An Ontology-Driven Agent-Based Clinical Guideline Execution Engine

David Isern, David Sánchez, and Antonio Moreno

ITAKA Research Group - Intelligent Tech. for Advanced Knowledge Acquisition
Dept. of Computer Science and Mathematics. University Rovira i Virgili*
43007 Tarragona, Catalonia (Spain)
{david.isern,david.sanchez,antonio.moreno}@urv.cat

Abstract. One of the hardest tasks in any healthcare application is the management of knowledge. Organisational information as well as medical concepts should be represented in an appropriate way in order to improve interoperability among existing systems, to allow the implementation of knowledge-based intelligent systems, or to provide high level support to healthcare professionals. This paper proposes the inclusion of an especially designed ontology into an agent-based medical platform called HeCaSe2. The ontology has been constructed as an external resource, allowing agents to coordinate complex activities defined in any clinical guideline.

1 Introduction

In order to exploit the great potential that clinical practice guidelines (GLs) offer to improve patient's care delivery quality, tools for adopting them within the clinical routine are required [1]. In this sense, one of the biggest problems is the gap between the codification of these GLs and their use/interpretation in a real organisation. One of the solutions consists on adding content background to the GLs by the use of medical *ontologies*.

The use of ontologies in medicine supposes an important advantage in order to provide a common understandable framework to make explicit the involved medical concepts as well as their relations and properties. Ontologies also provide a high level model of the daily work flow that can be adapted to the particular circumstances of any healthcare organisation. Kumar et. al. [2] studied the implementation of a task ontology named Context-Task Ontology (CTO) to map the knowledge required in the implementation of GLs. They noted that this approach had some drawbacks, such as the difficulty to define and know exactly which relations are needed, as well as the requirement of expert's intervention. The same authors later described the use of ontologies to define clinical guidelines by adding a hierarchy of classes to represent medical procedures and plans

* The work has been partially supported by the K4CARE project (IST-2004-026968). D. Sánchez would like to acknowledge the support of *Departament d'Innovació, Universitats i Empresa de la Generalitat de Catalunya i del Fons Social Europeu.*

R. Bellazzi, A. Abu-Hanna, and J. Hunter (Eds.): AIME 2007, LNAI 4594, pp. 49–53, 2007.

[3]. However, this implied a high level of complexity as compared to flow-chart-based representations. Serban et al. [4] proposed the use of an ontology to guide the extraction of medical patterns contained in GLs in order to reconstruct the captured control knowledge. All these works suggest the use of UMLS as a central corpus. Ciccarese et al. [5] introduced an architecture that linked a care flow management system and a guideline management system by sharing all the data and ontologies in a common layer. They proposed to represent medical and organisational information in those ontologies, but they did not use non taxonomic relations in the ontologies. Moreover, Davis and Blanco [6] suggested the use of taxonomies to model the clinical life cycle knowledge. They also described a state-based data flow model to represent all dependencies between enterprise entities.

Previous papers ([7,8]) introduced an agent-based system called HeCaSe2 that proposes an open architecture that models different entities of a healthcare organisation. HeCaSe2 includes the management of clinical guidelines by doctors in the diagnosis or treatment of diseases. All these tasks require an external element to be more flexible and efficient: a representation of the care flow and the terminology used across all entities. In order to address this issue, this paper proposes the inclusion in HeCaSe2 of an application ontology that covers three main areas: *a*) representing all medical terminology used by all partners, *b*) modelling healthcare entities with its relations, and *c*) collecting semantic categories of those medical concepts.

2 HeCaSe2: A Distributed Guideline-Based Health Care System

HeCaSe2 is a dynamic multi-agent system that maps different entities in a healthcare organisation (*i.e.* medical centres, departments, services, doctors, patients) as agents with different roles. This system provides interesting services both to patients (*e.g.* booking a visit with a doctor, or looking up the medical record) and to doctors (e.g. support in the application of a GL to a patient). Guidelines are used to provide a high level supervision of the activities to be carried out to address a specific pathology.

At the top, the patients are represented in the system by *User Agents* (UA). Any UA can talk with the *Broker Agent* (BA). The BA is the bridge between users and medical centres, and it is used to discover information about the system. All UAs can ask this agent in order to find medical centres satisfying certain criteria. The BA covers the medical centres located in a city or in an area. Any user can access the system through a *Medical Centre Agent* (MCA) that centralises and monitors the outsider's accesses. A MCA monitors all of its departments, represented by *Department Agents* (DAs), and a set of general services (represented by *Service Agents* (SAs)). Each department is formed by several doctors (represented by *Doctor Agents* (DRA)) and more specific services (also modelled as SAs). Moreover, in each department there is a *Guideline Agent* (GA) that performs all actions related to guidelines, such as looking for a desired

GL, storing and/or changing a GL made by a doctor, etc. This GA contains only GLs related to the department where it is located (the knowledge is close to the entity that will use it). Each department also contains an *Ontology Agent* (OA) that provides access to the designed medical ontology and complements the information provided by the GA. At the bottom of the architecture there is the *Medical Record Agent* (MRA) which stores all patient medical records in the medical centre.

3 Ontological Representation of Medical Knowledge

Ontologies define the basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary [9]. From the available representation languages, the medical ontology has coded using *OWL DL* ([10]) and constructed following the *101* methodology [11].

The designed ontology is composed by relations established among the agents associated to a healthcare organisation. It has three main groups of concepts: *a*) *agent-based health care concepts*, *b*) *semantic types* of the used concepts (entities and events), and *c*) *medical concepts* related to the managed GLs (see Fig. 1).

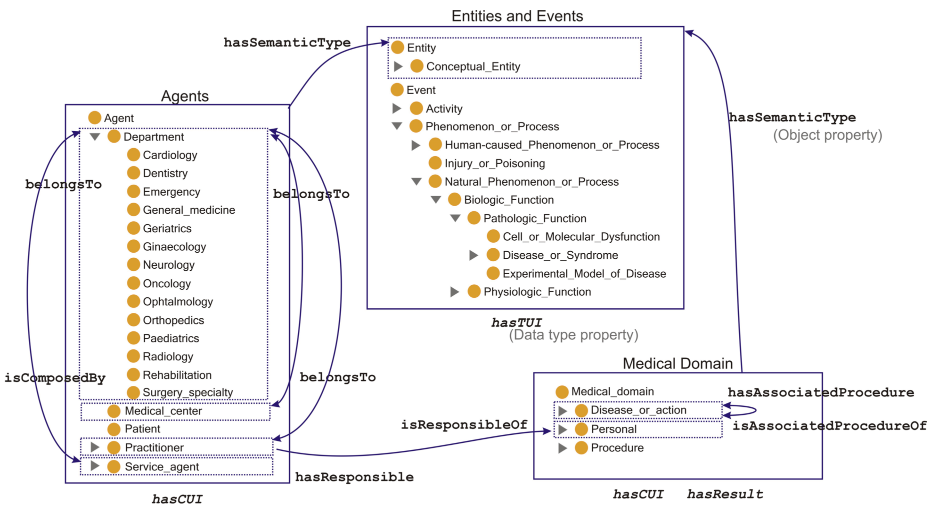

Fig. 1. Subset of the designed *Medical Ontology*

The first group of concepts concerns the multi-agent system. The *Agent* class encloses all main concepts and properties related with the internal organisation. That portion of the medical ontology is composed by *Medical centres*, *Departments*, *Patients*, *Practitioners* and *Services*.

All these elements have internal relations, such as *Oncology* `is-a` *Department* that `belongsTo` *Medical-center* which `isComposedBy` *Service_agents*. More complex relations between doctors and services are also mapped, such as *Nurse* `belongsTo` *Department* because a nurse can be located in any department, or *Family_doctor* `belongsTo` (*General medicine* ∪ *Emergency* ∪ *Paediatrics*) that means that an instance of family doctor could belong to any instance of three departments. Relations between *Agent* subclasses are inspired in the typical structure of healthcare organisations. The inverse relations are also available in order to know which kind of doctors compose a department or which kind of services are located in a department or medical centre.

Although most of the departments are similar in medical centres, it is possible to represent different variations. In those cases, a specialisation of the ontology could be made by creating subclasses. The parent class would keep all common features and the siblings would contain specific features or resources for each one.

The next set of concepts are the semantic types of medical terms according to its context. That portion of the ontology is intended to avoid language ambiguity, and the UMLS Semantic Network was used.

Currently, UMLS defines 135 different semantic types divided in two groups: meanings concerned with healthcare organisations or entities, and meanings related with events or activities in a daily care flow. Those hierarchies are named *Entity* and *Event* respectively, and are organised as a taxonomy with `is-a` relations between concepts, such as *Disease_or_Syndrome* `is-a` *Pathologic_function*.

All this information is used by agents to know exactly which is the function of any required concept and further connections with others. For instance, if a concept is a *Finding*, and a *Finding* `isResponsibilityOf` a *Practitioner*, the agent knows that a patient's finding should be given by a practitioner.

Finally, the last part of the ontology represents the specific vocabulary used in clinical guidelines. This part systematises all specific knowledge required in any guideline execution engine, divided in *Diseases*, *Procedures* and *Personal data*. It is necessary to define a set of relations between each concept and its identifier (*Code Unique Identifier* or CUI), its semantic type, which entity of the system is responsible of its accomplishment, and the produced result (*i.e.* if it as number, a Boolean, an enumerate or a complex object). The established relations are bidirectional because it is interesting to know that the finding *Active_cancer* `isResponsibilityOf` a *Family_Doctor*, and also the inverse relation is important for the family doctor in order to know his responsibilities. Each agent can access the concepts related to its own domain and be aware of the consequences of the application of an action.

4 Conclusions

The inclusion of a medical ontology in the HeCaSe2 multi-agent system has been discussed. As shown in the introduction, the use of ontologies in the medical domain is increasing and offers some advantages such as making domain assumptions explicit, separating domain knowledge from operational knowledge and sharing a consistent understanding of what information means.

In the present work, the designed medical ontology brings the following advantages to the guideline-based execution system: a) to identify the required actors that are able to accomplish an action and to know the source of a data, b) to adapt the execution framework to the particular casuistry of any healthcare organisation without modifying the MAS implementation, and c) to provide an application independent context. Thus, by changing the ontology and its relations, the execution procedure is changed.

Note that the only issue that should be addressed is the manual definition of the appropriate task ontology. This question usually requires the intervention of a domain expert but UMLS provides a large corpus of concepts and relations that can be reused.

References

1. Quaglini, S., Stefanelli, M., Cavallini, A., Micieli, G., Fassino, C., Mossa, C.: Guideline-based careflow systems. Artif Intell Med 20, 5–22 (2000)
2. Kumar, A., Quaglini, S., Stefanelli, M., Ciccarese, P., Caffi, E.: Modular representation of the guideline text: An approach for maintaining and updating the content of medical education. Medical Informatics and the Internet in Medicine 28, 99–115 (2003)
3. Kumar, A., Ciccarese, P., Smith, B., Piazza, M.: Context-Based Task Ontologies for Clinical Guidelines. In: Pisanelli, D.M. (ed.) Ontologies in Medicine. Studies in Health Technology and Informatics, vol. 102, pp. 81–94. IOS Press, Amsterdam (2004)
4. Serban, R., Teije, A.t., Harmelen, F.v., Marcos, M., Polo-Conde, C.: Extraction and use of linguistic patterns for modelling medical guidelines. AI in Medicine 39, 137–149 (2007)
5. Ciccarese, P., Caffi, E., Boiocchi, L., Quaglini, S., Stefanelli, M.: A Guideline Management System. In: Fieschi, M., Coiera, E., Li, Y.-C.J. (eds.) 11th World Congress on Medical Informatics, MedInfo 2004, San Francisco, USA. Studies in Health Technology and Informatics, vol. 107, pp. 28–32. IOS Press, Amsterdam (2004)
6. Davis, J.P., Blanco, R.: Analysis and Architecture of Clinical Workflow Systems using Agent-Oriented Lifecycle Models. In: Silverman, B.G., Jain, A., Ichalkaranje, A., Jain, L.C. (eds.) Intelligent Paradigms for Healthcare Enterprises. Studies in Fuzziness and Soft Computing, vol. 184, pp. 67–119. Springer, Heidelberg (2005)
7. Isern, D., Moreno, A.: Distributed guideline-based health care system. In: 4th International Conference on Intelligent Systems Design and Applications (ISDA-2004), Budapest, Hungary, pp. 145–150. IEEE Press, Orlando (2004)
8. Isern, D., Valls, A., Moreno, A.: Using aggregation operators to personalize agent-based medical services. In: Gabrys, B., Howlett, R.J., Jain, L.C. (eds.) KES 2006. LNCS (LNAI), vol. 4252, pp. 1256–1263. Springer, Heidelberg (2006)
9. Neches, R., Richard, F., Finin, T., Gruber, T., Patil, R., Senator, T., Swartout, W.: Enabling Technology for Knowledge Sharing. AI Magazine 12, 36–56 (1991)
10. McGuinness, D., Harmelen, F.v.: OWL Web Ontology Language (2004)
11. Noy, N.F., McGuinness, D.L.: Ontology Development 101: A Guide to Creating Your First Ontology. Technical Report SMI-2001-0880, Stanford Medical Informatics (2001)