory solutions, namely combining rules with a distributed context-aware approach. We are interested in this approach because in our experience, profile and policy are better represented as rules.

Defeasable context-aware logics. In this approach, we propose to consider that all knowledge is considered strict (in the DeLP sense of the word), while knowledge coming from an external context and modeling a policy would be defeasable. This way, if policies of two contexts are compatible, they will both be considered, but if an inconsistency is raised, only the "foreign" policy will be defeated by the argumentation process.

6 Conclusion and Future Works

While we studied approaches from a formal point of view, concentrating on what relates to heterogeneity and context, we are aware that other characteristics may influence the choice of a formalism. In particular, (1) *performance*: non standard formalisms have variable algorithmic complexity. If they are too complex, chances are that a critical system such as a hospital knowledge base may not want to use them. Certain results have to be provided very fast to make life-saving decisions; (2) *temporal information*: temporal information is crucial in several cases in patient records, e.g., validity of a blood test etc. While this information can be put in plain RDF in the data, a specific formalisms for temporal reasoning could be added to the local representation.

Our aim is to propose a solution based on standard semantics, thus keeping minimal gap between HCLS ontology already deployed in various systems and proposed solution. Our planned future work concerned the theoretical study of (1) policy-enabled context-aware logics and (2) query mechanism that can exploit the context-aware knowledge bases.

References

1. Szolovits, P.: Artificial Intelligence in Medicine. Westview Press, Inc. (1982)
2. Rector, A.L., Qamar, R., Marley, T.: Binding Ontologies & Coding Systems to Electronic Health Records and Messages. In: Proc. of 2nd International Workshop on Formal Biomedical Knowledge Representation, KR-MED 2006, Sun SITE Central Europe (CEUR), vol. 222 (2006)
3. Guha, R.V., McCool, R., Fikes, R.: Contexts for the Semantic Web. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) ISWC 2004. LNCS, vol. 3298, pp. 32–46. Springer, Heidelberg (2004)
4. Akman, V., Surav, M.: Steps Toward Formalizing Context. AI Magazine 17(3), 55–72 (1996)
5. Serafini, L., Stuckenschmidt, H., Wache, H.: A Formal Investigation of Mapping Languages for Terminological Knowledge. In: [31], pp. 576–581 (2005)
6. Cuenca-Grau, B., Kutz, O.: Modular Ontology Languages Revisisted. In: Proc. of IJCAI 2007 Workshop on Semantic Web for Collaborative Knowledge Acquisition, SWeCKa 2007 (2007)
7. Sahay, R., Fox, R., Hauswirth, M.: Semantic Service-oriented Design and Development Methodology for Enterprise Healthcare Integration. In: Proc. of 5th International Conference on Web Information Systems and Technologies, WEBIST 2009, pp. 159–166. INSTICC Press (2009)
8. Spackman, K.: SNOMED CT Style Guide: Situations with Explicit Context. Technical report, SNOMED CT (2008)
9. Rector, A.L., Brandt, S.: Why do it the hard way? The Case for an Expressive Description Logic for SNOMED. Journal of the American Medical Informatics Association 15, 744 (2008)
10. Bicer, V., Laleci, G.B., Dogac, A., Kabak, Y.: Artemis Message Exchange Framework: Semantic Interoperability of Exchanged Messages in the Healthcare Domain. SIGMOD Record 34, 71–76 (2005)
11. Guha, R.V.: Contexts: a Formalization and Some Applications. PhD thesis, Stanford University (1991)
12. Horridge, M., Patel-Schneider, P.F.: OWL 2 Web Ontology Language, Semantics and Abstract Syntax. Technical report, W3C (2009)
13. Borgida, A., Serafini, L.: Distributed Description Logics: Assimilating Information from Peer Sources. Journal on Data Semantics 1, 153–184 (2003)
14. Bouquet, P., Giunchiglia, F., van Harmelen, F., Serafini, L., Stuckenschmidt, H.: C-OWL: Contextualizing Ontologies. In: Fensel, D., Sycara, K., Mylopoulos, J. (eds.) ISWC 2003. LNCS, vol. 2870, pp. 164–179. Springer, Heidelberg (2003)

15. Serafini, L., Tamilin, A.: DRAGO: Distributed Reasoning Architecture for the Semantic Web. In: Gómez-Pérez, A., Euzenat, J. (eds.) ESWC 2005. LNCS, vol. 3532, pp. 361–376. Springer, Heidelberg (2005)
16. Homola, M.: Distributed Description Logics Revisited. In: [31]
17. Kutz, O., Lutz, C., Wolter, F., Zakharyaschev, M.: $\mathcal{E}$-connections of abstract description systems. Artificial Intelligence 156(1), 1–73 (2004)
18. Bao, J., Caragea, D., Honavar, V.G.: On the Semantics of Linking and Importing in Modular Ontologies. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 72–86. Springer, Heidelberg (2006)
19. Zimmermann, A.: Integrated Distributed Description Logics. In: [32], pp. 507–514.
20. Zimmermann, A., Le Duc, C.: Reasoning on a Network of Aligned Ontologies. In: Calvanese, D., Lausen, G. (eds.) RR 2008. LNCS, vol. 5341, pp. 43–57. Springer, Heidelberg (2008)
21. Meilicke, C., Stuckenschmidt, H., Tamilin, A.: Supporting Manual Mapping Revision using Logical Reasoning. In: Proc. of 23rd AAAI Conference on Artificial Intelligence, AAAI 2008, pp. 1213–1218. AAAI Press, Menlo Park (2008)
22. Béziau, J.Y., Carnielli, W., Gabbay, D.M.: Handbook of Paraconsistency. College Publications (2007)
23. Huang, Z., van Harmelen, F., ten Teije, A.: Reasoning with Inconsistent Ontologies. In: [31], pp. 454–459
24. Chesñevar, C.I., Maguitman, A.G., Loui, R.P.: Logical Models of Argument. ACM Computing Survey 32(4), 337–383 (2000)
25. García, A.J., Simari, G.R.: Defeasible Logic Programming: An Argumentative Approach. Theory and Practice of Logic Programming 4(1–2), 95–138 (2004)
26. Motik, B., Horrocks, I., Sattler, U.: Bridging the gap between OWL and relational databases. In: Proc. of 16th International Conference on World Wide Web, WWW 2006, pp. 807–816. ACM Press, New York (2007)
27. Analyti, A., Antoniou, G., Damásio, C.V.: A Principled Framework for Modular Web Rule Bases and Its Semantics. In: Proc. of 11th International Conference on Principles of Knowledge Representation and Reasoning, KR 2008, pp. 390–400. AAAI Press, Menlo Park (2008)
28. Eiter, T., Ianni, G., Krennwallner, T., Polleres, A.: Rules and Ontologies for the Semantic Web. In: Reasoning Web, 4th International Summer School 2008, RW 2009, pp. 1–53. Springer, Heidelberg (2008)
29. Euzenat, J., Scharffe, F., Zimmermann, A.: Expressive alignment language and implementation. Deliverable D2.2.10, Knowledge Web NoE (2007)
30. Euzenat, J., Polleres, A., Scharffe, F.: SPARQL Extensions for Processing Alignments. IEEE Intelligent Systems 23, 82–84 (2008)
31. Kaelbling, L.P., Saffiotti, A. (eds.): Proc. of 19th International Joint Conference on Artificial Intelligence, IJCAI 2005. Professional Book Center (2005)
32. Calvanese, D., Franconi, E., Haarslev, V., Lembo, D., Motik, B., Tessaris, S., Turhan, A.Y. (eds.): Proc. of 20th International Workshop on Description Logics, DL 2007. Bolzano University Press (2007)