1 An use case for DAML+OIL: an ontology in the ophthalmology domain

M. Taboada¹, M. Argüello¹, J. Des² and D. Martínez¹

¹ University of Santiago de Compostela, Spain
² Hospital of Monforte, Spain

1.1 Introduction

This report describes how we have developed a medical ontology for a heuristic application in the ophthalmology domain. This ontology has been built by reusing pre-existing knowledge theories, and it has been developed in two phases:

1) Firstly, the ontology has been modeled using the domain knowledge schema specification of CommonKADS [Schreiber et al., 1999], which in turn uses Unified Modelling Language (UML) [Rumbaugh et al., 1999] in order to represent graphically the objects, class and relations of the domain knowledge.

2) Currently, the ontology is being represented in DAML+OIL [Fensel et al., 2001].

In this report, we will illustrate in some depth how a medical ontology has been developed for a specific domain and a specific application, by reusing pre-existing medical theories. In addition, we will show how objects, classes and relations of a medical domain (usually in any object-oriented approach) can be specified in the Web language DAML+OIL.

1.2 The clinical domain

Our medical ontology has been modeled for a heuristic application in the domain of the ‘Red Eye’ (RE) diagnosis. The latter is a common activity carried out in primary care units. However, a lot of general practitioners have difficulty to make a precise diagnosis. So, some recommendations oriented to facilitate this kind of diagnosis have been provided [Davey, 1996; Bertolini and Pelucio, 1995]. These advices are focused on helping to the physicians to:

1) Distinguish appropriately between minor and serious causes.
2) Encourage him to treat minor problems themselves, as most cases are benign and can be managed efficiently by him, such as Conjunctivitis, Blepharitis, Subconjunctival Hemorrhage or Episcleritis.

3) Refer appropriately those cases that require ophthalmologic consultation, such as Acute Angle Closure Glaucoma, Keratopathy, Uveitis or Scleritis, which must be referred as soon as possible.

Recommendations about carrying out the diagnosis of the red eye include to take a careful history and to make a focused ophthalmology examination. Following these recommendations, a diagnosis-aid system in this clinical domain should guide the practitioner on, at least,

1) carrying out the anamnesis,
2) directing the ophthalmology examination,
3) providing advice about diagnostic hypothesis, and as a result of this,
4) suggesting actions to carry out, such as to treat the patient, to refer him to the specialist and/or to require some complementary test.

Taking into account this, the medical ontology must include, at least, knowledge about the ophthalmology anamnesis, examination, pathologies and relations among all of this knowledge.

1.3 Developing and specifying the ophthalmology ontology

Our domain ontology has been developed by reusing some theories in the core library described in [Falasconi and Stefanelli, 1994], and later, by extending these theories with some descriptions taken from INTERNIST-I [Miller et al., 1982] and with more specific concepts to our clinical domain. We have also followed the ICD-9-MC\(^*\) classification (The International Classification of Diseases, 9\(^{th}\) Revision, Clinical Modification, 1978, Commission on Professional and Hospital Activities, Ann Harbor. Michigan 48705), for representing pathologies. Currently, there is a 10\(^{th}\) Revision, but the 9\(^{th}\) Revision is still followed in Spain. In this way, we have obtained a standard representation vocabulary, which is being revised by following UMLS (Unified Medical Language Standard).

The library of medical ontologies of [Falasconi and Stefanelli, 1994] specifies a set of medical conceptualizations taken from both medical literature and implemented systems, including general categories of medical knowledge, such as the following theory levels:

- **Generic Patient**, which models the medical activities for each patient,
- **Test and Therapies**, representing actions undertaken by medical agents,
✓ *Diseases*, modelling each disease as a clinical process whose evolution is described through finding and clinical abstraction values over time,
✓ *Findings, Clinical-state-abstractions, Drugs, etc.*

Fig. 1 shows a small part of our domain ontology. The higher levels in this ontology correspond to the core ontological part (labelled as Medical Central Ontology, MCO), following the approach proposed by [Heijst et al., 1997]. MCO includes the reused general categories of medical knowledge, such as 'Generic-Patient', 'Tests', 'Diseases', 'Findings', 'Clinical-State-Abstractions', etc. As the hierarchical structures into MCO are traversed, the medical concepts are more specific to our clinical domain, giving rise to the peripheral ontological part (labelled as Medical Peripheral Ontology, MPO). In particular, Fig. 1 shows some concepts from our ontology with regard to some medical theories. In this report, we will focus in Generic-Patient and Test, in order to describe the most important aspects used during the specification in DAML+OIL.

Figura 1: A Small Part of our Medical Domain Level
1.3.1 Generic Patient

Generic-Patient models the information to record during each patient visit, such as patient identifying data, patient visit record and the anamnesis. The latter includes chief complaint, history of present illness and past medical history.

In DAML+OIL, objects are described by giving a name for the object class. The following text box shows the definition of the set of classes in Generic-Patient.

```xml
<daml:Class rdf:ID="PatientIdentifyingData">
</daml:Class>
<daml:Class rdf:ID="PatientVisitRecord">
</daml:Class>
<daml:Class rdf:ID="Anamnesis">
</daml:Class>
<daml:Class rdf:ID="ChiefComplaint">
</daml:Class>
<daml:Class rdf:ID="HistoryOfPresentIllness">
</daml:Class>
```

The set of attributes, which describes each class of objects, is defined by a kind of DAML+OIL property (named `daml:DatatypeProperty`), which relates objects to datatype values. For example, we have create the attribute `Name`, which maps strings into XML Schema,

```xml
<daml:DatatypeProperty rdf:ID="Name">
<rdfs:comment>
Name is a DatatypeProperty whose range is xsd:string.
Name is also a UniqueProperty (can only have one name)
</rdfs:comment>
.rdf:type
<daml:DatatypeProperty rdf:ID="Name">
<rdfs:range
<rdfs:range
date:resource="http://www.daml.org/2001/03/daml+oil#UniqueProperty"/>
<rdfs:range
<rdfs:range
date:resource="http://www.w3.org/2000/10/XMLSchema" #string"/>
</daml:DatatypeProperty>
```
or the attribute Date of Birth, which is a subproperty of the Date property. The latter maps dates into XML Schema.

```xml
<owl:DatatypeProperty rdf:ID="Date"
<rdfs:comment>
  Date is a DatatypeProperty whose range is
  xsd:date.
</rdfs:comment>
<rdfs:range
  rdf:resource="http://www.w3.org/2000/10/XMLSchema#date"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="DateOfBirth"
<rdfs:comment>
  DateOfBirth is a subProperty Of Date and it is also
  a UniqueProperty (can only have one date)
</rdfs:comment>
<rdfs:subPropertyOf rdf:resource="#Date"/>
<rdf:type
  rdf:resource="http://www.daml.org/2001/03/daml+oil#
  UniqueProperty"/>
</owl:DatatypeProperty>
```

Figure 2 shows how we can describe the internal structure of a class by using property restrictions. These restrictions can be added to the class by using

```xml
<daml:Class rdf:about="#PatientIdentifyingData">
```
Fig 2: Adding two property restrictions to the class PatientIdentifyingData

In an object-oriented approach, one standard relation is an aggregation. For example, for each patient there can be one or several visit records. This information can be modeled by an aggregation relation 1:N between PatientIdentifyingData and PatientVisitRecord. We have specified this kind of relation by:

1) defining a new property `hasVisit`, which is a binary relation connecting the two classes implied in the aggregation.

```
<daml:ObjectProperty rdf:ID="hasVisit">
  <rdfs:domain rdf:resource="#PatientIdentifyingData"/>
  <rdfs:range rdf:resource="#PatientVisitRecord"/>
</daml:ObjectProperty>
```

2) adding a new property restriction on the class PatientIdentifyingData, such as is displayed in Fig. 3, where a minimum cardinality of 1 is specified as the aggregation relation is 1:N.
If a class implied in an aggregation relation is defined in another file, the class must be referred with a fully qualified URL. For example, the file

http://aiff.usc.es/~elchus/daml+oil-medicalontology/daml+oil-gp.daml
defines all concepts regarding to Generic-Patient. So, in order to define the aggregation between PatientVisitRecord and Test, we have defined a new property (madeTests) and added it to PatientVisitRecord as a new restriction.

```xml
<daml:ObjectProperty rdf:ID="madeTests">
  <rdfs:domain rdf:resource="#PatientVisitRecord"/>
  <rdfs:range rdf:resource="http://aiff.usc.es/elchus/medicalontology/daml+oil-test.daml#Test"/>
</daml:ObjectProperty>

<daml:Class rdf:about="PatientVisitRecord">
  <rdfs:comment>
  Several tests can be made or recommended for their accomplishment during each patient visit, ie: Zero or more tests can be aggregated to one 'PatientVisitRecord' (through the object property 'madeTests').
  </rdfs:comment>
  <rdfs:subClassOf>
    <daml:Restriction daml:minCardinality="0">
      <daml:onProperty rdf:resource="#madeTests"/>
    </daml:Restriction>
  </rdfs:subClassOf>
</daml:Class>
```

1.3.2 Test

We have specified the set of following classes and their hierarchies (some of these disjoint), according to the theory level Test (see Fig. 1).
<daml:Class rdf:ID="Test">
</daml:Class>

<daml:Class rdf:ID="ComplementaryTest">
<rdfs:subClassOf rdf:resource="#Test"/>
</daml:Class>

<daml:Class rdf:ID="PhysicalExamination">
<rdfs:subClassOf rdf:resource="#Test"/>
<daml:disjointWith rdf:resource="#ComplementaryTest"/>
</daml:Class>

<daml:Class rdf:ID="EyePhysicalExamination">
<rdfs:subClassOf rdf:resource="#PhysicalExamination"/>
</daml:Class>

<daml:Class rdf:ID="SystemicPhysicalExamination">
<rdfs:subClassOf rdf:resource="#PhysicalExamination"/>
<daml:disjointWith rdf:resource="#EyePhysicalExamination"/>
</daml:Class>

<daml:Class rdf:ID="ExplorationPhase">
</daml:Class>

<daml:Class rdf:ID="PrimitivePhase">
</daml:Class>

<daml:Class rdf:ID="Technique">
</daml:Class>

<daml:Class rdf:ID="IntrumentalTechnique">
<rdfs:subClassOf rdf:resource="#Technique"/>
</daml:Class>

<daml:Class rdf:ID="No-IntrumentalTechnique">
<rdfs:subClassOf rdf:resource="#Technique"/>
<daml:disjointWith rdf:resource="#IntrumentalTechnique"/>
</daml:Class>

<daml:Class rdf:ID="PharmacologicalAction">
</daml:Class>
Some properties, as DateOfRequest, DateOfAccomplishment, ApplicantPhysician, have been specified as subproperties of properties defined in 'Generic-Patient'.

```xml
<daml:DatatypeProperty rdf:ID="DateOfRequest">
  <rdfs:comment>
    DateOfRequest is a subProperty Of Date, which has been defined en 'Generic-Patient', and it is also a UniqueProperty (can only have one date)
  </rdfs:comment>
  <rdfs:subPropertyOf rdf:resource="http://aiff.usc.es/elchus/medicalontology/daml+oil-gp#Date"/>
  <rdf:type rdf:resource="http://www.daml.org/2001/03/daml+oil#UniqueProperty"/>
</daml:DatatypeProperty>

<daml:DatatypeProperty rdf:ID="DateOfAccomplishment">
  <rdfs:comment>
    DateOfAccomplishment is a subProperty Of Date, which has been defined en 'Generic-Patient', and it is also a UniqueProperty (can only have one date)
  </rdfs:comment>
  <rdfs:subPropertyOf rdf:resource="http://aiff.usc.es/elchus/medicalontology/daml+oil-gp#Date"/>
  <rdf:type rdf:resource="http://www.daml.org/2001/03/daml+oil#UniqueProperty"/>
</daml:DatatypeProperty>

<daml:DatatypeProperty rdf:ID="ApplicantPhysician">
  <rdfs:comment>
    ApplicantPhysician is a subproperty of IdentifyingNumber.
  </rdfs:comment>
  <rdfs:subPropertyOf rdf:resource="#IdentifyingNumber"/>
  <daml:equivalentTo rdf:resource="http://aiff.usc.es/elchus/medicalontology/daml+oil-gp #PhysicianIdentifyingNumber"/>
</daml:DatatypeProperty>

<daml:DatatypeProperty rdf:ID="JustificationOfAccomplishment">
  <rdfs:comment>
    JustificationOfAccomplishment is a DatatypeProperty whose range is xsd:string, and it is also a UniqueProperty (can only have one name)
  </rdfs:comment>
  <rdf:type rdf:resource="http://www.daml.org/2001/03/daml+oil#UniqueProperty"/>
</daml:DatatypeProperty>
```
Binary relations between objects (1:N cardinality) have been specified by defining a new `daml:ObjectProperty` and adding a new restriction on the class. For example in Fig. 1 the relation labeled as ‘Can be carried out by’ between *Exploration Phase* and *Primitive Phase* in the Test theory, is a 1:N binary relation with one relation attribute, named ‘Choice Criteria’. Firstly, two new properties have been defined in this case: one for defining the binary relation and another for defining the relation attribute. Secondly, two restrictions have been added to the class *Exploration Phase*.

```xml
<daml:ObjectProperty rdf:ID="CarriedOutBy">  
  <rdfs:domain rdf:resource="#ExplorationPhase"/>  
  <rdfs:range rdf:resource="#PrimitivePhase"/>  
</daml:ObjectProperty>

<daml:ObjectProperty rdf:ID="ChoiceCriteria">  
  <rdfs:domain rdf:resource="#ExplorationPhase"/>  
  <rdfs:range rdf:resource="#OrderedListOfCriteria"/>  
</daml:ObjectProperty>

<daml:Class rdf:about="ExplorationPhase">  
  <rdfs:comment>    
    Several primitive phases can be carried out by a method of examination, ie: One or more primitive phases can be carried out to one 'Exploration-Phase' (through the object property 'CarriedOutBy').  
  </rdfs:comment>  
  <rdfs:subClassOf>    
    <daml:Restriction daml:minCardinality="1">      
      <daml:onProperty rdf:resource="#CarriedOutBy"/>    
    </daml:Restriction>  
  </rdfs:subClassOf>  
  <rdfs:subClassOf>    
    <daml:Restriction daml:Cardinality="1">      
      <daml:onProperty rdf:resource="#ChoiceCriteria"/>    
    </daml:Restriction>  
  </rdfs:subClassOf>  
</daml:Class>
```
On the other hand, a n-ary relation has been specified by defining n-1 properties and adding n-1 restriction to one class. For example, in Fig. 1 the relation ‘Collects by using’ is a 4-ary relation, which represents the knowledge about techniques to be used, pharmacological actions to be applied, and signs to be collected in each phase. Three properties have been specified: ‘Collects’ between PrimitivePhase and Sign, ‘ByUsing’ between PrimitivePhase and Technique, and ‘ByApplying’ between PrimitivePhase and PharmacologicalAction.
1.3.3 Relation between Findings and Clinical-State-Abstractions

Clinical-State-Abstractions are more abstract descriptions than findings, and are connected to findings by relations that map data from specific descriptions to higher level descriptions. These relations have been modeled as rule schemas [Schreiber et al., 1999], which are association relations between two expressions. An expression can be modeled as a class containing four slots [Schreiber et al., 2000]: 1) a concept/relation, 2) a slot, 3) an operator (such as equal, greater, etc.) and 4) a value. We have also considered that expressions can be nested by using logical operators (such as and, or). Two examples of abstraction rules are shown in Figure 5.

\[
\begin{align*}
\text{Distance-from-Snellen-Chart.decimalvalueOD} &< 0.8 \\
\text{Distance-from-Snellen-Chart.decimalvalueOD} &< 0.8 \\
\text{abstracted-visual-line.difference} &\geq 2
\end{align*}
\]

\text{ABSTRACT}

\text{Visual-acuity-with-no-pinhole.abstracted} = \text{‘decreased’}

(a) An example of a qualitative abstraction by using a rule schema

\[
\begin{align*}
\text{Intraocular-pressure.abstracted} &= \text{‘increased’} \\
\text{[Cloudy-cornea.Presence} &= \text{‘yes’} \\
\text{Dolor-ocular.Presence} &= \text{‘yes’} \\
\text{Halos-around-lights.Presence} &= \text{‘yes’}
\end{align*}
\]

\text{ABSTRACT}

\text{Glaucoma-aparicion-brusca.Presence} = \text{‘yes’}

(b) An example of a definitional abstraction by using a rule schema

\textbf{Fig. 5: Two examples of abstraction rules}

Each expression, such as, \textit{‘Distance-from-Snellen-Chart.decimalvalueOD<0.8’} can be specified by defining a new class, for example, \textit{DjSCOD-evaluation}, as an intersection of the class \textit{‘Distance-from-Snellen-Chart’} and a DAML+OIL restric-
tion on a property (in this case, \textit{decimalvalueOD}). This restriction can use user-defined datatypes.

\begin{verbatim}
<daml:Class rdf:ID="DfSCOD-evaluation">
  <daml:intersectionOf rdf:parseType="daml:collection">
    <daml:Class rdf:about="#Distance-from-Snellen-Chart"/>
    <daml:Restriction daml:cardinality="1">
      <daml:onProperty rdf:resource="#decimalvalueOD"/>
      <daml:hasClass rdf:resource="http://aiff.usc.es/~elchus/medicalontology/daml+oi1-test-dt#under18"/>
    </daml:Restriction>
    </daml:intersectionOf>
  </daml:Class>

<daml:Class rdf:ID="DfSCOI-evaluation">
  <daml:intersectionOf rdf:parseType="daml:collection">
    <daml:Class rdf:about="#Distance-from-Snellen-Chart"/>
    <daml:Restriction daml:cardinality="1">
      <daml:onProperty rdf:resource="#decimalvalueOI"/>
      <daml:hasClass rdf:resource="http://aiff.usc.es/~elchus/medicalontology/daml+oi1-test-dt#under18"/>
    </daml:Restriction>
    </daml:intersectionOf>
  </daml:Class>

<daml:Class rdf:ID="AVLD-evaluation">
  <daml:intersectionOf rdf:parseType="daml:collection">
    <daml:Class rdf:about="#Abstracted-Visual-Line"/>
    <daml:Restriction daml:cardinality="1">
      <daml:onProperty rdf:resource="#diference"/>
    </daml:Restriction>
    </daml:intersectionOf>
  </daml:Class>
\end{verbatim}
A set of nested expressions can be specified by means of the boolean combination provided by DAML+OIL, such as `daml:intersectionOf`, `daml:unionOf` and `daml:complementOf`.

```
<daml:Class rdf:ID="Decreased-Visual-Acuity-Evaluation">
  <daml:unionOf rdf:parseType="daml:collection">
    <daml:Class rdf:about="#DfSCOD-evaluation"/>
    <daml:Class rdf:about="#DfSCOI-evaluation"/>
    <daml:Class rdf:about="#AVLD-evaluation"/>
  </daml:unionOf>
</daml:Class>
```

A rule can be expressed by a name, an antecedent and a consequent:

```
<daml:Class rdf:about="Decreased-Visual-Acuity-Abstraction">
  <rdfs:subClassOf>
    <daml:Restriction daml:Cardinality="1">
      <daml:onProperty rdf:resource="http://aiff.usc.es/~elchus/medicalontology/daml+oil-gp#Name"/>
    </daml:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <daml:Restriction daml:Cardinality="1">
      <daml:onProperty rdf:resource="#Decreased-Visual-Acuity-Antecedent"/>
    </daml:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <daml:Restriction daml:Cardinality="1">
      <daml:onProperty rdf:resource="#Decreased-Visual-Acuity-Consequent"/>
    </daml:Restriction>
  </rdfs:subClassOf>
</daml:Class>
```
where the range of the antecedent is the class \textit{Decresed-Visual-Acuity-Evaluation} and the range of the consequent is the class \textit{Abstracted-Decresed-Visual-Acuity}.

\begin{verbatim}
<daml:ObjectProperty rdf:ID="Decresed-Visual-Acuity-Antecedent">
  <rdfs:domain rdf:resource="#Decresed-Visual-Acuity-Antecedent"/>
  <rdfs:range rdf:resource="#Decresed-Visual-Acuity-Evaluation"/>
</daml:ObjectProperty>

<daml:ObjectProperty rdf:ID="Decresed-Visual-Acuity-Consequent">
  <rdfs:domain rdf:resource="#Decresed-Visual-Acuity-Consequent"/>
  <rdfs:range rdf:resource="#Abstracted-Decresed-Visual-Acuity"/>
</daml:ObjectProperty>
\end{verbatim}