The HealthAgents ontology: Knowledge representation in a distributed decision support system for brain tumours

BO HU[1,5], MADALINA CROITORU[2], ROMAN ROSET[3], DAVID DUPPLAW[1], MIGUEL LURGI[3], SRINANDAN DASMAHAPATRA[1], PAUL LEWIS[1], JUAN MARTÍNEZ-MIRANDA[3] and CARLOS SÁEZ[4]

[1] *ECS, University of Southampton, Southampton SO17 1BJ, UK*
E-mail: {bh, dpd, sd, phl}@ecs.soton.ac.uk
[2] *LIRMM, 161 rue ADA, F34392 Montpellier Cedex 5, Montpellier, France*
E-mail: croitoru@lirmm.fr
[3] *MicroArt, Parc Cientific de Barcelona, Baldiri Reixac 4-6, 08028, Barcelona, Spain*
E-mail: {rroset, mlurgi, jmartinez}@microart.eu
[4] *ITACA - Universidad Politécnica de Valencia, Spain*
E-mail: carsaesi@upvnet.upv.es
[5] *SAP Research, Belfast BT37 0QB, UK*
E-mail: bo01.hu@sap.com

Abstract

In this paper we present our experience of representing the knowledge behind HealthAgents, a distributed decision support system for brain tumour diagnosis. Our initial motivation came from the distributed nature of the information involved in the system and has been enriched by clinicians' requirements and data access restrictions. We present in detail the steps we have taken towards building our ontology starting from knowledge acquisition to data access and reasoning. We motivate our representational choices and show our results using domain examples employed by clinical partners in HealthAgents.

1 Introduction

HealthAgents (HA) (González-Vélez et al., 2009) aims to develop an agent-based, *distributed* decision support system (d-DSS) that employs clinical information, Magnetic Resonance Imaging (MRI) data, spectral output from Magnetic Resonance Spectroscopy (MRS), high-resolution magic angle spinning spectroscopy (HR-MAS) and cDNA Microarray gene expression data. The aim of this project is to help improve brain tumour management by providing non-invasive alternatives to biopsies for diagnosis. A predecessor project, INTERPRET[1], has shown that single voxel MRS data can aid in improving brain tumour classification. HA builds on these results and further employs multi voxel MRS data, as well as HR-MAS and gene expression data for a more comprehensive picture to guide diagnosis. Moreover, HA has built a *d*-DSS. Its distributed nature allows the system to benefit from participation of other clinical centres than those originally contributing data to the project. In this way the evidence base for enhanced classification performance is increased. The HA system is designed and built as a multi-agent system, with great emphasis on declarative representations for agent interfaces to data, other agents and human users. They bring up a diverse set of concerns which are accommodated in the developed knowledge representation schemata.

[1] `http://azizu.uab.es/INTERPRET/index.html`

User requirements were acquired through interviewing domain experts from multiple clinical centres. Those user requirements directly applied to the problem of representing knowledge inside the system encompass the following main aspects:

- System Functionality: given the distributed nature of the system, the ontology has to function primarily as a common inter-lingua for different knowledge bases. While this functionality could be achieved via an encompassing database schema, using an ontology makes expressive knowledge encoding easier. Indeed, while the project now functions with a comprehensive range of methods for brain tumour diagnosis (such as MRI, MRS, HR-MAS, gene expression profiles) which are being flexibly added to the diagnostic mix in different clinical centres, it is likely that other modalities will be added in the future. Since an ontology makes these ingredients explicit, it is used in this project to serve as a common vocabulary and provide access to databases in an integrated manner.
- Clinician Terminology: following on from the previous requirement the ontology has to act as a shared conceptualisation of the application needs. This means that the terminology employed has to be validated by the clinical users of the system, and the tests leading to the validation cut across hierarchical abstractions introduced in knowledge engineering, and often focus on the use of familiar terminologies in specific work-spaces. Moreover, different hospitals might use standard nomenclature that refer to the same object in different ways. As a consequence, decisions about nomenclature have to be taken in close collaboration with the clinical partners.
- Legacy System Integration: a consequence of the two above mentioned requirements, but still an important element in itself is the smooth translation from existing data descriptors (such as database files, application dependent parsing files, etc.) towards the agreed nomenclature. This is not a straightforward process and its difficulties (both from the perspective of semantic tradeoffs and a technological viewpoint) have to be carefully analysed and addressed. This additional requirement is one of providing mappings between the ontology and the various legacy database schemata of centres that join the HA network.

This paper reports on the process of meeting the above requirements, addressing the principles and the pragmatism that has shaped their fulfillment.

1.1 Technical Background

Brain tumours remain an important cause of morbidity and mortality and afflict a large percentage of the European population. In children over 1 year of age, brain tumours are the most common solid malignancies that cause disease-related death. Diagnosis using MRI and MRS is non-invasive, but only achieves variable, 60-90% accuracy depending on the tumour type and grade (Julià-Sapé et al., 2006). The current gold standard classification of a brain tumour by histopathological analysis of biopsy is an invasive surgical procedure and in addition to health care costs and stress to patients, incurs a high risk of morbidity. Studies have shown that stereotactic brain biopsy has significant risks, with an estimated morbidity of 2.4-3.5% (Favre et al., 2002; Hall, 1998) and a death rate of 0.2-0.8% (Favre et al., 2002; Field et al., 2001). For tumours that evolve slowly (e.g. pilocytic astrocytoma in children), repeated biopsies may not be advisable nor practical. Non-invasive methods to monitor tumour progression become necessary, so the classification accuracy of methods based on MRS data needs to be improved with the help of additional information coming from HR-MAS and gene expression data. This falls under the ambit of HA.

A centralised Decision Support System (DSS) based on MRS data and histopathological diagnosis for classifier labels, is already available from the INTERPRET project. HA aims to decentralise the process in a distributed decision support framework that allows multi-site data partitioning and sharing. Agent technology is employed to power the *d*-DSS.

Agents encapsulate core chunks of functionality, and the combinatorial possibilities of joining the output of one agent to the input of another generates the overall behaviour of the system aligned to functional specification that users require. As such, the interfaces between agents themselves, between agents and end users and between agents and the clinical data, that require agent-based processing, need to be carefully designed. In a multi-site development environment such as that required for HA, it is these interfaces that can hinder or foster system integration and correct behaviour. Declarative specification of these interfaces help separate platform dependent details of message passing elements from the functionally specific constructs that individual agent developers at remote locations use. This aspect is directly related to the system functionality requirements mentioned in the previous section.

In order to describe the data acted upon by the intelligent processing and classification algorithms at the heart of HA's success, as well as the categories that earmark the output types of these algorithms, we construct the HA domain ontology (HaDOM) as the knowledge representation framework for the system. One of the guiding features of this work was the need to ensure that the domain ontology's concepts and relations could be mapped with relative ease onto database schemata typically used in clinical settings. At the very least, an ontology devised to support intelligent information processing must be capable of answering the same queries that a database-driven system can. This corresponds to the last item of system requirements, namely the smooth transition between the legacy terminologies as employed by various representations and HaDOM. Below we shall describe how we can view an ontology as a construct that organises the set of questions one can ask of a particular domain of knowledge. Syntactical support for this equivocation between declarative definition and interrogative procedures will be discussed at some length. In this context, curation and maintenance of referentially consistent descriptions in the face of variation in terminological practices is an issue that concerns our work from the very outset, and interfacing legacy databases is addressed as an integral part of this knowledge engineering exercise. This will be discussed further in the sequel, and the structure of the ontology will reflect such pragmatic requirements which are not necessarily addressed by a first-principles description of the domain of brain tumours. This process will be detailed from the light of addressing the second requirement, namely the terminology used throughout the system based on the different standards relevant to the domain and the corresponding usage.

The third aspect of interfacing is about offering end users access to the processing and functionalities built into the software, whether as elementary as data retrieval or involving a range of diagnostic queries that experienced clinicians would want to target at the available data. It is inevitable that there could be several different requirements that different types of clinical users might have. For instance, the requirements of neurooncologists specialising in children's diseases seem to differ from those of adults in the nature of the details they require of a graphical interface to the information. Once again, the ability to manipulate content using concepts and descriptors from the relevant domains, independent of how the content might be rendered on screen is a requirement that feeds into the ontology design exercise. This last point will also fall under the third requirement by concentrating on the technological difficulties of migrating from existing notations for information and data to the new vocabulary part of the ontology.

1.2 Modelling Language

Several structured modelling languages (such as RDF, Topic Maps[2], Concept Maps[3]) have been considered in order to represent the HaDOM. The Web Ontology Language (OWL) is used for the reasons enumerated below. Please note that the choice of a modelling language has also been analysed from the viewpoint of user requirements: system functionality (F), clinical terminology compliance (T) and the translation from existing terminologies to the ontology(M).

[2]`http://www.topicmaps.org/`
[3]`http://cmap.ihmc.us/conceptmap.html`

1. OWL is XML compliant. Terms in HaDOM are to be transferred from one agent to another across the internet. An XML compliant language allows us to reuse existing parsers and interpreters.(F)
2. OWL is widely used and adopted as a W3C standard. It is expected that being accepted as an organisational standard would give OWL more advantages than other languages, including extensibility, continuity and technical support. For instance, a rule enhanced version of OWL, SWRL, is being developed and might prove useful when further extending HaDOM.(M)
3. OWL is expressive. OWL provides universal and existential quantifications to restrict terms in HaDOM. OWL-full also allows one to use enumerations – case-based aggregation of umbrella concepts. These constructs facilitate compositional definition of complex concepts. Furthermore, OWL provides support for declaring concepts disjoint, an expression useful for drawing distinctions between conceptual categories when the same name may be used to describe them in different contexts. This is particularly relevant when legacy database schemata are being mapped onto our ontology. (T)
4. OWL separates the so-called TBox containing mainly concepts and axioms from ABox consisting of instances. On the one hand, this separation also helps to maintain integrity of HaDOM. On the other hand, OWL-full allows defining concepts by directly referencing instances, effectively combining ABox and TBox. This is a necessity when enumerating possible status of patients or variants of a particular tumour type. (T)
5. OWL supports reasoning. Based on Description Logics (DL) (Baader et al., 2003), OWL provides automated classification with regard to defined concepts. At design time, such a capability helps to detect inconsistencies and modelling errors. Although the increased expressivity of the language normally results in high computational complexity of reasoning, logic-based inferences on HaDOM are normally carried out off-line and thus complexity is not an issue.(F)

1.3 Mapping Languages

The interface between HaDOM and legacy relational databases is currently implemented using D2RQ (Bizer and Seaborne, 2004). D2RQ aims to provide a bridge between relational databases and RDF graphs. Databases can then be manipulated using RDF toolkits such as Jena[4] and Sesame[5]. The current version of D2RQ only provides one way mapping, i.e. relational databases are considered read-only. A fragment of a typical D2RQ mapping script is shown in Fig 1.

In HA, the functionality of the federated architecture is driven by agents with well-defined tasks. Hence, the mapping languages are native to those agents which perform the mapping tasks. While this will be elaborated later on, we point out that the fragments in Fig 1 are examples of D2RQ scripts which are employed by DatabaseAgents to translate an RDQL query into an SQL query. This allows a term to be mapped between the RDQL references tables and SQL tables.

2 HA Domain Ontology

In this paper we follow the distinction between *domain ontology*, *upper ontology* and *application ontology* (Hu et al., 2007). While the first concentrates on modeling a specific domain of interest; the second focuses on common objects that are generally applicable across a wide range of domain ontologies. The third provides a core descriptive scaffold articulating the needs of an application on hand. This specificity requires the introduction of concepts that do not necessarily occur in upper ontologies, although they might encompass several domains of application. We focus on ontologies which aim to facilitate particular applications such as the HA system instead of general purpose ones, e.g. UMLS Metathesaurus[6] and MeSH[7]. Generally speaking, the HA domain

[4] `http://jena.sourceforge.net/`
[5] `http://www.openrdf.org/`
[6] `http://www.nlm.nih.gov/research/umls/`
[7] `http://www.nlm.nih.gov/mesh/meshhome.html`

```
# D2RQ Namespace
@prefix d2rq: <http://www.wiwiss.fu-berlin.de/suhl/bizer/D2RQ/0.1#>.
@prefix : <http://www.healthagents.net/hadv.owl#> .
#---------------------------------------------- # Database
db1:healthagents_db rdf:type d2rq:Database;
  d2rq:jdbcDSN "jdbc:mysql://localhost/healthagents_db";
  d2rq:jdbcDriver "com.mysql.jdbc.Driver";
# ---------------------------------------------- # Mapping
db1:mri_cm rdf:type d2rq:ClassMap ;
  d2rq:class :MRI_Series_Image;
  d2rq:uriPattern "http://www.healthagents.net/hadv.owl#mri_img_@@MRI.IDCASE@@_@@MRI.IDF@@";
  d2rq:dataStorage db1:healthagents_db.

db1:has_filename rdf:type d2rq:DatatypePropertyBridge;
  d2rq:property :has_file_name;
  d2rq:column "MRI.FILENAME";
  d2rq:belongsToClassMap db1:mri_cm;
  d2rq:datatype xsd:string.

db1:has_description rdf:type d2rq:DatatypePropertyBridge;
  d2rq:property :has_description;
  d2rq:column "MRI.DESCRIPTION";
  d2rq:belongsToClassMap db1:mri_cm;
  d2rq:datatype xsd:string.%
# ----------------------------------------------
```

Figure 1 D2RQ mapping fragment

ontology is used to determine *what* is in the domain of discourse of the HA system, e.g. patient records, types of tumours, parts of the brain, etc. Two of the main components using the domain ontology are:

- The ClassifierAgents which describe their inputs using metadata that corresponds to concepts defined in the ontology and output diagnostic class labels which are defined as subconcepts of Diagnosis and Histopathology, as histopathological descriptors of biopsied tissue are considered a gold standard for classification.
- The DatabaseAgents which retrieve data from (legacy) databases. DatabaseAgents populate the ontology using the retrieved data. Wherever a mismatch is identified between database fields and ontological concepts, a local mapping is used to resolve the discrepancy.

Communication between agents using the HA domain ontology requires the initiating agent to extract necessary terms from the domain ontology. This can be done in two ways: i) parsing the ontology *on request* and traversing the concept hierarchy to locate the right concepts or terms; and ii) extracting and reusing the concepts or terms *off-line*. The targeted agent needs to understand the meaning of the used terms (e.g. Astrocytoma or has_date) by consulting the ontology. This process is illustrated in Fig 2 showing that two agents, one of which is the *database agent*, communicate by referencing the domain ontology.

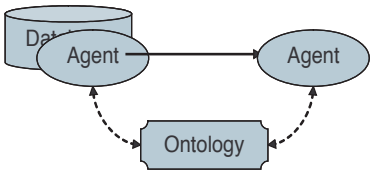


Figure 2 Communicating through domain ontology

*2.1 Purpose of Ha*DOM

Ontology, in the philosophical sense, is the study of *what* is – entities, and the relationships among the entities. Basically, it tries to answer questions such as "what exists or can exist in

the domain of discourse?" and "what are the relations between the objects in the domain of discourse?". This view of ontologies has been revised and modified by the knowledge engineering and artificial intelligence communities in order to fit better with the goals of knowledge acquisition and knowledge management. In knowledge management, an ontology is a compendium of organised terms and concepts that drive actions, and has a praxis-oriented structure. In knowledge acquisition of the domain specifics, they reflect the epistemological stance that the knowledge engineer takes up in completing this task. In either form, a standard definition – an explicit, consensual specification of a conceptualisation of a domain (Gruber, 1993) – provides a suitable working definition. In this paper, we do not commit to what an ontology is, but merely *how* it is that an ontology circumscribes what may be used in the HA system. Following Quine, we state that "to be is to be the value of a bound variable," (Quine, 1953) but the variable is very much a part of the symbolic order of the software system, and despite its declarative formal foundation, this representation is given meaning in use. Glimpses of this approach show up in mapping issues discussed in the paper, in retrieving answers to queries (binding variables in quantifiers) and so on. As such, in this project we eschew refinements and extensions of upper level ontologies in favour of a more pragmatic approach of ensuring that the declarative framework met the requirements of an application domain, and the validation that we sought was framed in that context. Indeed, the adequacy of the representation scheme, its fidelity to the relevant parts of (say) the neurooncological domain as conceived by its practitioners, rests upon the interpretations it supports and promotes in the context of clinical practice.

The HaDOM follows this more opportunistic approach to defining "things" in HA. In other words, HaDOM captures the expertise and information necessary to facilitate diagnosis and prognosis of different types of brain tumours and management issues of brain tumour patients. Such knowledge is elicited and formalised in a machine-processable manner and with explicit definitions, providing the ground on which consensus can be described and verified. This is particularly important for a distributed environment such as the one envisioned by HA, since it is not rare in such environments for a meaningful conclusion to be drawn upon suggestions and observations by experts with different background knowledge and using different terminology.

While the inclusion of clinical practitioners in system usage serves to ratify the faithfulness of domain representation, its use amongst software agents to facilitate interoperability requires stringent regimentation. Software agents and human users share the load of pattern recognition and diagnosis encoded. Hence the knowledge representation scheme needs to be both expressive and sound, (*cf.* the above discussion on OWL). When we give instructions to software agents and when software agents communicate with each other, HaDOM specifies the terms of reference in the language spoken by all participants for conveying the intended messages. Examples of such conversations are "retrieve cases of all patients under age 5" and "fetch a case of glioma from Hospital A" where underlined words are concepts from HaDOM.

A domain ontology, however, is not sufficient for establishing consensus among software agents. HaDOM defines *what* software agents talk to each other about, but not *how* they talk — how the messages are composed, what speech acts are accommodated and so on. This is beyond the scope of a domain ontology. In HA, a separate ontology defines the concrete means for passing the information encoded with HaDOM. This communication language (HAL) defines the format of different types of messages that are sent back and forth among agents, parameters that are necessary to reconstruct agent behaviour, the encoding and decoding methods for extracting information from such messages and house-keeping information with respect to messages. For instance, a classifier agent might submit an instance of Database_Request_Msg to a database handling agent to "retrieve a validated case with feature X, Y, and Z". How the message itself is interpreted is regulated by HAL while the actual content —"retrieve a validated case" with the specified features—would be composed using instances of HaDOM.

In practice, each agent is equipped with an ontology parser to understand the domain ontology. Upon receiving a request, the agent first consults HaDOM for the meaning of different terms

appearing in the request. It then carries out the tasks that it is instructed to perform, e.g. retrieving data from a database, classifying data against a set of labels, etc. When it finishes, the agent composes an answer/response to the request using again concepts defined in HaDOM.

HaDOM benefits from the reasoning capabilities inherent in the selected knowledge representation and reasoning formalisms. A DL based formalism provides automated subsumption-based inference, *eg.*, Melanocytic Tumour is a subcategory of Meningeal Tumour. Hence, an instance of the former type would automatically inherit the characteristics and constraints of Meningeal Tumour. HaDOM, however, does not contain knowledge of problem solving methods. That is to say, the domain ontology captures only the static model rather than the inference procedures. Typical examples of the former are "patient", "a particular type of tumour", "MRS scans with their parameters", etc. while examples of the latter are "due to the fact that ... the tumour is malignant" or when referring to MR spectra, "all peak areas with ... characters suggest ...". Such separation is based on both theoretical and practical considerations. On the one hand, such inferences are built using hand-crafted rules, machine learning techniques, etc. which, currently, are not ready to be built into a declarative knowledge representation formalism. On the other hand, a medical diagnosis is typically a complicated process with ambiguity and uncertainty for which a framework of logical inference that is streamlined for taxonomic knowledge is hardly adequate (Rector, 1999). This, however, does not preclude building a reasoning system on top of HaDOM; indeed the classification tasks within HA exemplify the use of non-deductive reasoning while being grounded in terms for which a deductive, declarative formalism has been created. Other reasoning mechanisms, based on ontological concepts could be used to switch between different classification protocols. For instance, if certain patterns are present in a patient's MRI and/or MRS scans, an inference may be made to suggest the use of pattern-specific classifiers and even exclude certain possibilities from the final diagnosis if they are eliminated by clinical knowledge or perhaps an oncologist's understanding of the nature of the biochemical pathways involved. Such reasoning systems should rely on HaDOM to express the underlying knowledge model and be developed in close collaboration with clinical specialists, a task we have made preliminary investigations into, but have not integrated into the current implementation.

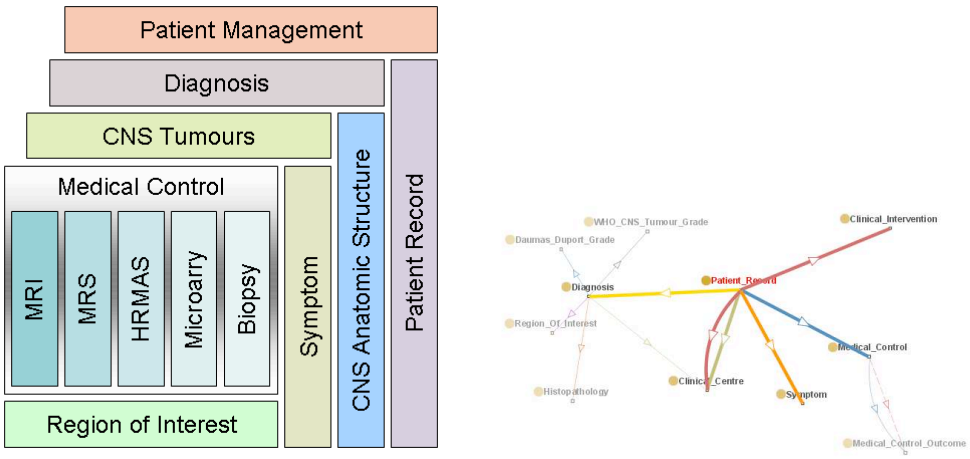
*2.2 Structure of Ha*DOM

HaDOM defines information related to brain tumours (and tumours affecting the central nervous system), e.g. brain tumour diagnosis, prognosis, patient management, etc., in the context of the HA project. The primary goal of HaDOM is to address the functionalities that are envisioned in HA and drive such functionalities smoothly. The HA project has been strongly influenced by several other projects, namely, INTERPRET[8] and eTUMOUR[9]. The impact of these two projects on HA is reflected in the legacy terms in HaDOM that facilitate a smooth migration of INTERPRET data into HA databases.

HaDOM comprises several relatively independent modules, each focusing on a particular aspect of diagnosing brain tumours. Fig 3(a) indicates the dependability among different modules. For instance, *Medical Control* consists of five medical imaging modules; *Diagnosis* relies on anatomic information, CNS Tumour types (based on WHO CNS tumour classification), symptoms and results from medical controls. Some top-level concepts of HaDOM are listed here and certain concepts will be detailed in the following sections. When defining the conceptual structure and concept names of HaDOM, we worked very closely (ontology validation meetings every 3 months throughout the whole duration of the project, joint demo programming workshops every 6 months etc.) with domain experts such as neurosurgeons, biochemists, and oncologists, to build up the picture of how a person is first recommended to the hospital, how he/she goes through all the medical exams and tests, how knowledge from different domains are projected upon this patient,

[8] `http://azizu.uab.es/INTERPRET`
[9] `http://www.etumour.net/`

and how a patient is managed during his/her treatment. The multiple domains of expertise – from neurooncology, medical imaging and spectroscopy, gene expression, and so on – address different levels and scales, both temporally and spatially. As such, tying together the knowledge modelling of each of the domains into a single domain ontology would require bridging relations which are placeholders for as yet unspecified scientifically validated explanations. In this context, patient identity offers a conceptual handle as a site for knowledge integration, wherein these multiple discourses bear meaning in the context of aetiology and progression of a type of disease (Mol, 2003). Thus a meta-level organising structure for the ontology can be viewed as a star-shaped graph with Patient_Record (Fig 3(b)) at the centre linking together all the related information about the patient coming from different domains of specialism. Again, such a choice for knowledge modelling is influenced by the nature of clinical practice, rather than a description of knowledge about cells and tissues from physiological and spatio-anatomical perspectives. The latter, physical reality of biomedicine might have had a closer fit to the sense of ontology as a study of "what exists," as the underlying, causal organisers of medical intervention and management protocols.



(a) Dependency of clinical concepts in HaDOM

(b) Neighbours of Patient_Record

Figure 3 HaDOM conceptual structures

- Patient_Record also known as electronic health record (EHR) is the most important concept in HaDOM. It acts as an entry point into the ontology, specially when the task of rendering information onto graphical user interfaces is undertaken. It connects a particular patient to his/her examinations, diagnosis, treatment, prognosis, etc.
- Patient is introduced to establish links between current EHR with chronological or historical records. It holds the necessary information regarding a patient needed for the purpose of diagnosis, treatment, and patient management. It can also facilitate the anonymisation process by creating a unique URI for a patient without exposing his/her identity.
- Symptom is defined with attributes has_date and has_description. An instance of Symptom should be referred to by instances of Patient_Record when symptoms of a patient should be recorded.
- Clinical_Centre is referred to by instance of Patient_Record. Information regarding clinical centres becomes necessary when the origin of medical examination data should be recorded.
- Clinical_Intervention is introduced to be compliant with INTERPRET and/or eTUMOUR database schemata. Clinical_Intervention is the parent concept of various methods used to treat patients with a diagnosed tumour. Sub-concepts of Clinical_Intervention include Therapy

which in turn has Chemo_Therapy and Radio_Therapy as sub-concepts, Adjuvant_Method that might aid tumour treatment, and Surgical_Removal as the surgical removal of cancerous tissue.

- Medical_Agent is an umbrella term for substances used in examination and treatment. For instance, in HaDOM, the treatment of brain tumour requires Anaesthetic_Agent, Anti_Convulsant and Steroid which are sub-concepts of Medical_Agent; the MR imaging model might require injection of contrast enhancing substances. The medical/biochemical agent used in a particular treatment will be introduced as instances of Medical_Agent or one of its sub-concepts, with names and the administered dosage documented.
- Medical_Control is the parent concept of all the medical investigation modules including Biopsy, HRMAS, Magnetic_Resonance, and Microarray. Among such different modules, Magnetic_Resonance has child concepts MRI and MRS and Biopsy has child concept Stereotactic_Biopsy.
- Medical_Control_Outcome records all the information produced by and interpreted from a Medical_Control module.
- Region_Of_Interest is a non-clinical concept. It is the area in or related to a patient's central nervous system that arouses clinician's concerns. It is normally instantiated as a mass, enhancement, or highlighted area in medical images or as a tissue to be examined *ex vivo*.
- CNS_Anatomic_Structure describes the major organs and parts of organs related to the human brain. We use separate concepts for the functional aspect (e.g. Brain_Stem) of a particular organ and its structural aspect (e.g. Brain_Stem_Structure). A few properties are introduced to describe the spatial relationships, e.g. spatial_connected_to and spatial_within.
- Diagnosis refers to terms in WHO CNS Tumour classification. An instance of Diagnosis is reported in a Patient_Record and is associated with a particular instance of Region_Of_Interest as an instantiated relation of the anatomical structure in the ontology.
- CNS_Tumours is the WHO classification of Tumours affecting Central Nervous System. The hierarchical structure of WHO classification is faithfully re-constructed in HaDOM. Further extension and modification will be made compliant with WHO classifications. Indeed, we have both the 2002 and 2007 classification indices in the ontology.

The above categories are the top level concepts that are defined as the direct sub-concepts of the root concept, $\top$ (e.g. $\langle$`owl:Thing`$\rangle$). Note that several categories are introduced in order to accommodate legacy terms and concepts from existing databases schema, such as the INTERPRET databases.

2.2.1 Patient record

Instances of patient record should be regarded as the point of reference of a system that uses HaDOM (as shown in Fig 5). Normally, when a new patient P is admitted or reported, a new instance of Patient_Record is created, which includes a reference to an instance of Patient concept to record personal information of P. Instances of Symptom are created to describe the complaints of P. Instances of Medical_Control are introduced including those of different imaging modules so as to document information regarding the individual examinations that P has undertaken. Instances of Diagnosis are used to note down diagnostic details while instances of Clinical_Intervention serve to keep tracks of treatments and surgeries.

In order to retrieve information of a particular patient, instances of Patient_Record again serves as the main entry point. For instance, assume that one wants to find all the patients who have astrocytic tumour. He/she "glues" instances of different concepts together using a Patient_Record instance as in the following query.

```
SELECT ?patient WHERE
    (?pr, hadv:record_of, ?patient) AND
    (?pr, hadv:diagnosis, ?diag) AND
    (?diag, hadv:is_who_class, ?tumour) AND
    (?tumour, rdf:type, Astrocytic_Tumour)
```

If a particular visit of patient P is identified by the URI x, the clinical history of P is accessed using the following pseudo-RDQL query.

```
SELECT ?patient_record WHERE
    (?patient_record,  hadv:record_of,  ?x)
```

2.3 Medical control and relevant concepts

A number of technologies are employed in brain tumour diagnosis. In HaDOM, we enumerate four approaches, namely *Biopsy*, *HRMAS*, *Magnetic_Resonance*, and *Microarray*. We defined them as sub-concepts of Medical_Control with properties that link necessary information, e.g. has_date property keeps a time-stamp on every medical examination.

Among the four approaches, Magnetic_Resonance is the main research focus of HA project. It has MRI (for Magnetic Resonance Imaging) and MRS (for Magnetic Resonance Spectroscopy) as sub-concepts.

A medical control instance produces outcomes that are defined as Medical_Control_Outcome including Medical_Image, Textual_Report, and data concepts for each of different test modules (See Fig 4). Medical_Image covers the results of high throughput MRS and MRI. In practice, MRI outputs a stack of images taken at fixed intervals. This is reflected in the ontology as MRI_Image connecting to MRI_Image_Sequence through *part-whole* relationship. Textual_Report refers to a paper or the electronic reports generated by clinicians. It might contain the conclusions and descriptions on a set of images taken with respect to a particular patient.

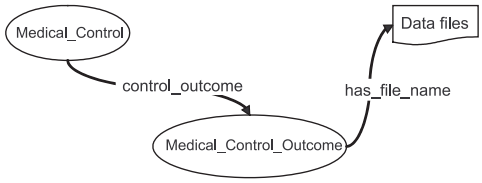


Figure 4 Medical_Control, Medical_Control_Outcome, and the actual data files

The actual MR data might be in two forms: processed data and raw data. In order to trace diagnosis, it is necessary to have both forms of data available and linkable from a particular patient record. The actual data file and relevant information are kept as instances of MR_Data:

$$\begin{aligned}\mathsf{MR\_Data} \doteq\ & \ldots \sqcap \forall\, \mathsf{has\_description.String} \sqcap \forall\, \mathsf{has\_file\_name.String} \sqcap \\ & \forall\, \mathsf{has\_creation\_date.String} \sqcap \forall\, \mathsf{has\_creation\_id.String} \sqcap \ldots\end{aligned}$$

3 Structuring HaDOM to fit praxis

A declarative knowledge representation is an enabler of separation of knowledge from particular models of its use. However, streamlining the ontology for efficient use in the context of a particular application such as HA must be balanced against the need to have the ontology serve as a vehicle for knowledge sharing independent of it. The ontology developed reflects these contrary pulls, and we address such ontological features in this section.

*3.1 Modularising Ha*DOM

Many technologies and methods used to detect and diagnose brain tumours are yet to reach a mature stage. This is made explicit in the fact that in 2007, halfway through the HA project,

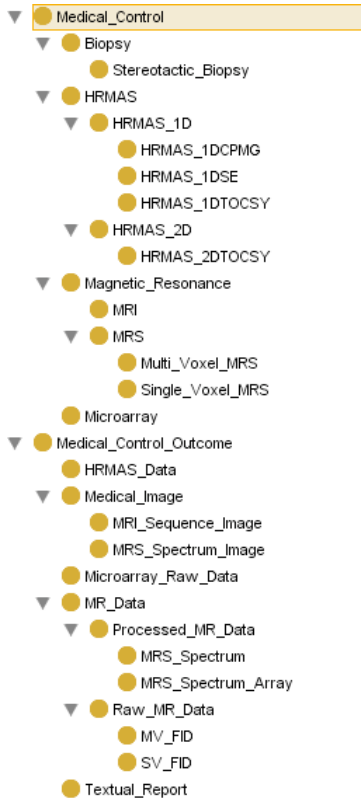


Figure 5 Medical_Control, Medical_Control_Outcome and their sub-concepts

WHO released a new classification of tumours affecting Central Nervous System with major changes to the terminology as well as the taxonomies (`http://www.who.int/en/`). In light of further changes being highly likely, we revise HaDOM into a modular structure that confines changes locally to a module. HaDOM has a core kernel containing the essential concepts from the domain and the top level conceptual relations between these concepts, and five modules below:

- `haMedCrtl.owl` extends the core with concepts detailing the various medical examination methods, the results generated, and materials used in such examinations.
- `haClassifier.owl` enhances the core with knowledge prescribing how the input and output of automated classification methods should be constrained in the HA framework;
- `haSecurity.owl` introduces a layer of system security-specific concepts and properties. When designing a health care system, one needs to not only accommodate the needs for clinical use but also observe patient privacy and safety issues, especially in a distributed environment as envisaged by the HA network. Reflected in HaDOM is a dedicated module for security concerns. We exercise a policy rule based security model regulating the access rights of HA users. Basic concepts to facilitate such an approach are
  - Access_Right regulating who can manipulate the data and how;
  - User which is further divided into Software Agent, Clinician, Patient, and System_Admin;
  - Resource as the data and methods available to users of HA system.
- `CNS_Anatomic_Structure.owl` details the anatomic structure of the human brain and central nervous system.
- `CNS_Tumour.owl` gathers tumour types with or without histopathology results. Paediatric_Non_Histo is the parent concept for tumours. This concept is not necessarily required

under histopathological studies. On the other hand, the WHO 2002 classification and the WHO 2007 classification co-exist under CNS_Tumour_Histopathology. With the help of clinical experts, we mapped the 2007 classification against the 2002 one and marked tumour types from 2007 with *deletion*, *creation*, *split*, *merge*, *generalisation*, and *specialisation*, similar to the types of changes proposed in (Noy and Musen, 2004). With such a markup, we can establish correspondences between different WHO classifications and easily "revive" legacy patient records dating back to 1950s[10]. Note that a change between the two classifications may lead to marking a concept with different actions. For instance, the revision on Choroid_Plexus_Carcinoma suggest *creation* of "Neuroepithelial tissue tumours" in 2007, *deletion* of "Choroid plexus tumours" in 2002, *specialisation* of "Choroid plexus tumours" under "Neuroepithelial tissue tumours" in 2007 and *deletion/creation* of all the sub-type of "Choroid plexus tumours" including "Choroid plexus carcinoma".

Moreover, we refine tumour types with tumour grading systems. Two different grading schemata are introduced in HaDOM: the *Dauma Duport* grading system and the *WHO* grade as instances of Dauma_Duport_Grade and WHO_CNS_Tumour_Grade respectively who are in turn sub-concepts of CNS_Tumour_Grading.

3.2 Modelling Anatomical Structure of Central Nervous System

Representing anatomic knowledge has been extensively studied topic and many different approaches have been proposed including the comprehensive Foundational Model of Anatomy (FMA) (Rosse and Mejino, 2003) for humans. In HaDOM, anatomical knowledge is used to establish the connection between diseases and human organs and we focus on anatomical knowledge of the central nervous system only. When constructing an ontology, establishing connections with existing ontologies such as FMA is recommended so as to maintain consistency with regard to the latest advances in the domain of discourse. This is not strictly applicable in HaDOM for the following reasons. In order to connect diseases and human organs, *part-whole* relationships are necessary to infer potential damage to other neighbouring areas and to the neural functions that the entire region of the brain presents, of which the tissues under consideration is only a part. FMA uses a separate spatial ontology and models the partonomy information in the Anatomical Structural Abstraction model. Specialist reasoning systems other than a DL-based one are needed; this makes the implementation unnecessarily complicated.

Using (Damasio, 1995) as a reference, we sought a balance between a knowledge model containing an exhaustive and refined coverage of human CNS and a parsimonious construction that is sufficient for the HA framework. The fine line between domain and application ontology is identified with the help of clinical experts working closely with the HA development team. The criteria for opting to place a part of CNS in or out of the anatomical model is whether it is mentioned in the patient's EHR, whether its neighbouring parts are referred to in patient's EHR, and whether its subparts are used in the patient's EHR. Using T_{EHR} as the set of anatomical terms that appear in patient EHRs, this choice criterion is formalised thus:

$$\{\mathsf{C} \mid \mathsf{C} \in T_{\mathrm{EHR}} \vee (\exists\, \mathsf{D} \in T_{\mathrm{EHR}} \wedge \mathrm{adjacent}(\mathsf{C}, \mathsf{D})\,) \vee (\exists\, \mathsf{D} \in T_{\mathrm{EHR}} \wedge \mathrm{partof}(\mathsf{D}, \mathsf{C})\,)\}$$

We refine adjacent(x, y) to be spatially *left_to*, *right_to*, *beneath*, *above*, *connected_to*, *inner*, *outer*, *restriction_surround*, etc. A brain tumour might damage brain tissue which inevitably affects the corresponding neurological functions. In HaDOM, a series of neurological functions are defined as instances of concept Nerve_Function and are associated with brain anatomical structure using property has_function. By doing so, one is then able to infer potential damage to normal muscle movements and senses based on the location of the brain tumour and other tumours of the CNS, and thus cross check with a patient's observed symptoms.

[10]Available from UK West Midland Brain Tumour Registration.

In HaDOM, we adopt the approach to modelling part-whole relationships in (Hahn et al., 1999) using only the subsumption relationship *is-a* inherent in DLs. Partonomy is emulated with *is-a* hierarchies of concepts introduced particularly for representing structural knowledge. CNS is viewed as a series of three coexisting concepts: the structural concepts which end normally with "_Str", the anatomical concepts themselves and the part concepts which are normally with suffix "_Prt". For instance, brain stem is defined by the combination of Brain_Stem_Str, Brain_Stem and Brain_Stem_Prt with two subsumption relationships, i.e. Brain_Stem $\sqsubseteq$ Brain_Stem_Str and Brain_Stem_Prt $\sqsubseteq$ Brain_Stem_Str. Among the triadic combination, Brain_Stem is the one holding all the taxonomical knowledge while Brain_Stem_Str and Brain_Stem_Prt are the bridge to establish partonomical chain of anatomical structures.

Based on this triadic combination, the left cerebral hemisphere is defined as

$$\begin{aligned} \text{Left\_Cerebral\_Hemisphere} &\sqsubseteq \text{Left\_Cerebral\_Hemisphere\_Str} \sqcap \ldots \\ \text{Left\_Cerebral\_Hemisphere\_Prt} &\sqsubseteq \text{Left\_Cerebral\_Hemisphere\_Str} \\ \text{Left\_Cerebral\_Hemisphere\_Str} &\sqsubseteq \text{Cerebrum\_Prt} \\ \text{Cerebrum\_Prt} &\sqsubseteq \text{Cerebrum\_Str} \end{aligned}$$

Hence, if a tumour is identified within Left_Cerebral_Hemisphere, we can safely infer along the partonomical chain that it is also within Cerebrum structure and thus is part of Main_Brain structure. For simplicity, we define anatomy specific knowledge belong to the entire structure at "xxx_Str" and use the anatomical concepts to usher in references to conventional anatomy terminology.

*3.3 DICOM'ising Ha*DOM

The Digital Imaging and Communications in Medicine (DICOM)[11] standard was initiated by the American National Electrical Manufacturers Association (NEMA)[12] to regulate the distribution and viewing of medical images, and later become a global standard adopted by clinical authorities and manufacturers from major European and North America countries. DICOM has become an increasingly common format for receiving scans from hospitals. Therefore, even though DICOM descriptors are tied closely to implementation details, i.e. how image files are composed, stored, transferred, etc. rather than at the conceptual structure of the domain of discourse, we enrich HaDOM with a DICOM reference module to enable smooth migration to DICOM compatible system.

Among DICOM standards, the Image Information Object Definitions (IOD) impinge on HaDOM. IOD impose a standard format when transferring medical images. Depending on the purposes of medical studies and the nature of associated data, IOD differentiate `Patient Module` for patient data, `Series Module` for information related to particular imaging modules, `Study Module` for information about the entire medical study, etc. Correspondences are manually crafted to facilitate inspecting HaDOM concepts in a DICOM apparatus. More specifically, `Patient Module` in DICOM perfectly matches the Patient concept from HaDOM with nearly one-to-one correspondence between DICOM and HaDOM properties. `Study Module` is translated into Patient_Record in that HaDOM's patient record comprises all the information concerning a patient on a particular disease from the first visit that he/she made to one of HA member hospitals until the end of his/her treatment. `Series Module` stays one level below `Study Module` and is mapped to Case_Record including information of a particular visit of a patient. Image Module details how images are taken. Depending the image types, Image Information is saved in respective sub-concepts of Medical_Control and Medical_Control_Outcome.

[11] `http://medical.nema.org/`
[12] `http://www.nema.org/`

4 Facilitating Data Interoperability with HaDOM

In the HA framework, a domain ontology is the locus of reference for participating agents and centres to align their local vocabularies. Hospitals joining the HA network can either adopt the ontology-derived database schema provided, or they can retain their local database schemata and data gathering processes based on such schemata. In the latter case, a mapping between these local databases and HaDOM is needed to enable communication between hospitals and the HA system. This, in turn, will allow information to be read in from local hospital databases to the HA system and be compliant with HaDOM, and thus feed into the goal of building and refining classifiers.

The mapping between database schemata and ontologies, at present, cannot be automatically generated. Instead, a manual or semi-automatic method will be performed during the installation process of HA software. In order to help create such mapping, a user friendly interface has been developed. It should be noted that the mapping requires a person with considerable database and clinical knowledge if the installation is to be successful. We have undertaken such a task for database schemata used in hospitals in Spain and the UK and have created a tool that facilitates this mapping procedure. The successful deployment of the system across these heterogeneous networks is validation of the ontology and mapping tools' representational adequacy.

*4.1 Communicating with Ha*DOM

HaDOM is used as the common reference point among different clinical centres/hospitals which maintain their own vocabularies and database schemata. As illustrated in Fig 7, such a design seeks to respect the integrity and independence of legacy databases. The discrepancy between such schemata is, however, resolved by dedicated interfaces between each individual schema and the common domain ontology HaDOM.

A typical scenario of using HaDOM starts with the visualisation of a particular patient record read from the local database. The visualisation is controlled by the ontology. Information read from local database is translated into a format compliant with HaDOM via a relational database to RDF interface to create instances of HaDOM. Such instances are then classified and displayed at the allocated sections in the HA graphical user interface (GUI).

Agents of different types are equipped with parsers understanding HaDOM which ensure HaDOM-compliant communication between them (see Fig 7). For instance, when querying a classifier, the handling agent would submit queries composed using terms drawn from HaDOM. For instance, the RDQL query illustrated in Fig 6 retrieves all the patients that are diagnosed by hospital "BCH".

```
SELECT ?patient %
WHERE(?record,<http://healthagents.net/hadv.owl#patient_number>,?patient),
   (?record,<http://healthagents.net/hadv.owl#diagnostic_centre>,?centre),
   (?centre,<http://healthagents.net/hadv.owl#has_name>,"BCH")
```

Figure 6 RDQL query example

*4.1.1 Mapping between Ha*DOM *and database schemata*

HaDOM provides a common reference point that local vocabularies and legacy database schemata can exploit to achieve data interoperability within HA. Mapping between ontologies and database schemata, however, is not an easy task. Although extensive research has been done (Kalfoglou et al., 2005), the problem is far from solved. Apart from the general issues associated with independently developed knowledge models, a major obstacle lies in the fact that the conceptual structure of ontologies and database schemata are significantly different. Ontologies tend to see

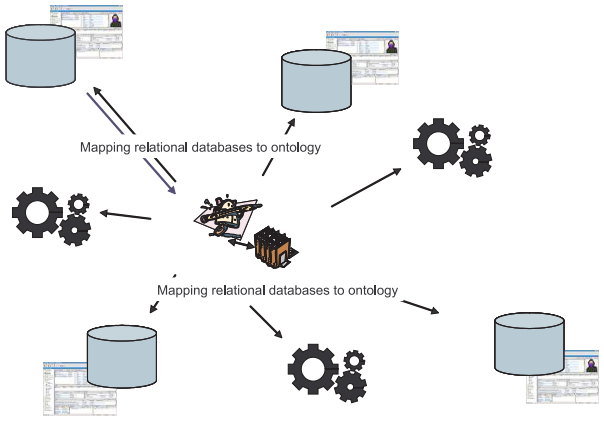


Figure 7 HaDOM to facilitate inter agent communications

the world through hierarchically layered abstractions that supervene on a set of instances, while database schemata work better in a world with vertical partitions that accumulate informative and discriminatory features (columns for attributes of an entity). Automated methods thus far are still not comparable with human data curators crafting mappings manually by unpacking the respective knowledge coding patterns. In the scope of HA, we evaluated several existing mapping algorithms to automatically identify correspondences between HaDOM concepts and database schemata currently installed in HA member hospitals. The results are summarised as follows.

- String (edit) distance algorithms (*c.f.* those discussed by Cohen et al. (2003)) gave the best results. An obvious reason is that the domain of discourse of HA is fairly small with well studied and well documented knowledge. It is expected that similar names are used for both concepts and database tables/columns. String distance methods, however, failed to handle acronyms, synonyms and names in different natural languages where the later, though not common in HA domain, might become more evident when HA framework is deployed widely to involve legacy databases from different countries. For instance, when processing data from existing databases, patient's gender may be "hombre" in Spanish, "männlich" in German, or "male" in English, all of which bear limited resemblance.
- Although algorithms based on WordNet solve the synonymy problem, they fail to achieve much better results on acronyms and terms from multiple natural languages.
- Structure-based and many other so-called semantics-enhanced matching algorithms (Rahm and Bernstein, 2001)(Kalfoglou et al., 2005) are not applicable. Such algorithms perform deep structure comparison between the source and the target knowledge models. However, HaDOM concepts and database tables might be conceptually different and thus do not provide many hints for structure based matching.

The above limitations/weaknesses rule out automated mapping methods in establishing connections between ontology and local database schemata. Manual mapping becomes inevitable.

4.1.2 Making mapping easier

Data interoperability has been studied by both conventional database community and the new semantic web community (Benslimane et al., 2007)(Bussler et al., 2005). It is our contention that although many algorithms have been proposed and implemented, data interoperability between ontologies and databases is far from satisfactorily addressed. Before a mature automated mechanism can be found, mapping between ontology and database schemata is still a human labour intensive task with heavy involvement of domain experts. Manually crafting mappings is

not straightforward either. Concepts in HaDOM can be mapped to tables in databases, columns from a particular table, or columns from several tables. Similarly, although concept properties are frequently in one-to-one correspondence with table columns, they can also take values from several columns across tables or be merged into single columns. While HaDOM constrains domain vocabulary, its effective use requires instantiating its concepts with entries extracted from databases. Unstructured database schemata makes translating values in table cells difficult. In practice, we cannot presume values in table cells are always predictable. When defining the database schema, if one enumerates all the possible values for table columns (e.g. Patient.Gender = {"Male", "Female", "M", "F"}), only a handful possible values need to be coded in the mapping scripts in the ideal situation. If, however, one does not enumerate the values but rather constrains the values as any string of length 4—a common practice in hospitals, cells can take up any arbitrary strings. Such a scenario becomes more likely when the HA framework is widely deployed and takes in legacy databases from new members joining the network.

In order to simplify the mapping process, we restrict ourselves to map concepts in the ontology only to tables or parts of tables and properties to concatenations of table columns. We observe the independence of tables to avoid using many database join operations which have a significant impact on the efficiency of database querying. This design principle was reinforced by introducing a housekeeping property to every concept. This property gathers all the information unable to map to any ontological entities as a string separated by "+". For instance, when mapping databases from Birmingham Children Hospital (BCH), Patient is extended as

$$\mathsf{Patient} = \ldots \sqcap \exists \mathsf{has\_id.String} \sqcap \exists \mathsf{has\_name.String} \sqcap \exists \mathsf{gender.String} \sqcap \exists \mathsf{concept\_identifier.String} \sqcap \ldots$$

where concept_identifier gathers the information that is unique to each hospital. In the case of BCH, it has the following D2RQ code "BCH@ + @@PATIENT_TBL.P_EU_ID@@ + . . ." to collect information useful only to BCH.

When addressing the discrepancies introduced by ambiguous database schema specification, we use Jena ARQ[13] to keep mapping scripts less database dependent. A Java property file stores all the locale information and is continuously extended once new values are identified. Human data curators, normally the persons maintaining databases, need to keep the property file up-to-date.

4.1.3 The HA OntoDB Mapping Toolkit

The HA Mapping Toolkit is a software application developed for mapping between a HA legacy database and the HaDOM. Motivated by the idea of automating the mapping process between an ontology (concepts and properties) and a relational database schema, we designed the toolkit with a "drag and drop" feature to facilitate ease of use. This toolkit allows the user to relate concepts in a given ontology to entities present within a relational database with the final goal of obtaining a mapping script, using the D2RQ language for its representation.

The D2RQ framework contains a mapping language for treating non-RDF relational databases as virtual RDF graphs, and a platform that enables applications to access these graphs through Jena and Sesame APIs, as well as over the Web via the SPARQL protocol. The full specification of D2RQ language as well as its platform are available from http://www4.wiwiss.fu-berlin.de/bizer/d2rq/spec/. The generated mapping scripts allow any given user, who does not need to know the organisational schema of a database, to query the database via the SPARQL language powered by an ontology.

In the following, we present the functionality of the toolkit and its usage in a practical setting. This allows us to illustrate the approach taken for its design, and also conveys the full extent of its capabilities. In doing this, we start by roughly describing a typical workflow of a user working with the application. A user is presented with the option of loading an OWL ontology visualised through the built-in interface. At the same time, the user can load a relational database schema specified either by an XML file or by access to the location of the actual database server. The database schema is visually available through the interface provided by the application. Once

[13] `http://jena.sourceforge.net/ARQ/`

the ontology specification and the database schema are loaded, the workflow execution begins by presenting the user with a directed graph that shows the entities (nodes) within the database and the relationships (*e.g.* foreign keys) amongst them. Apart from this, a series of windows appear which are used to specify the concepts and entities to be associated by dragging graphical renderings of related items and dropping them into a common space to articulate their association. Figure 8 shows a screenshot of the application with an ontology and a database loaded. At the centre of the window, the graph representing the database schema is displayed.

In order to improve usability, the workspace of the mapping toolkit is divided into four different areas used to present the different type of information involved in the mapping process:

- **The ontology area** shows, in two windows, the concepts available within the ontology, and the properties of the currently selected concept (marked 1 in Fig 8);
- **The visualisation area**, apart from the graph mentioned above, presents two more tabs, one displaying the D2RQ file being generated by the mapping process and the other, a table presenting the data available on the database for the selected entities over the schema (marked 2 in Fig 8);
- **The database area** shows the schema of the specified database (tables and their fields); also makes available a window with suggestions of database schema tables as the mapping candidates of the currently selected ontology concept (marked 3 in Fig 8);
- **The mapping area** displays the D2RQ specifications and the way of presenting the information of the mapping. In this space all the D2RQ specifications can be filled in to obtain a complete mapping description. This area also fosters two subspaces: one on the side of the ontology (ontological concepts and properties), and the other on the side of the database (database tables and fields, that are being related within the current mapping description) (marked 4 in Fig 8).

Given the intuitive character of the application's interface, the only thing to be done in order to relate a concept in the ontology (or any of its attributes) to an entity within the database is to select the desired object, drag it to the correct window within the mapping area, and do the same for the corresponding entities in the database. Figure 8 shows how easily the mapping process is carried out, with a few concepts already mapped and an ontology concept being dragged to the mapping area to relate it with its counterpart within the database.

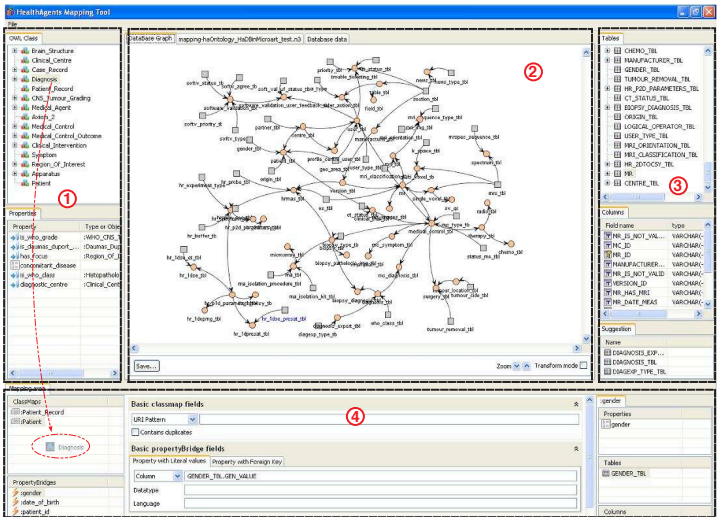


Figure 8 GUI of HA Mapping Toolkit

In order to complete the mapping script the user must repeat the drag-and-drop action for each concept from the ontology that needs to be mapped. At any time during the development of the mapping, the user is allowed to visualise the current mapping script file. Another useful functionality is the possibility of querying the database, at any time, using the tab window provided for that specific purpose. In doing this we illustrate a statement, articulated in the introduction, that an ontology provides a conceptual integration of the range of questions one would expect to have answered about a domain. As such, the meanings of terms get explicated with reference to the answers obtained from queries that involve these concepts. This helps the user to decide, based on the information stored in the database, to which concepts and given entities from the database schema is related. At the end of the mapping process the user has, within a mapping script file, all the mapped concepts and the correspondence with the original database entities for direct editing (Figure 9 shows an example of the resultant mapping script file).

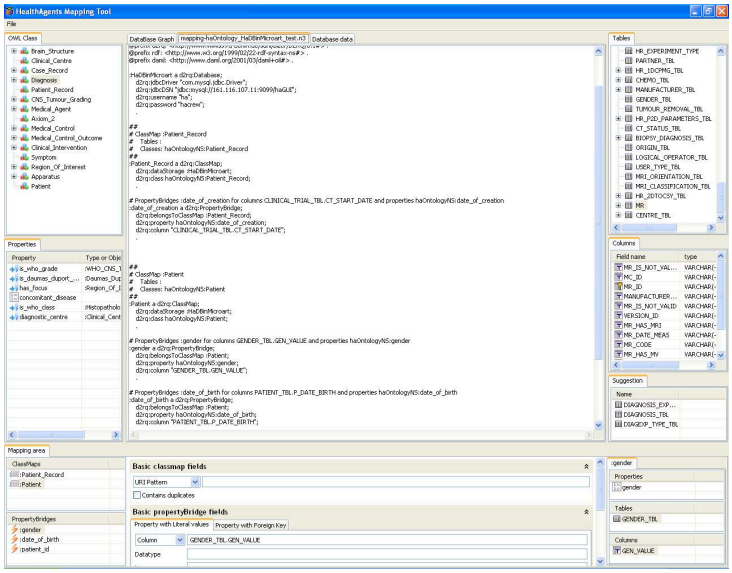


Figure 9 D2RQ file containing the mapping description.

4.1.4 Accessing data functionality

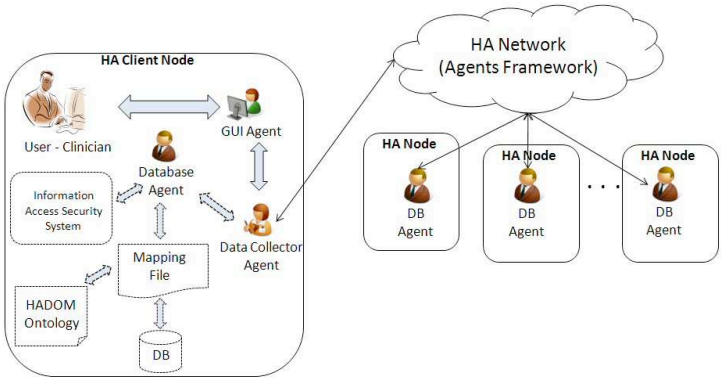


Figure 10 Information flow between a client node and the HA network when retrieving medical cases.

The mapping of a database into HaDOM is performed by the Database (DB) Agent using the mapping script containing correspondences between the database schema and the ontology. The

DB Agent is also responsible for checking permissions before the secure delivery of the requested information. This process is made through the validation of the user who sends the SPARQL petition to the DB Agent via a GUI Agent. Each user has an unique ID which is obtained and maintained by the GUI Agent and attached to all the messages containing SPARQL queries sent to the DB Agent. If the user has enough privileges to see the medical cases retrieved by the DB Agent of a particular clinical center, and the requested cases are marked as *public* then the results are passed to the GUI Agent who shows them to the user. Fig 10 illustrates the described flow of information between a HA client node and all the other client nodes in the network. Further details of the security arrangements are provided in a companion paper in this issue.

Each HA node owner of a database with information to share must also have its own DB Agent with the corresponding mapping file. Nevertheless, any HA node can also join to the HA network even if the node does not have its own database (the dotted components within the client node are optional as is shown in Fig 10) but may want to retrieve information from the databases of other medical institutions.

In order to execute the HA DSS at a clinical node, the HA Framework needs to be running and the Data Collector Agent invoked. If this initial requirement is satisfied, the DSS can be started and the first screen is used by the user to log on into the system (Fig 11).

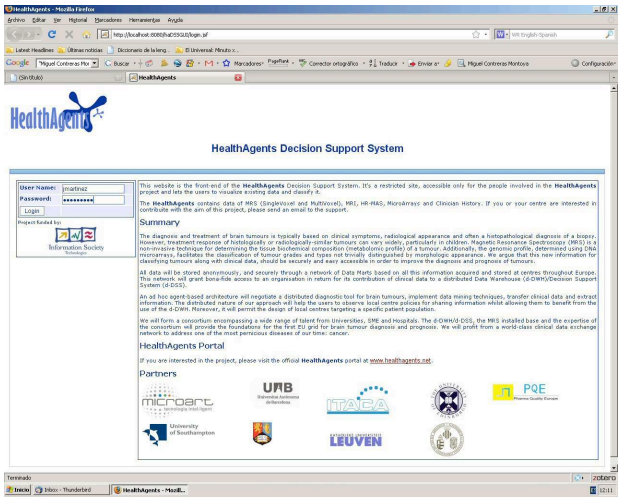


Figure 11 HA DSS login screen.

During the log on process, the user is authenticated according to his/her security permissions and if the user name and password are correctly registered, then the GUI Agent is started. The first task of the GUI Agent is to know if the node of the user has its own DB Agent. If a DB Agent exists, then the GUI Agent builds a SPARQL query to get some fields that are presented on the next screen. This information contains data such as the possible values for the *age*, *gender* of the patients, all the values for the patients' *geographical origin* and *tumour locations*. The values retrieved are used to fill the combo boxes that the user can manipulate to define a search criterion for a patient's case notes (see the *search neurooncological cases* screen on Fig 12).

The next screen after the authentication is where the user can request for the neurooncological records from his/her own database (if it exists) or from all the available external clinical centres. As stated before, in this screen the user can define the search criteria to filter all the medical data available on the HA network. These parameters include specific information of the patient such as the *gender*, the *range of age*, the *geographical origin* or the *tumour location* (if it is already available). Once the user has set the parameters for a search, the GUI Agent builds the corresponding SPARQL sentence which is sent to the Data Collector Agent who distributes the query and collects all the results retrieved from all DB Agents. After the GUI Agent receives the collected data, it presents the neurooncological records obtained to the user. The cases shown on

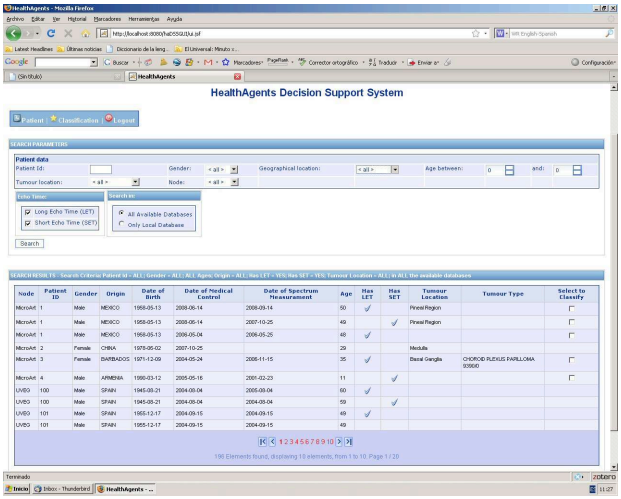


Figure 12 HA DSS screen for search neurooncological cases.

Fig 12 were obtained in a pilot study using three distributed clinical nodes, two of them with a different database schema and running with different database engines (MySQL and Oracle ver. 10). The neurooncological records listed in the results were obtained from two of the three nodes (those with different database schemata and engines) while the results of the third clinical node were hidden given the lack of user's permissions.

5 Reasoning with HaDOM

5.1 Subsumption, instance classification, and other ontological reasoning

HaDOM provides a controlled vocabulary for the use of classifiers and the construction of GUIs. For instance, depending one the requirements of the user, whether she is a radiologist or surgeon or oncologist, HaDOM provides a well-structured model to simplify the development of an adaptable user interface. Similarly, while HA classifiers take data from the data providers and generate classification labels using pattern recognition methods, HaDOM is used to regulate the input and output labels for the classifiers by offering a controlled vocabulary as a uniform communication interface. Classifiers are developed at different centres using different data sets and with the goal of resolving different questions. So, for instance, there may be classifiers developed to distinguish between low grade meningiomas and aggressive tumours, including high grade glioblastomas. The terminology belongs to the set of terms sanctioned by the WHO classification, and these classification labels reflects their actual usage amongst clinical practitioners. However, from a knowledge engineering viewpoint, such mixing of attributes of tumours (aggressive, or grade, or whether they have undergone metastasis) and its principal conceptual identity (glial tumours) has to be disentangled to create the conceptual hierarchical structure. Else, idiosyncratic usage in different clinical centres would make difficult the job of coordinating the outputs of classifiers developed in one centre but deployed elsewhere in the HA network for software agents. Using the capabilities of logical reasoning offered by OWL, identification of clusters of concepts used across the HA network in diagnosis offers potential future benefits for medical advances in this domain.

In this context, reasoning is defined as making explicit statements that are implicitly encoded in the representation. In the following, we will present reasoning capabilities of HaDOM. Generally speaking reasoning mechanisms supported by HaDOM relate to subsumption. More precisely, the ontology allows for one to infer out of (i) "A is a subclass of B" and (ii) "x is a member of A" that "x is also a member of B". If, for example, "Glioblastoma Multiforme" is considered as a subclass of "Astrocytoma" and an unknown mass is classified as "Glioblastoma Multiforme" based on its appearance and bio-chemical characteristics, then we can automatically infer that this unknown

mass is an astrocytoma. Another reasoning pattern is the inheritance of properties along the conceptual hierarchy. An ontology allows one to infer from (i) "A is a subclass of B" and (ii) "B has property P" that "A has property P". In the previous example, this unknown mass bears all the features defined on an astrocytoma and will automatically inherit the necessary constraints defined directly on Astrocytoma. Defaults and exceptions might be applicable in this case when complete domain knowledge is not available, but these specifications are not encoded directly in the ontology as elements to perform reasoning with, but only as properties to retrieve information about. Building upon the above mentioned mechanism we can also utilise reasoning to ensure the correctness of the knowledge acquisition process. If, for example, it is known that a patient is characterised by three properties: age, sex and location and a database query only retrieved two of such fields, then missing information is flagged up and notification of appropriate action provided.

5.2 Better information accessibility

HaDOM underpins the HA Evidence-based Search System (EbSS)(Matthews, 2008) for well-targeted information extraction from on-line literature and patient databases. In evidence-based medicine (EBM), pieces of evidence from various scientific studies are evaluated and applied to ensure that the best outcomes can be expected based on the current status of knowledge. Hence, in EBM, identifying and retrieving appropriate information is critical. In many cases, such information is not readily available and the clinicians and other information requestors are overwhelmed by a large number of publications from online repositories such as PubMed[14], *e*medicine[15], etc. Finding useful information from such sources could be time consuming and inefficient. Classifying clinical research against a domain ontology imposes a schematic view over the information sources that helps the requestors to quickly zoom in and identify the most relevant information. For instance, when searching for diagnosis and prognosis information with respect to *Choroid Plexus Carcinoma*, HaDOM allows one to extend queries to not only the parent concepts of this particular type of brain tumour but also the new tumour type defined in WHO 2007 classification by means of the links/properties among HaDOM concepts. Similar systems based on general ontologies have been successfully commercialised (c.f. goPubMed[16]). HA EbSS, different from such general-purpose information portals, takes advantage of HaDOM in generating queries and filtering search results that tuned specifically against HA domains.

Rendering information in a meaningful way also has implications for how well information is conveyed and apprehended (Herman et al., 2000). In the HA system, information is collected from different sources and is displayed based on the nature of the request and identity of users. This lays down two requirements on HA user interface, namely integrating and role-based information provision. When implementing the HA system, the integrating requirement is facilitated by annotating clinical data using concepts from HaDOM and projecting it onto patients' EHRs in a chronological manner. From EHR, therefore, users can navigate to the clinical history of a patient, the various clinical investigation performed on him/her, etc. It is also possible to retrieve relevant clinical research literature displayed alongside patient's EHR via the HA EbSS. The HA system also practices a strict information filtering process based on the roles of users. Currently, HaDOM defines a list of roles that can be played by a human user or a software agent. Associated with each role is its rights and authorisation that are used to annotate fragments of patient data. When browsing and navigating through a patient's EHR, one is presented with the data that he or she has clearance for and is prohibited from viewing or modifying those for which he or she has not been granted access rights.

[14] `http://www.ncbi.nlm.nih.gov/pubmed/`
[15] `http://emedicine.medscape.com/`
[16] `http://www.gopubmed.com/`

6 Evaluation and discussion

The ontology presented in this paper is currently functioning as the inter-lingua between different agents within the HA system. This has two implications for ontology evaluation. First, this application-centric role of the ontology means that the evaluation has to look at how this shared conceptualisation benefits the system from a communication viewpoint. Meanwhile, as explained above, HaDOM serves to specify *what* information is passed around. This means that the evaluation will also have to take into consideration the application's domain of discourse. Regarding the first point, a measure of how effective a particular ontology is in the context of an application, we will need a number of other, yet related ontologies that can be used in this context to compare ours with. However, this application-based approach to ontology evaluation is not suitable given the fact that the ontology is only used in task-specific ways, and it is difficult to generalise this observation. Also, given the novelty of our project a comparison with related ontologies is not possible on a large, integrated scale. A very small number of different modules developed for describing brain structures or general tumour classes could be related as separate modules. However, the relevance of such modules to our evaluation is seen more in the context of the second raised point above, namely the domain of discourse. Unfortunately when trying to evaluate our ontology from a "domain" viewpoint, another problem occurs: it is hard to determine who the right participants are, if a user-based evaluation is to be performed and what criteria should be used to interpret the results. Indeed, in this case, given the main purpose of the ontology it is not clear who the right users are and what such qualitative evaluation means (see (Brewster et al., 2004)). Moreover, comparing such different ontologies is only possible if they can all be plugged into the same application and this takes us back to the initial point detailed above.

Given this rationale we will evaluate our work by **validating** the ontology with respect to its purposes. According to Gangemi et al. (2006) an ontology validation needs to look at three different aspects: task assessment, agreement assessment and topic assessment. These three points correspond to the initial requirements presented in Section 1. Indeed, the system functionality validation will ensure the task assessment, meeting the clinician terminological requirements addresses the topic assessment and the smooth transition from existing data nomenclature towards the agreed nomenclature will allow for agreement assessment. Thus, in validating HaDOM we will assess the work presented in this paper with respect to:

- System functionality. In the light of this requirement we have demonstrated how the ontology functions as a common vocabulary amongst the different databases and how it is used by various agents within HA.
- Clinician terminology. We have validated the ontology throughout the duration of the project by having regular meetings with clinical partners in order to discuss both the terminology used and how it will impact on the development of the system. The terms used in HaDOM are those that have been agreed upon with the domain experts and which are consistent with the envisaged use of the system.
- The smooth translation from existing data nomenclature towards the agreed nomenclature. This has been demonstrated by the development of the Mapping Toolkit detailed at length in the previous sections. Not only does this toolkit facilitate creating mappings between legacy databases and the HA system but also, without such a toolkit, the manual creation of such scripts would be impossible for the new clinical partners joining HA in the future.

6.1 System functionality

The HA system is effectively running with data nodes residing in both Spain (i.e. Universitat Autónoma de Barcelona, UAB, and Universitat de Valéncia, UV) and the United Kingdom (i.e. The Birmingham Children's Hospital, BCH). Each data provider is allowed to maintain the integrity of their legacy data to avoid disruption to existing tools and systems. In the meantime, the heterogeneity inherent in the independently collected data is tackled by means of HaDOM.

Although further evaluation of the HA system is necessary, usability and reliability studies of the current release of the system have confirmed that:

- HaDOM is sufficiently expressive to cover the legacy data from all participating hospitals and clinical centres. Data has been faithfully converted and no knowledge loss has been reported.
- HaDOM is capable of representing inputs and outputs of classifiers and other data processing agents. HaDOM serves as the unified language to ensure service and data interoperability within HA.
- Modularised HaDOM enhances the extensibility of the HA system and enables specialist software agents. For instance, data anonymising agents can be developed against each imaging and clinical module with their outputs projected upon HaDOM for alignment.

In summary, HaDOM successfully facilitates an unobtrusive mechanism to transfer heterogeneous data among different sites without requiring the active engagement of human users. On the other hand, a major disadvantage of HaDOM has been revealed during the evaluation. The HA classifiers normally offers class labels together with numeric values to justify and contextualise the classification. Thus far, HaDOM uses a URI to point to a data file holding such values (e.g. matrices) or treats them as strings using string type properties (e.g. hasParameter). While such approaches have been demonstrably successful in leveraging diagnostic classification tasks to be executed over the HA network, they offer access to the patterns in the data only through their algorithms and interfaces. This precludes any possibility of combining reasoning based on these numeric values directly with ontology based inferences. Although conceiving a new reasoning algorithm enhancing ontology with reasoning on concrete datatype is beyond the scope of HA project, research on integrating logic based knowledge representation formalisms with uncertainty is relevant to this task (c.f. (Lukasiewicz, 2008)(Costa, 2005)).

6.2 Clinical terminology

The diversity of the HA consortium offers a good test bed for HaDOM. The first version of HaDOM was mainly based on published literatures, interviewing various domain experts from UAB and observing the daily work of selected domain experts with the think-aloud protocol (Wright and Monk, 1990). This draft version was then reviewed by domain experts (potential HA users and clinical consultants) from UV and BCH in three consecutive steps. Firstly, the domain experts were given a pre-interview so as to build up essential knowledge on HaDOM and to introduce them to the idioms of knowledge representation languages (namely DL constructs). They then walked through the ontology with or without the help of knowledge engineers. A post-interview was performed against a questionnaire to collect their questions, comments, and observations. This variant of the usability evaluation method (Rubin, 1994) is based on practical considerations — the limited availability of clinicians prevents a prolonged interview and thus a guided one could ensure that necessary feedback was duly gathered. Expert feedback was used to revise HaDOM.

Moreover, HaDOM revision was reviewed against the eTUMOUR data model. One of eTUMOUR's objectives was to collect real patient data for establishing effective clinical decision support methods. eTUMOUR consortium overlaps with HA consortium and was expected to share a large amount of data with HA. Therefore, HaDOM should be compatible with the database schemata from eTUMOUR. One of the consequences is that HaDOM's naming and modelling conventions have to accommodate the design considerations in eTUMOUR. Such a link was made through the development team in MicroArt which was responsible for database design in both projects.

Finally, HaDOM was further evaluated through manually constructing mappings between the domain ontology and legacy database schema by domain experts. A major assumption behind such an approach is that one can safely conclude that HaDOM satisfies the applicability and usability requirements if a domain expert with limited knowledge on ontology engineering could establish the mappings correctly. Two experts from BCH responsible for handling patient data

were summoned for the study. With guidance from knowledge engineers, mapping was successfully constructed. Feedback from the two domain experts led to further changes on HaDOM including new names and conceptual structures.

6.3 Facilitating translation

The last phase in evaluating HaDOM was done with the help of the graphical mapping toolkit. Thus far, many mapping tools are available to suggest candidates between ontologies and database schemata (see for instance the survey by Rahm and Bernstein (2001)). A strong argument against adopting such automatic mapping methods in HA is that although labelled with "semantics", most approaches fail to inspect semantics in terms of cognitive expectations that emanate from working within established working practices within institutions. In addition, these cognitive biases get shaped within different perspectives and implicit conceptual models rooted in the users' educational, cultural and societal background (Fodor, 2004). It is unrealistic to expect effective automation for ironing out these potentially distinct conceptions, raising the need for accommodating these discrepancies when constructing mappings. The situation is aggravated when mapping HaDOM against legacy database schemata due to the fact that we have to observe the integrity, specificity and historically acquired idiosyncrasies of the latter.

Furthermore, in practice, we found the following difficulties prevented us from adopting automated mapping tools. Firstly, the diversity of the legacy database schemata made automated approaches less feasible. There were cases that one HaDOM concept was mapped to more than one database table combined through a series of *join* operations; one HaDOM concept was mapped to a number of columns that did not have obvious relations one could rationally describe; one HaDOM property was mapped to multiple table columns depending on whether or not it satisfied certain auxiliary conditions; etc. Secondly, many well-performed automated mapping tools rely on external data sources, e.g. WordNet (Miller, 1995), reference ontologies, or instance data. Such information was either not available from the legacy databases due to patient privacy concerns or not applicable because of the existence of a large number of hospital specific abbreviations and acronyms. Using an automated mapping tool would require tuning the tool against individual hospitals and would lead to prolonged validation phases. Thirdly, even if an automated tool had been used, human involvement would have been inevitable due to strict patient safety requirements. The benefit of adopting such tools was not evident giving the size of the problem—after modularisation, each domain expert, with even a rudimentary understanding of the meanings of the entries of the database schema, would not find it difficult to map and review mappings of about 30 concepts. The mapping toolkit, however, leverages basic string similarity metrics to recommend potential mapping candidates.

A preliminary usability study of the graphical mapping tool was carried out. Four people with different computing skills and different backgrounds were selected. They were presented with the HaDOM ontology and one of the real-life legacy database schemata and were asked to map a few preselected concepts. The feedback on the user interface and the automated generated D2RQ script was positive, suggesting that the layout was intuitive and significantly reduced typos and human errors. Negative comments include the confusing visualisation of the ontology, the difficulty of navigating through different tabs and windows and the lack of "intelligence" in recommendations—recommendations are mainly based on name matching. We expect to carry on development of the graphical mapping toolkit and perform formal usability and design studies beyond the HA project.

6.4 Concluding remarks

In conclusion, this paper presents our efforts towards building an ontology for HA. Our main motivation behind the work was driven by a desire to provide a declarative framework to separate the functionality of the system from an articulated interface derived from user requirements that

were informed by frequent meetings for validation with the clinical partners. We have shown how we implemented the ontology, as well as the mechanisms for accessing the data using the ontology by domain specific examples. The main contribution of the paper is two fold. On one hand we show a "hands on" example of building an ontology in practice, and how making it work in distributed settings requires translations and intermediate placeholders in order to include legacy representations. This is especially important in an era where more and more information is acquired and annotated with metadata so that methods for their (semi) automatic informed manipulation become essential. On the other hand we make explicit the implicit modeling choices when building an application ontology for a given domain. This has been an interesting process that could serve as a future reference point for similar work.

7 Acknowledgements

The knowledge representation problems addressed by this paper have streamed from the domain experts' requirements. Without this knowledge acquisition step the HaDOM ontology would have never successfully developed. Our acknowledgements go to domain experts across the HA project: Margarida Julia-Sapé, Andrew Peet, Francesc Estanyol, Liang Xiao, Yu Sun, Kal Natarajan, and Javier Vicente Robledo.

This paper reflects only the authors' views. The European Community is not liable for any use that may be made of the information contained herein. This research is carried out within the EU FP6 Project HA: Agent-Based Distributed Decision Support System for Brain Tumour Diagnosis and Prognosis [IST-2004-027214].

References

Baader, F., Calvanese, D., McGuinness, D., Nardi, D. and Patel-Schneider, P., eds (2003), *The Description Logic Handbook: Theory, Implementation and Applications*, Cambridge University Press.

Benslimane, D. adn Thiran, P., Lu, J., Wyss, C. and Göschka, K., eds (2007), *Third International Workshop on Database Interoperability*.

Bizer, C. and Seaborne, A. (2004), D2RQ - Treating Non-RDF Databases as Virtual RDF Graphs, *in* 'Proceedings of the 3rd International Semantic Web Conference'. poster presentation.

Brewster, C., Alani, H., Dasmahapatra, S. and Wilks, Y. (2004), Data driven ontology evaluation, *in* 'Proceedings of International Conference on Language Resources and Evaluation (LREC04)', Lisbon, Portugal.

Bussler, C., Tannen, V. and Fundulaki, I., eds (2005), *Semantic Web and Databases, Second International Workshop, SWDB 2004, Toronto, Canada, August 29-30, 2004, Revised Selected Papers*, Vol. 3372.

Cohen, W., Ravikumar, P. and Fienberg, S. (2003), A comparison of string distance metrics for name-matching tasks, *in* 'IIWeb', pp. 73–78.

Costa, P. (2005), Bayesian Semantics for the Semantic Web, PhD thesis, School of Information Technology and Engineering, George Mason University.

Damasio, H. (1995), *Human brain anatomy in computerized images*, Oxford university press.

Favre, J., Taha, J. M. and Burchiel, K. J. (2002), 'An analysis of the respective risks of hematoma formation in 361 consecutive morphological and functional stereotactic procedures', *Journal of Neurosurgery* **50**(1), 56–57.

Field, M., Witham, T., Flickinger, J., Kondziolka, D. and Lunsford, L. (2001), 'Comprehensive assessment of hemorrhage risks and outcomes after stereotactic brain biopsy', *Journal of Neurosurgery* **94**, 545–551.

Fodor, J. (2004), 'Having Concepts: a Brief Refutation of the Twentieth Century', *Mind & Language* **19**(1), 29–47.

Gangemi, A., Catenacci, C., Ciaramita, M. and Lehmann, J. (2006), Modelling ontology evaluation and validation, *in* 'Proceedings of the 3rd European Semantic Web Conference', pp. 140–154.

González-Vélez, H., Mier, M., Julià-Sapé, M., Arvanitis, T. N., García-Gómez, J. M., Robles, M., Lewis, P. H., Dasmahapatra, S., Dupplaw, D., Peet, A., Arús, C., Celda, B., Huffel, S. V. and Lluch i Ariet, M. (2009), 'HealthAgents: Distributed multi-agent brain tumor diagnosis and prognosis', *Applied Intelligence* **30**(3), 191–202.

Gruber, T. (1993), 'A translation approach to portable ontology specification', *Knowledge Acquisition* **5**(2), 199–221.

Hahn, U., Schulz, S. and Romacker, M. (1999), 'Part-whole reasoning: A case study in medical ontology engineering', *IEEE Intelligent Systems* **14**(5), 59–67.

Hall, W. (1998), 'The safety and efficacy of stereotactic biopsy for intracranial lesions', *Cancer* **82**, 1749–1755.

Herman, I., Melançon, G. and Marshall, M. S. (2000), 'Graph visualization and navigation in information visualization: A survey', *IEEE Transactions on Visualization and Computer Graphics* **6**(1), 24–43.

Hu, B., Dasmahapatra, S., Dupplaw, D., Lewis, P. and Shadbolt, N. (2007), 'Reflections on a medical ontology', *International Journal of Human-Computer Studies* **65**(7), 569–582.

Julià-Sapé, M., Acosta, D., Majós, C., Moreno-Torres, A., Wesseling, P., José Acebes, J., Griffiths, J. R. and Arús, C. (2006), 'Comparison between neuroimaging classifications and histopathological diagnoses using an international multicenter brain tumor magnetic resonance imaging database', *Journal of Neurosurgery* **105**(1), 6–14.

Kalfoglou, Y., Hu, B., Reynolds, D. and Shadbolt, N. (2005), Semantic integration technologies, 6th month deliverable, University of Southampton and HP Labs.

Lukasiewicz, T. (2008), 'Probabilistic description logic programs under inheritance with overriding for the semantic web', *Int. J. Approx. Reasoning* **49**(1), 18–34.

Matthews, M. (2008), EbSS: Evaluating on-line information retrieval dedicated to brain tumour, *in* 'Proceedings of the HealthAgents Workshop at the 8th Congress of the European Association of Neuro-Oncology (EANO2008)'.

Miller, G. A. (1995), 'WordNet; a Lexical Database for English', *Communications of the ACM* **38**(11), 39–41.

Mol, A. (2003), *The Body Multiple: Ontology in Medical Practice*, Duke University Press, Durham, NC.

Noy, N. and Musen, M. (2004), 'Ontology versioning in an ontology management framework', *IEEE Intelligent Systems* **19**(4), 6–13.

Quine, W. V. O. (1953), On what there is, *in* 'From a Logical Point of View', Harper & Row (New York).

Rahm, E. and Bernstein, P. (2001), 'A survey of approaches to automatic schema matching', *The VLDB Journal* **10**, 334–350.

Rector, A. (1999), 'Clinical Terminology: Why Is it so Hard', *Methods of Information in Medicine* **38**, 239–252.

Rosse, C. and Mejino, J. (2003), 'A reference ontology for biomedical informatics: the foundational model of anatomy', *Journal of Biomedical Informatics* **36**(6), 478–500.

Rubin, J. (1994), *Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests*, John Wiley & Sons.

Wright, P. and Monk, A. (1990), 'The use of think-aloud evaluation methods in design', *ACM SIGCHI Bulletin* **23**(1), 55–57.