$\mathcal{ORC}$: an $\mathcal{O}$ntology $\mathcal{R}$easoning $\mathcal{C}$omponent for Diabetes

Özgür Kafalı[1], Michal Sindlar[2], Tom van der Weide[2] and Kostas Stathis[1]

[1] Department of Computer Science
Royal Holloway, University of London, Egham, TW20 0EX, UK
{ozgur.kafali,kostas.stathis}@rhul.ac.uk

[2] Portavita B.V., 1000 BG Amsterdam, Netherlands
{m.sindlar,t.van.der.weide}@portavita.eu

Abstract

$\mathcal{ORC}$ is an ontology reasoning component that builds upon existing ontology modelling tools and techniques to support the integration and interpretation of multimodal medical information. We show how to embed $\mathcal{ORC}$ as a reasoning capability in reactive infrastructure agents that support intelligent agents operating in COMMODITY$_{12}$, a personal health environment for diabetic patients and the medical professionals that treat them. The benefits of the approach are illustrated by showing how medical information for patient profiles at different sources can be included in COMMODITY$_{12}$, thus extending the generality and potential of the resulting system. The approach also illustrates how ontologies can be combined with a variety of artificial intelligence tools and techniques to support e-Health activities on the Internet, thus contributing towards the vision of NetMedicine.

Introduction

The usefulness of medical data is greatly increased when data are available in formats that allow them to be integrated with other data. A well-known strategy for data integration, medical or otherwise, is through the annotation of multiple bodies of data using common controlled vocabularies or *ontologies* (Guarino, 1998). However, the very success of this strategy in medical systems has led to a proliferation of medical ontologies, which itself creates obstacles to integration (Bodenreider, 2009). As a result, choosing or creating the right ontology, in combination with associated tools and techniques, can be a non-trivial task.

We are motivated by our participation in the COMMODITY$_{12}$ project (Kafalı et al., 2013), a personal health system (PHS) aiming to support diabetic patients and the medical professionals that treat them. Data integration in this project follows a mediator-based approach (Hernandez & Kambhampati, 2004). Mediator-based integration concentrates on data translation of existing information sources. In COMMODITY$_{12}$ these sources include information entered through a Web interface, sensor information coming from devices that monitor patients, information from databases with patient data, and decision support information provided by logic-based agents. However, the translation between components is done statically, i.e., when a new message type needs to be processed throughout the system, all the translators of the mediator should be updated manually to support the required information flow. The issue then becomes how to standardize the domain knowledge, so that this can be used systematically as if it was originating from a single source.

The aim of this paper is to study how to complement the translation process of the COMMODITY$_{12}$ mediator with ontologies and semantic reasoning support. Our objective is to identify the relevant architecture, tools and techniques to support the development of ontologies so that it can be integrated into our PHS. We are particularly interested in reasoning about information accessed from the patient database or sensor devices in general, and information regarding data exchanges relevant to the profile of a diabetic patient in particular.

To address the aim and objectives of the semantic reasoner for COMMODITY$_{12}$, we study how to embed such a reasoner as a reasoning capability in what we call infrastructure agents, viz., reactive agents deployed to support the smooth (inter)operation of components in a PHS, including support for semantic reasoning and ontologies. We also describe a diabetic patient profile using an ontology formalised in OWL, and we present SWRL rules to provide semantic reasoning on the ontology concepts. We instantiate the ontology with individuals coming from a patient database. We implement the semantic reasoner in Java, and use Protégé to develop the ontology. Moreover, we show how the infrastructure agents that perform semantic reasoning can be combined with the existing logic-based agents of COMMODITY$_{12}$ providing monitoring, advice and diagnosis.

In the remainder of the paper, we first present the COMMODITY$_{12}$ architecture specialised to contain ontology managing components in the form of infrastructure agents. These agents manipulate a diabetes ontology, the details of which we describe next. We then present an illustrative example where we demonstrate how our approach works on patient data. Finally, we conclude the paper by summarising our contribution, comparing it with existing literature, and outlining our plans for extending it in the future.

COMMODITY$_{12}$ extended with Semantic Reasoning

COMMODITY$_{12}$ (COntinuous Multi-parametric and Multi-layered analysis Of DIabetes TYpe 1 & 2) aims to design, build, and validate an intelligent system for the analysis of multi-parametric medical data. It will uptake the existing cutting-edge technologies and extend these technologies by combining state-of-the-art networks, software interoperation, and artificial intelligence techniques in order to realize the concept of translational medicine by means of a PHS. A prototype PHS has been developed (Kafalı et al., 2013) that allows patients to take measurements of vital signals through a set of wearable sensors (e.g., heart rate and respiration rate), and enables doctors to input data about their patients via a web interface. This information is then processed to provide advice to the patients as well as alerts to the doctors. The main reasoning is provided by logic-based agents, built according to the $\mathcal{LAMA}$ agent model and deployed using the GOLEM agent platform (Bromuri & Stathis, 2008). The $\mathcal{LAMA}$ agents are developed using artificial intelligence tools and techniques combining deductive, hypothetical and temporal reasoning. An agent's knowledge-base consists of logic rules formulated according to medical guidelines plus a set of facts describing the environment in which the agent is situated. The left part of Figure 1 demonstrates the COMMODITY$_{12}$ architecture.

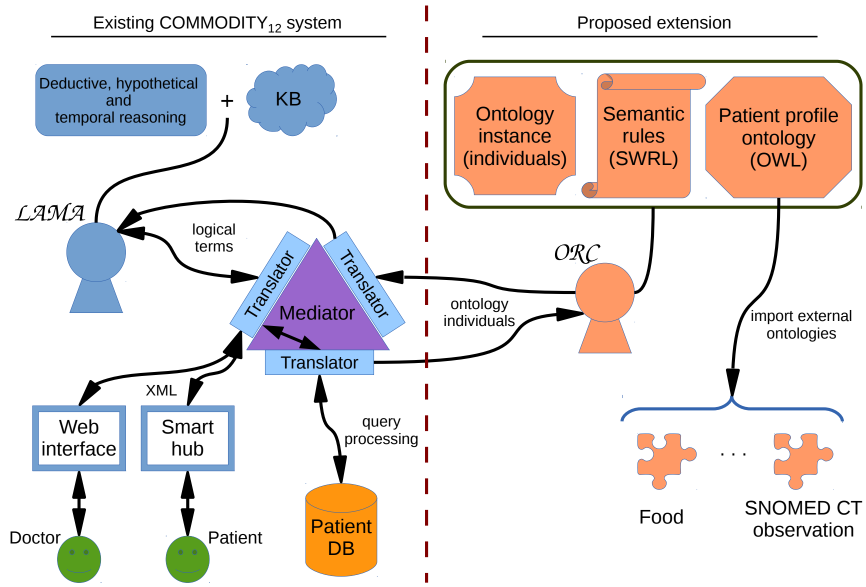

Figure 1. Architecture showing how to extend the COMMODITY$_{12}$ PHS (Kafalı et al., 2013) with ontologies and $\mathcal{ORC}$ agents to support semantic reasoning for diabetes patient profiles.

Data integration in COMMODITY$_{12}$ follows a mediator-based approach (Hernandez & Kambhampati, 2004) which concentrates on translation of data coming from different information sources. These sources feed valuable information to the system that include: (i) measurements from the patients via sensors, (ii) observations from doctors following patient visits, or (iii) profile information from the patient DB. Note that some of the arrows in Figure 1 are double-sided, representing flow of information to both sides, e.g., the doctor enters data about the patient which is sent to the $\mathcal{LAMA}$ agent for processing, and the resulting alert is displayed via the doctor's screen.

One major problem regarding the current architecture of the system is that each translator is specialised in a specific part of the system, which makes the effort of adding new concepts harder. Consider the following scenario: a new rule is added to the $\mathcal{LAMA}$ agent that requires the concepts of the patient's systolic blood pressure and living place to decide whether an e-consultation is justified[1]. Because there are multiple code systems that use different codes for these concepts, the translator needs to map all the codes onto something that the translator understands. Consequently, the same functionality would be implemented in all translators, which makes maintenance and testing costly and inefficient. Adding an ontology reasoner would not only standardize the domain knowledge at a single source, thus reducing the effort of maintenance and introducing new concepts, but it also aids in disease management through the use of patient profiles and the new information that is inferred with the ontology. To address this need, we have developed the $\mathcal{ORC}$ agent. It is developed in Java, provides semantic reasoning and consists of the following components as shown in the right part of Figure 1:

[1]E-consultation means that a specialist logs into the system, looks into the patient's health records, studies the patient's observation values, and provides advice without the patient having to visit the specialist.

- an OWL[2] ontology describing diabetes patient profiles and importing other relevant ontologies such as food and medical observations;
- patient data asserted to the ontology as OWL individuals, e.g., age, living place, blood pressure measurements;
- a set of SWRL[3] rules that are used to infer new properties about existing individuals;
- a processing component that retrieves patient information and infers ontology properties.

Currently, $\mathcal{ORC}$ agents are purely reactive in that they only respond to requests coming from $\mathcal{LAMA}$ agents. When such a request comes regarding the retrieval of a specific patient profile information, $\mathcal{ORC}$ first loads the ontology and retrieves relevant data from the patient DB. This data is then translated to OWL syntax and asserted to the ontology as individuals. The next step is to execute the SWRL rules to infer new properties about the individuals. These are then translated to logical terms and passed to the $\mathcal{LAMA}$ agent that needs those terms for further reasoning.

```
advise_e_consultation(Patient, T):-
   has_age(Patient, Age, T), Age < 80,
   has_no_hospital_access(Patient, T),
   \+ consulted(Patient, internist, years, 1, T),
   has_systolic_bps(Patient, [MostRecent, SecondMostRecent|Rest], months, 6, T),
   MostRecent > 140, SecondMostRecent > 140.
```

Listing 1: Prolog rule for advising e-consultation in the case of high blood pressure.

Let us now explain the functionalities of each agent via a rule for determining whether the patient needs e-consultation, based on the blood pressure monitoring outcome as well as other profile information (Sluiter et al., 2012). The rule in Listing 1 is given in Prolog syntax in the form of conclusion holds (the head of the rule) if (represented by the :- symbol) conditions hold (the body of the rule). $\backslash+$ is the negation-as-failure operator. A call of the form $\backslash+$P succeeds if all possible ways of solving P fail finitely. T represents the current time. The first two conditions check whether the patient is less than 80 years old and has no easy access to a hospital, respectively. Now, certain parts of this rule can be improved via semantic reasoning both in terms of (i) access to / inference on data and (ii) knowledge representation / development effort. For example, the second condition can be inferred by the $\mathcal{ORC}$ agent by looking at the patient profile and checking whether the patient lives in a rural area or not (providing inference on data). The last two conditions check whether the patient has not consulted an internist in the last year, and her last two blood pressure measurements are higher than 140 `mmHg` in the last six months, respectively. Evaluating these two conditions requires temporal reasoning which is provided by the $\mathcal{LAMA}$ agent. Observe that the rule uses negation-as-failure, which is acceptable here because the database holds complete knowledge of the data underlying the negated clauses. Systolic blood pressure can be measured as part of different examinations, which can result in relevant data being stored using different codes. However, the rule is only representing the generic concept, so that any type of systolic blood pressure measurement qualifies as a valid measurement for this rule. Coming back to the improvements provided by semantic reasoning, using an ontology with a class hierarchy for the systolic blood pressure concept, which offered by the $\mathcal{ORC}$ agent, removes the need to describe the sub-class relations in the $\mathcal{LAMA}$ agent (reducing development effort).

[2] `http://www.w3.org/TR/owl-ref/`
[3] `http://www.w3.org/Submission/SWRL/`

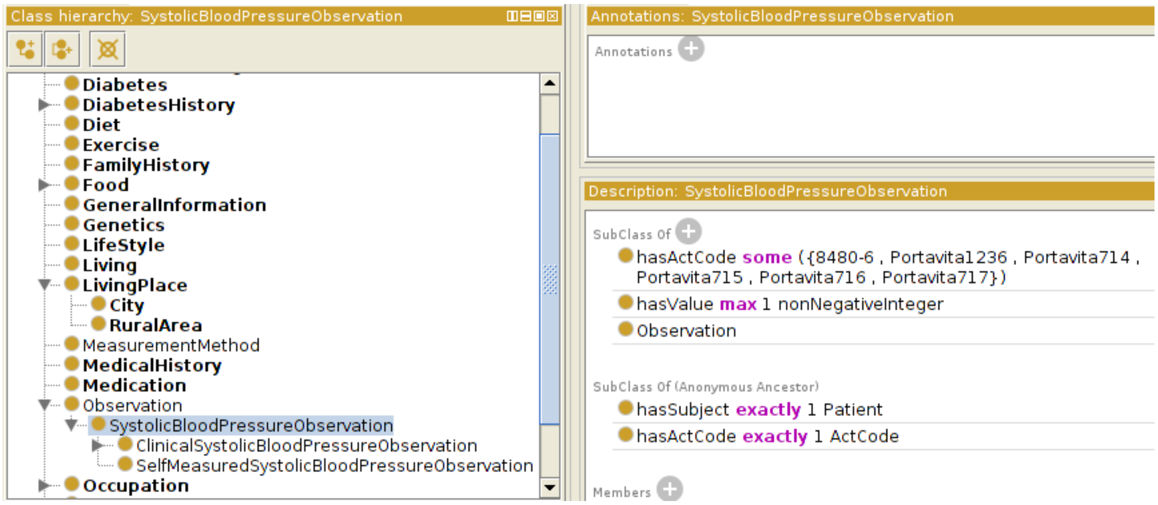

Figure 2. Class hierarchy in Protégé.

An Ontology for Diabetes Patient Profiles

In COMMODITY$_{12}$, we are interested in using an ontology for diabetes management. We started with exploring the literature to find an ontology that was sufficient for our purposes. Unfortunately, our search was unsuccessful (see Discussion section). As a result, we are presenting here our consortium's attempt to design such an ontology for diabetes management. Our approach is patient centric and aims to include only the necessary patient information so that different types of diabetes can be managed via a PHS. In other words, the schema defining the ontology is that of defining a patient profile.

To process information in profiles we embed $\mathcal{ORC}$ in infrastructure agents to support semantic reasoning given a taxonomy of diabetes concepts as well as the logical relations regarding those concepts. These will be represented in OWL, where concepts correspond to classes while the relations are given as either object or data properties (Guarino, 1998). On top of this design, individuals can be created to represent specific instances of classes. Moreover, logic rules described in SWRL allow inference of new properties for individuals. We identify the following classes to describe a patient profile based on the medical knowledge gained from the COMMODITY$_{12}$ consortium:

- *GeneralInformation*: contains information about the patient's age, gender and ethnicity;
- *LifeStyle*: describes the patient's living conditions, exercise habits and alcohol intake;
- *Work*: contains information about the patient's education and occupation;
- *Diet*: contains information about the patient's eating habits and details on restricted food;
- *MedicalHistory*: contains the history of patient's medical records and family history;
- *Observation*: contains the history of observations and measurements regarding the patient;
- *Genetics*: describes the patient's genetic information.

Figure 2 shows part of the class hierarchy for our patient profiles as developed with the ontology development tool Protégé[4]. The highlighted class *SystolicBloodPressureObservation* on the left panel is a sub-class of *Observation*. Thus, it describes a more specific type of observation. Further details of the class are shown on the right panel. Figure 3 demonstrates the data properties for the classes, as well as rules formulated in SWRL. The panel on the left shows part of the data

[4] http://protege.stanford.edu/

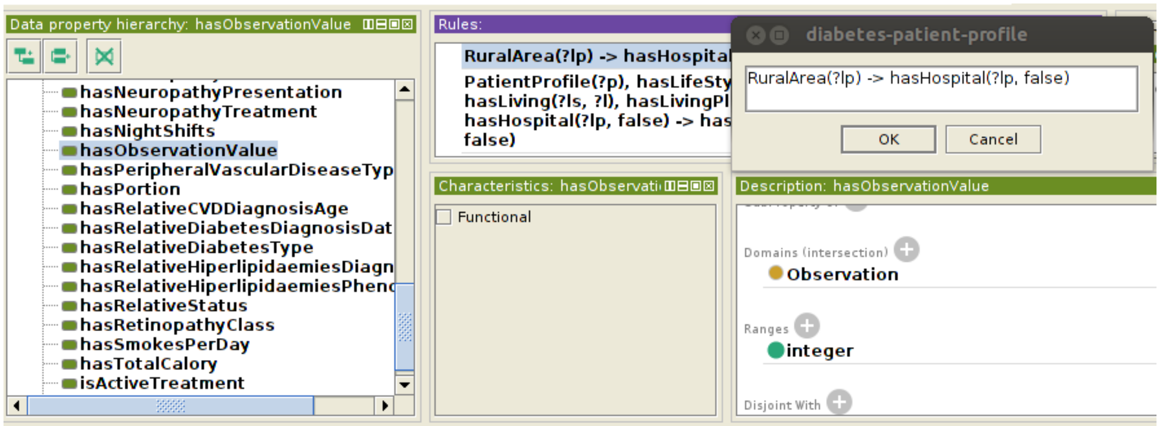

Figure 3. Data properties and SWRL rules in Protégé.

properties defined in our ontology. The highlighted property *hasObservationValue* has the class *Observation* as its domain, and describes the measured value as an integer (bottom right panel). The properties of a particular class are inherited by its children (i.e., subclasses), so that all types of blood pressure observations have this property. The top right tab describes a simple SWRL rule.

Following our running example, the highlighted rule infers that a rural area has no access to a hospital. This kind of reasoning allows us to avoid describing in the $\mathcal{LAMA}$ agent's knowledge-base whether a hospital is easily accessible for the patient. That is, the $\mathcal{ORC}$ agent provides this information via semantic reasoning. Lack of space does not allow us to provide the full details here, but we hope to report them in future work.

An Illustrative Example: The Portavita Patient Database

We have developed a prototype implementation for the $\mathcal{ORC}$ agent and we are in the process of integrating it with the COMMODITY$_{12}$ system through the mediator interface. The current implementation as well as the OWL design for the ontology is available online[5]. Here, we demonstrate the workings of our approach using the sample rule given in Listing 1 on patient data extracted from the Portavita database. Recall that one of the conditions in the rule is related to the concept *systolic blood pressure*. Measurements of systolic blood pressure can enter the database as part of multiple examinations performed by doctors, with a company-specific coding system used for classifying these measurements.

In order to unify the concepts for describing blood pressure with a standardized vocabulary, we have integrated the Portavita-specific ontology with the widespread SNOMED CT ontology. Figure 4 illustrates this integration. The lower part shows the class hierarchy for the Portavita ontology focusing on the *systolic blood pressure* concept, of which the class *SystolicBloodPressure-Observation* is shown in more detail in Figure 2. The upper part demonstrates the relevant concepts from the SNOMED CT ontology. Note that SNOMED CT's *Systolic blood pressure (observable entity)* is mapped to the *ClinicalSystolicBloodPressureObservation* class from the Portavita ontology, as illustrated by the arrow that crosses the dashed line. This denotes an OWL *equivalence* relation. Mapping a custom ontology onto a standardized one in such a way has the benefit of giving flexibility as to where equivalence or subclass relations are put; if one wishes to, e.g., re-

[5] `http://dice.cs.rhul.ac.uk/article.php?id=12`

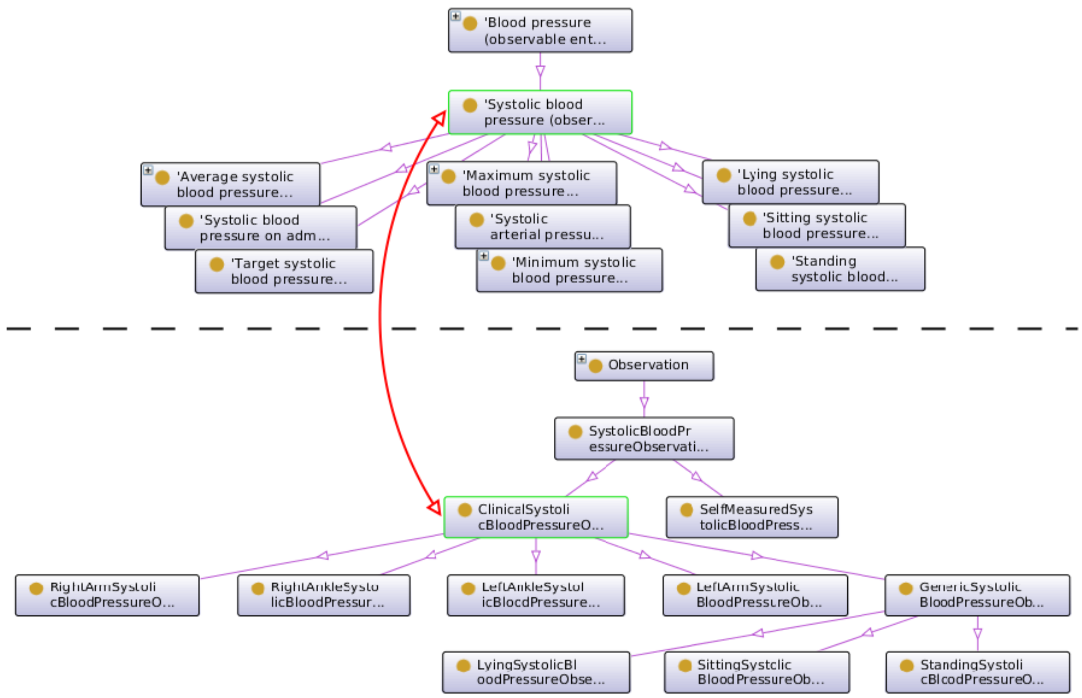

Figure 4. Correspondence of blood pressure concepts in the ontologies of SNOMED CT (above dashed line) and Portavita (below dashed line).

gard instances of Portavita's *SelfMeasuredBloodPressure* concepts as instances of SNOMED CT's concept of *Systolic blood pressure*, then it would suffice to "lift" the equivalence relation one level up in the Portavita ontology. Additionally, one could posit an equivalence relation between *Sitting systolic blood pressure (observable entity)* and *SittingSystolicBloodPressureObservation*.

It was noted before that systolic blood pressure is stored as part of several examinations, amongst which the Ankle Brachial Pressure Index (ABPI) and a generic blood pressure examination. A single code is used for storing generic blood pressure observations, along with an optional attribute that can be used to further specify whether the patient was lying, sitting, or standing. In this respect, the Portavita code system oversimplifies the blood pressure observation with respect to SNOMED CT, even though the measurement method (lying/sitting/standing) can be distinguished in a different way. In contrast, in Portavita's ABPI examination the observations of systolic blood pressure at the left / right arms and ankles are stored with four separate codes, thus specializing the concept further than SNOMED CT does in this respect. Apart from enabling cross-relations with standardized ontologies, the definition of a custom ontology makes such matters explicit, and thereby can provide novel insights regarding the underlying data model.

We now consider the patient data in Table 1 with regards to the rule of Listing 1 to determine whether the patient applies for an e-consultation. In order to perform semantic reasoning on this data, first the $\mathcal{ORC}$ agent uses the data in Table 1 to assert the individuals regarding this patient to the ontology as illustrated by the pseudo-code in Listing 2. To relate codes to concepts, $\mathcal{ORC}$ consults the *hasActCode* property in the ontology to find concepts that have a given act code. For example, *8480-6* is translated to *GenericSystolicBloodPressureObservation* and *Portavita714* to *ClinicalSystolicBloodPressureObservation* (see Figure 4).

Patient ID	Age	Living Place
82319	73	Lutjebroek

Patient ID	Observation ID	Measurement Date	Code	Value
82319	11783457	25-01-2013	8480-6	133
82319	11827346	15-02-2013	Portavita714	141
82319	12053457	01-04-2013	8480-6	157

Table 1
Information stored for patient with ID 82319: administrative data shown on the top table, observations shown on the bottom table with codes 8480-6 for generic systolic blood pressure and Portavita714 for systolic blood pressure in left ankle.

```
for every patient p:
  add to the ontology:
    individual      p.patientId   with class     "PatientProfile"
    individual      p.livingPlace with class     "LivingPlace"
    object property p.patientId  "hasLivingPlace" p.livingPlace
    object property p.patientId  "hasAge"         p.age

for every observation o:
  add to the ontology:
    individual      o.observationId  with class          o.code
    data property   o.observationId "hasValue"           o.value
    data property   o.observationId "hasMeasurementDate" o.measurementDate
    object property o.patientId     "hasObservation"     o.observationId
```

Listing 2: Pseudo-code to instantiate the ontology with data from the database.

After the individuals are created, the SWRL rules are executed to infer additional information about the individuals by refreshing the reasoner and realizing its knowledge-base. After this step, it follows from the knowledge-base that both types of systolic blood pressure observations in Table 1 are instances of *ClinicalSystolicBloodPressureObservation*. This greatly simplifies querying data for the rule of Listing 1, which pertains to systolic blood pressure in general.

Once the ontology reasoning is performed, the resulting individuals are translated to logical terms. Part of the output produced by the $\mathcal{ORC}$ agent is shown in Listing 3. Note that the predicate *has_no_hospital_access* is inferred since the patient lives in *Lutjebroek*, which is regarded as a rural area. Furthermore, to obtain all systolic blood pressure observations, $\mathcal{ORC}$ can now simply query for all individuals of type *ClinicalSystolicBloodPressureObservation* and translate those to the format that the $\mathcal{LAMA}$ agent requires. According to this data and the rule, the $\mathcal{LAMA}$ agent draws the conclusion that an e-consultation is suitable for the patient.

```
has_no_hospital_access(patient_82319, 13-05-2013).
has_systolic_bp(patient_82319, 133, 25-01-2013).
has_systolic_bp(patient_82319, 141, 15-02-2013).
has_systolic_bp(patient_82319, 157, 01-04-2013).
```

Listing 3: Output of $\mathcal{ORC}$.

Discussion

We have presented $\mathcal{ORC}$, an ontology reasoning component that builds upon existing ontology modelling tools and techniques to support the integration and interpretation of multimodal medical information. We have illustrated how to embed $\mathcal{ORC}$ as a reasoning capability in reactive infrastructure agents supporting intelligent agents operating in COMMODITY$_{12}$, a personal health environment for diabetic patients and the medical professionals that treat them. The benefits of the approach have been exemplified by showing how medical information for patient profiles at different sources can be included in COMMODITY$_{12}$, thus extending the generality and potential of the system. The approach also illustrates how ontologies can be combined with a variety of artificial intelligence tools and techniques to support e-Health activities on the Internet.

Approaches using ontologies and semantic reasoning in medical knowledge already exist (Bouamrane, Rector, & Hurrell, 2008; Cantais, Dominguez, Gigante, Laera, & Tamma, 2005; Curé, 2005; Latfi, Lefebvre, & Descheneaux, 2007; Li & Ko, 2007). However, as we are interested in the representation of ontologies for disease management in general and the management of diabetes in particular, we are going to avoid comparing our work with application of ontologies in telehealth environments such as smart homes for helping elderly people (Latfi et al., 2007), or similar frameworks.

We will focus on approaches that complement our work in the application context, for instance that of Cantais et al. (2005), who present a food ontology to help diabetic patients with their diet. Although our focus has been on more general coverage of a diabetic patient profile, we foresee that a food ontology could strengthen what we have presented here. Other approaches complement our work in terms of techniques. In (Bouamrane et al., 2008), an ontology-driven approach is used for medical history modelling to customise patient questionnaires. The system uses semantic reasoning to compute similarities among different illnesses so that it can generate further questions based on previous answers provided by the patient. This idea of similarity metrics can complement the way we process historical profile information and the conclusions we draw from such information, e.g., to detect patients with abnormal working hours which can be used to adapt the patient's medication accordingly.

Another possible use of ontologies is presented by Curé (2005), who propose a self-medication system with the use of drug and symptom databases, and an electronic health record for patients. While such a work partly describes medical concepts about diabetes, we are not aware of any work that operates in the context of a personal health system like ours, i.e., starting from the retrieval of diabetic patient records from a medical database as well as aggregating information from a variety of sensors, to displaying the diagnosis to the doctor / patient using agent technology.

Semantic reasoning with ontologies in our work has been embedded as an agent capability for special types of agents that we have called infrastructure agents. These agents' sole task is to reason about ontologies, and they are similar to ontology agents used in FIPA[6]. The adoption of such agents in COMMODITY$_{12}$ not only allows us to support data integration in a systematic way but also enables semantic interoperability between agents that communicate with each other to provide monitoring, advice and diagnosis.

We are currently working towards extending the work presented here in three different directions: (i) We provided a generic ontology for diabetes patient profiles resulting from the medical knowledge gathered in the COMMODITY$_{12}$ consortium, and extended it with observations regard-

[6]www.fipa.org

ing blood pressure concepts as described in Portavita and SNOMED CT. We plan to integrate other ontologies such as for food and alcohol, to enrich the concepts regarding a patient's diet, e.g., calory values for meals, alcohol intake based on drink types. (ii) We created a prototype implementation for the $\mathcal{ORC}$ agent which consists of specialised translators to convert concepts among different components. We plan to provide generic translators based on a concept ontology such that they can be reused and updated from a single source. A sample translation from OWL to Prolog is given in (Samuel et al., 2008). (iii) We believe that the semantic reasoner component ($\mathcal{ORC}$) improves the overall efficiency of the system by simplifying some of the agent reasoning performed by the temporal reasoner ($\mathcal{LAMA}$). We plan to run experimental tests to evaluate our system using patient data from the clinical trials that are to be conducted as part of COMMODITY$_{12}$.

Acknowledgements

This work was partially supported by the EU FP7 Project COMMODITY$_{12}$ (`www.commodity12.eu`). We thank the medical doctors in the COMMODITY$_{12}$ consortium for providing us with the information for describing a diabetic patient profile.

References

Bodenreider, O. (2009, January). Special issue: Biomedical ontology in action. *Appl. Ontol.*, *4*(1), 1–4.

Bouamrane, M.-M., Rector, A., & Hurrell, M. (2008). Using ontologies for an intelligent patient modelling, adaptation and management system. In *Proceedings of the OTM 2008 Confederated International Conferences* (pp. 1458–1470).

Bromuri, S., & Stathis, K. (2008). Situating cognitive agents in golem. *Engineering environment-mediated multi-agent systems*, 115–134.

Cantais, J., Dominguez, D., Gigante, V., Laera, L., & Tamma, V. (2005). An example of food ontology for diabetes control. In *Proceedings of the International Semantic Web Conference 2005 workshop on Ontology Patterns for the Semantic Web.*

Curé, O. (2005). Ontology interaction with a patient electronic health record. In *CBMS '05: Proceedings of the 18th IEEE Symposium on Computer-Based Medical Systems* (p. 185-190).

Guarino, N. (1998). *Formal ontology in information systems*. IOS Press.

Hernandez, T., & Kambhampati, S. (2004, September). Integration of biological sources: current systems and challenges ahead. *SIGMOD Rec.*, *33*(3), 51–60.

Kafalı, Ö., Bromuri, S., Sindlar, M., van der Weide, T., Pelaez, E. A., Schaechtle, U., … Stathis, K. (2013). COMMODITY$_{12}$: A smart e-health environment for diabetes management. *Journal of Ambient Intelligence and Smart Environments, IOS Press (To appear).*

Latfi, F., Lefebvre, B., & Descheneaux, C. (2007). Ontology-based management of the telehealth smart home, dedicated to elderly in loss of cognitive autonomy. In *OWLED.*

Li, H.-C., & Ko, W.-M. (2007). Automated food ontology construction mechanism for diabetes diet care. In *2007 International Conference on Machine Learning and Cybernetics* (Vol. 5, p. 2953-2958).

Samuel, K., Obrst, L., Stoutenberg, S., Fox, K., Franklin, P., Johnson, A., … Peterson, J. (2008, May). Translating OWL and Semantic Web Rules into Prolog: Moving toward description logic programs. *Theory Pract. Log. Program.*, *8*(3), 301–322.

Sluiter, A., van Wijland, H., Arntzenius, A., Bots, A., Dijkhorst-Oei, L.-T., van der Does, F., … van Balen, J. (2012). Landelijke Transmurale Afspraak Diabetes mellitus type 2. *Huisarts Wet*, *55*(1), S1–12.