Published in the proceedings of The International Conference on WWW/Internet 2004
October 6-9, 2004, Madrid (Spain)

USING ONTOLOGIES TO BUILD VIRTUAL WORLDS FOR THE WEB

Wesley Bille[*], Olga De Troyer, Frederic Kleinermann, Bram Pellens, Raül Romero
WISE Research Group
Vrije Universiteit Brussel
Pleinlaan 2, B-1050 Brussel, Belgium
{Wesley.Bille, Olga.DeTroyer, Frederic.Kleinermann, Bram.Pellens, Raul.Romero}@vub.ac.be

ABSTRACT

Today, the development of a VR application is still a long and difficult task. You need to be skilled in Virtual Reality (VR) technology to be able to develop a Virtual World. There are several technologies like VRML, Java 3D and X3D available to create 3D for the World Wide Web. To be able to develop a Virtual World for the Web, one needs to know how to use these technologies from a programmer's perspective or how to use authoring tools supporting one of these technologies. To ease the use of VR, there is a need to have modeling tools that allow the designer to model his Virtual World much more in terms of the domain expert point of view. In this paper, a new approach called "VR-Wise" for designing and developing Virtual Worlds, is introduced. This approach uses the domain expertise to shorten and facilitate the generation of a virtual world. To do this, our approach uses ontologies. As ontologies are describing a domain or part of a domain, the information contained in the ontologies can be used to express the Virtual World in terms of the terminology of the domain for which the Virtual World has to be generated. Therefore, the use of ontologies in our approach can be seen as a way of grasping already available knowledge. This allows a non-VR-skilled person to model his Virtual World in terms of his domain expertise. The proposed approach is in particular interesting for VR on the Web since the Virtual Worlds on the Web are usually limited to desktop VR and do not need the same level of realism and complexity as immersive or augmented VR. Furthermore, with the realization of the Semantic Web, more and more domain ontologies will become available. The proposed approach can open up the development of VR for the Internet to a much broader community.

KEYWORDS

Virtual Reality, Conceptual Modeling, WWW, Semantic Web, VR-Wise

1. INTRODUCTION

Today, Virtual Reality (VR) is used for solving different problems in different domains such as industry, military, medicine, teaching and entertainment. Also on the WWW, VR is appearing in applications for education, real state agencies, tourism, culture, …

The development of a VR application for such domains is still a long and difficult task, especially when it comes to VR including features such as complex behaviors and collision detection. Currently there are quite a number of different ways to develop VR applications. The available software can be categorized either as toolkits or as authoring tools. Toolkits (like VRML [1], Java3D [10], X3D [12], Unreal [14]) are programming libraries that provide a set of functions with which a skilled programmer can create VR applications. Authoring tools (like 3D Studio Max [8], Cosmo Worlds [15]) are complete programs with a

[*] Research Assistant for the Fund of Scientific Research – Flanders (FWO)

graphical interface for creating worlds without the need to resort to do detailed programming. However, these tools are still not intuitive enough for a non-VR-skilled person to use them. The problem here is that the Virtual World someone wants to create still must be expressed in terms of low-level building blocks of the VR technology and not in high-level terms of the domain of interest. Another problem is that for more advanced worlds, we still need some kind of scripting language in order to define for example complex behaviour. These scripting languages are again very low-level.

Another problem with current design approaches is the communication between the domain expert and the VR developer. During the first phases of the design process, communication is one of the major activities. Nowadays, brainstorming sessions are held and notes are made using sketches. The usable way of communication during this process is natural language. However, natural language and sketches are informal, very often ambiguous, and incomplete. Very often mistakes are made because words can be interpreted in a variety of ways. The consequence of such misunderstandings is that a vast part of a VR system might need to be remodeled or rebuild. This results in delays and increasing development costs. Another disadvantage of natural language and sketches is that it cannot be processed automatically. Therefore, there is a need to have modeling tools which will help to formalize the communication between the domain expert and the designer and allow modeling the Virtual World much more from the viewpoint of the domain under consideration.

Very little research on conceptual modeling for VR is done. Care should be taken with the terms ‘modeling’ and ‘high-level’ when used in the context of VR. In VR the term ‘modeling’ is usually used to indicate the process of shaping the concepts of the Virtual World. The term ‘high-level’ usually refers to the possibility to focus on what to render instead of how to render it. The aim of our research is to be able to specify a Virtual World at a conceptual level and (semi-) automatically derive the Virtual World from this conceptual specification. Related work is this direction and also the work on design methodologies in VR is limited. In the recent years, some people have started to develop methods which separate the design process into a high level and a low level phase, helping the developer to design realistic interaction techniques [11][13]. Others have developed methods which take a more user-centered approach to the specification, design, development and evaluation of the Virtual Environment [3]. However none of the methods makes use of a domain-based approach although this will make the creation of a Virtual World more intuitive.

In this paper, we describe how Virtual Worlds on the Web can be created using an (existing) domain ontology. The domain ontology is used for specifying the world in a more intuitive way. The proposed approach, called “VR-Wise”, starts by specifying the Virtual World at a conceptual level, in terms of the domain and free from any implementation details. For this purpose ontologies are used. The use of ontologies has the following advantages:

- **Grasping the knowledge of a domain**
  The domain knowledge is captured in the domain ontology. Such an ontology for a specific domain of interest can be either existing or can be created explicitly for this purpose. When a new one is created, the knowledge of a domain expert is incorporated in the domain ontology.

- **Expressing the Virtual World much more in terms of the domain**
  By using an ontology for capturing the knowledge of the specific domain, the design of the Virtual World can be expressed using the domain terminology. Therefore, it is easier for the domain expert to understand the design. For instance, the requirement to have a bottle of wine standing on a shelve in the Virtual World can be modeled as an object ‘wine bottle’ ‘on top of’ an object ‘shelve’, rather than bothering him with exact representations and positions of the bottle and the shelve in the Virtual World.
- **Generating the Virtual World more easily**
  It is possible to derive a number of properties for an object from the knowledge available in the domain ontology. For instance, a dishwasher could be described in the domain ontology as having a length, a depth and a height. Using this knowledge, our tool may be able to deduce that a suitable VR primitive to represent the dishwasher is a box. Using a traditional VR authoring tool, the designer would have to choose the correct VR primitive and map the length, depth and height of the dishwasher to the corresponding attributes of the VR primitive himself. Therefore, using the VR-Wise approach, it becomes easier to generate (semi-) automatically the required Virtual World.

- **Fits the Semantic Web concept**
  Using the ontology approach, the Virtual World will automatically be annotated with an ontology and therefore fits exactly the Semantic Web concept. This will open many interesting possibilities such as allowing search engines to explore a Virtual World, allow information exchange between on-line Virtual Worlds, allow agents to migrate from one Virtual World to another, and so on.

In the VR-Wise approach, ontologies are used explicitly as a conceptual modeling tool, allowing a non-VR-skilled person to model his Virtual World using more intuitive modeling concepts and more oriented towards the domain of knowledge. From this conceptual specification, code for the Virtual World can be (partially) generated.

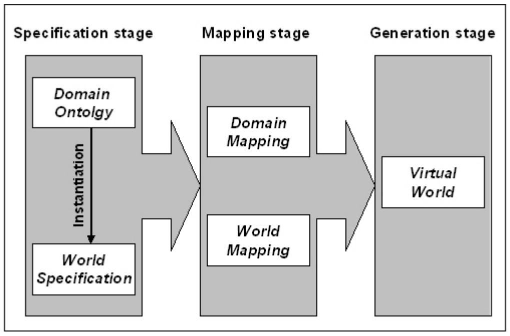
The rest of the paper continues by giving a detailed overview of the VR-Wise approach. In order to illustrate the idea behind the approach, an example of a 3D e-shop is worked out. The paper finishes by drawing conclusion and discussing the limitations of the current stage of the research as well as future work.

2. OVERVIEW OF THE APPROACH

Ontologies are very popular these days and therefore they are used in different domains. In its most simple form, we can say that an ontology is an abstraction of a computer-based lexicon, thesaurus, glossary or some other type of structured vocabulary, suitably extended with knowledge about a given domain [4][5].
Ontolgies are considered as the solution for many problems that are related with terminology.

In this research we will not use ontologies to solve terminology problems, but to facilitate the development of Virtual Worlds. A domain ontology can be considered to be a representation of a domain conceptualization describing possible concepts and relationships between these concepts. The information contained in such a domain ontology is well suited to be used as a starting point to specify a Virtual World at a conceptual level. This can be done by allowing the designer to select concepts from the domain ontology that are needed in the Virtual World and by extracting properties and relationships between those concepts from the domain ontology. In this section, we will give an overview of the VR-Wise approach and present the architecture of our experimental tool. Although still in an experimental phase, the availability of such a tool has several advantages. It allows us to experiment with new modeling concepts, it can be used as a fast prototyping tool for VR and finally, our tool allows us to test the feasibility of our approach against real case scenarios.

Figure 1: overview of the software architecture



The development phase is divided into three stages, namely a *specification stage*, a *mapping stage* and a *generation stage* (see Fig. 1). As we will see later on, throughout the different stages, in addition to a domain, a number of different (internal) ontologies are used. All ontologies currently use the DAML+OIL ontology language [16] defined by means of the DAML Ontology. This ontology was already available. It was developed to be the normative reference on the precise syntax of the language constructs. In fact, it is the machine-readable RDF Schema definition [6] of DAML+OIL [16]. Note that there are more recent ontology language formats such as OWL [7]. The use of DAML+OIL in our prototype has no specific reason but is just an implication of what was available at the time we started the implementation. Note however that our software can easily be updated to more recent language formats.

2.1 Specification stage

In the specification stage, the designer specifies his VR application at a conceptual level using the domain terminology. First of all, he needs to select a domain ontology (or first create a new one using his domain knowledge). Note that it is possible that more than one domain ontology is needed for a single Virtual World. The domain ontology will be used to select the concepts from the domain that are needed in the Virtual World. However to instantiate the concepts, to place the different instance in the World, and to specify behavior and interaction for these instances general conceptual VR modeling concepts are needed. These modeling concepts are defined and described in the *VR Conceptual Modeling Ontology*.

The Virtual Reality Conceptual Modeling Ontology

As already indicated, this ontology describes the conceptual modeling concepts that a non-VR-specialist can use to specify the Virtual World in an intuitive way. Some of the concepts in this ontology are 'InFrontOf', 'Behind', 'LeftOf', 'InsideOf'. These are examples of high-level modeling concepts to place objects in the World. Different sets of these *spatial relations* and *reference frames* are available to allow a designer to select the relations which are most appropriate for his purpose and which are most intuitive for him. Other concepts allow to describe complex objects, to specify behavior and to express interaction.

The Domain Ontology

The Domain Ontology describes, at a conceptual level, the concepts available in the domain together with their properties and relations between them. In our current prototype, the format of this ontology is fixed, but we plan to investigate how to relax this limitation and use existing ontologies that describe the domain for which the Virtual World needs to be created. This way, our tool can reuse lots of the knowledge already created in the context of the Semantic Web.

The World Specification

The actual description of the Virtual World at a conceptual level is done using the World Specification. This is done in terms of the domain concepts by instantiating concepts from the Domain Ontology. These are the instances that will populate the Virtual World. The World Specification contains these instances, describing how they are related, where they are placed in the World, how to interact with them and what behavior they have.

2.2 Mapping stage

After the specification stage, the next thing to do is specifying how the domain concepts and the instances will be represented in the Virtual World. This is done by mapping the concepts defined in the Domain Ontology and the instances defined in the World Specification onto VR building blocks. By VR building blocks we mean VR primitives (e.g. sphere, box), constraints and so on. In this stage, also a number of (internal) ontologies and mappings are used.

The VRML Ontology

Our current prototype is using VRML as the VR implementation language. In the VRML Ontology we have defined the concepts available in VRML, like the primitive object types (e.g. sphere, box, etc.) that can be displayed using VRML as well as invisible object types like gravity. Also the necessary parameters needed in order to create instances of these object types, different constraints and connections available, etc. are defined in this ontology. This ontology is used to define the mapping from the Domain Ontology and the World Specification towards a working VRML world.

The Meta Mapping

The Meta Mapping describes how the tool keeps track of the mapping of the DAML+OIL concepts onto the VRML concepts. A mapping is defined by means of a *Source* and a *Target*. The Source concept is defined to be a Concept (from the VR Conceptual Modeling Ontology). The Target concept is defined as a VR-ObjectType (from the VRML Ontology). Both are the most top-level concepts in their respectively ontologies. This way we are able to map every VR Conceptual Modeling concept onto concepts in the VRML syntax.

The Domain Mapping

The Domain Mapping defines a default representation of the domain concepts in the Virtual World. Actually, this is an instantiation of the Meta Mapping. This mapping is introduced to avoid that the designer has to specify the same mapping for several similar instances of the same concept. Here, a default mapping for the domain concepts used can be defined. In this way, instances created for such a domain concept will have a default mapping. However, not all instances of the same concept will have exactly the same properties. To overcome this problem the default mapping can be overwritten in the *World Mapping*.

The World Mapping

Using the World Mapping we can overwrite the default mapping specified for a concept in the Domain Mapping. For instance, a default appearance may have been specified in the Domain Mapping for the concept ‘can’ (a can of soft drink). Our Virtual World may contain hundred instances of this concept. However, one of these instances can have a bit of rust on it. This implies that the appearance will differ a bit from the one used by default. To overcome this problem, the World Mapping allows that for single instances of a concept the default mapping can be overwritten.

2.3 Generation stage

The last stage is the code generation. Our experimental tool transforms information from the Domain Ontology, the World Specification and both mappings into a working Virtual World. This is done by generating VRML code. Note that we may have several variants of the tool, each generating different formats. At this moment, the tool can also generate the Virtual World in the ODE format [17]. This format is not specific for VR for the Web. In future, the tool can be extended with formats like Java3D, X3D and so on.

3. E-SHOPPING EXAMPLE

As an illustration of the VR-Wise approach, we will present an example based on a virtual e-shop. The shop contains a number of products, food and non-food. A client can walk through the shop and explore the products in 3D. Figure 2 shows a snapshot of the virtual shop.

Figure 2: example of the 3D e-shop

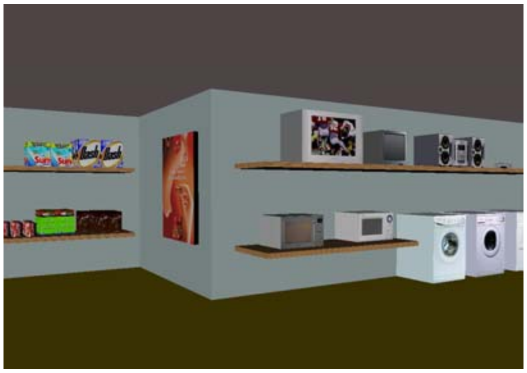
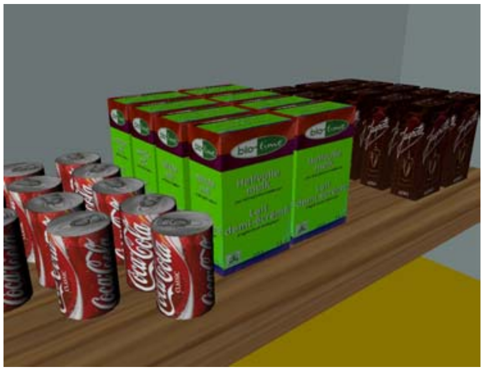


Figure 3: principle of massification applied to VR



This example also illustrates the need for the involvement of a domain expert into the development of the Virtual World. In [2], it is stated that a virtual 3D-shop makes the e-shopping experience more natural, attractive and fun for customers, but also a number of design problems are identified. One of them is the lack of experience. Design guidelines for VR stores have to be derived from real-world store design knowledge. One of the examples is massification. Massification means the display of a mass of several identical instances of the same product (e.g. cans of soft drink, packs of coffee etc.). When the number of displayed items for a product is below a given minimum, it is empirically known that there will be negative effects on sales of that product. That minimum varies for each product. The theory of massification doesn't apply to more exclusive and expensive products (e.g. television, washing machine etc.). This principle is shown in figure 3. Recent emerging 3D sites are not making use of this principle. One of the possible reasons may be that the VR developer probably was not aware of this issue. It could be easily solved by involving the domain expert more closely into the design. Our approach offers this possibility. The VR-Wise approach even goes one step further and allows the domain expert to be a virtual e-store designer.

As explained, the first stage in the VR-Wise approach is the specification stage. The user has to specify the concepts from the 3D e-shop at a conceptual level using a Domain Ontology. This is done by means of a Graphical User Interface (GUI). The following is an extraction of DAML code (generated by the GUI) that gives a part of the definition of the television concept.

```
<daml:Class rdf:ID="Television">
    <rdfs:subClassOf rdf:resource="http://wisepc.vub.ac.be/CMOntology.daml#Concept" />
    <rdfs:subClassOf>
            <daml:Restriction daml:cardinality=”1”>
                <daml:onProperty rdf:resource="#Weight"/>
                <daml:toClass rdf:resource="http://www.w3.org/2000/10/XMLSchema#decimal"/>
                <ulo:defaultValue>1.0</ulo:defaultValue>
            <daml:cardinality>1</daml:cardinality>
            </daml:Restriction>
    </rdfs:subClassOf>
    ...
```

In order to populate the Virtual World with one or more televisions, the designer will have to create the necessarily number of instances of these concepts. This is done using a GUI that will generate the World Specification. The following piece of code shows the large television(-instance) you can see in figure 2.

```
<RDFNsId2:Television rdf:ID=”28in_TV”>
    RDFNsId2:Weight='46'
    RDFNsId2:Height='59.2'
    RDFNsId2:Depth='55.3'
    RDFNsId2:Width='77.1'
    RDFNsId2:Appearance='./sony.jpeg'
</RDFNsId2:Television>
```

Now only the population (concept instances) for the Virtual World is created. In order to use these objects in the Virtual World, the designer has to place them in the world. This can be done by giving an exact position to an object or by placing objects in the world relative to other objects. Suppose the concept ‘shelve’ is defined in the Domain Ontology. Then, the designer can define an instance ‘upper_shelve’ and give it an exact position in the World. Then, he can place the large television ‘*on top of*’ the ‘upper-shelve’. This results in the following specification (extraction of the World Specification):

```
<RDFNsId2:Television rdf:about=”#28in_TV”>
<RDFNsId1:spatial-relation>
        <RDFNsId1:reference-object rdf:resource=”#upper_shelve” />
        <RDFNsId1:type rdf:resource='http://wisepc.vub.ac.be/CMOntology.daml#OnTopOf'/>
        <RDFNsId1:reference-frame
                  rdf:resource='http://wisepc.vub.ac.be/CMOntology.daml#FRF'/>
        </RDFNsId1:Spatial-Relation>
</RDFNsId2:Television>
```

The next stage in the approach is the mapping stage. Now decisions have to be made on how the objects will be represented in the Virtual World. The following piece of code illustrates this by mapping the Television concept from the Domain Ontology onto a Box representation in VRML. Note that also all attributes have to be mapped. We illustrate this by mapping the weight of a Television onto the mass of a Box. This needs also to be done for the height, depth, width and appearance attributes of the Television concept. In the current prototype, this mapping must be defined by the designer (using a GUI). Here, some help may be needed from a VR-specialist if the designer has too little experience. In Section 4 we discuss what can be done to minimize the input of the VR-specialist.

```
<RDFNsId1:ConceptMapping rdf:ID='Television_map'>
    <RDFNsId1:Source rdf:resource='file:///D:/Ontologies/shopdomain.daml#Television'/>
    <RDFNsId1:Target rdf:resource='http://wisepc.vub.ac.be/VRMLOntology.daml#Box'/>
    <RDFNsId1:Representation>
        <RDFNsId1:AttributeMapping rdf:ID='Weight_Map'>
            <RDFNsId1:Source rdf:resource='file:///D:/Ontologies/shopdomain.daml#Weight'/>
            <RDFNsId1:Target rdf:resource='http://wisepc.vub.ac.be/VRMLOntology.daml#Mass'/>
        </RDFNsId1:AttributeMapping>
    </RDFNsId1:Representation>
    ...
```

Now all necessary information is available to generate the VRML world. The result is like shown in figure 2.

4. CONCLUSIONS AND FURTHER RESEARCH

In this paper we introduced a new approach, called VR-Wise, for designing and developing Virtual Worlds. In this approach, the domain expertise is used for a faster and easier generation of the World. This approach allows modeling a Virtual World at a conceptual level and in terms of concepts from the application domain. This is realized by using a domain ontology.

Using the knowledge in the domain ontology and a set of high-level modeling primitives for VR, which have been especially developed for our approach, a non-VR-specialist can specify the Virtual World by instantiating concepts in the domain ontology and placing them in the Virtual World. Actually the complete specification can be made by a domain expert and only some minor help from a VR specialist is needed during the mapping stage when it comes to more complex Virtual Worlds. During this mapping stage, the conceptual level is mapped to the VR implementation environment, which in our case is VRML. During the last phase, the code for the Virtual World is generated. This approach gives advantages as the developed Virtual World will better fulfill the requirements of the end-user. It will also shorten the specification process. Other advantages are the possibility of fast prototyping, the exploitation of existing domain knowledge and the improved communication possibilities between the domain expert, the end-user and the VR-specialist.

To prove the feasibility of our approach and to be able to experiment with different modeling concepts we implemented an experimental tool. The Virtual Worlds that currently can be generated are rather simple and do not yet support all the latest features of VR. However, it shows that the approach is feasible for simple Virtual Worlds such as the ones currently suited for the Web. The design and implementation of this tool also has given us a lot of valuable feedback and topics for further research. More in particular, the current research concentrates on conceptual modeling primitives for behavior and interaction [9]. For instance, we plan to investigate how the real-life properties of concepts can help to infer automatically the VR-representation of the concepts. For example, if we know that a ball is spherical than we can infer that a sphere is a good representation for this concept. This would further limit the need for a VR-expert.

We believe that the approach described can open up the use of VR, and in particular on the Web, to a much broader community than nowadays. It is more intuitive and easier for a non-VR-specialist to design a Virtual World.

ACKNOWLEDGEMENT

This research is partially funded by the FWO (Fund of Scientific Research – Flanders), and by the IWT (Institute for the encouragement of Innovation through Science and Technology in Flanders) in the context of the VR-DeMo project (Virtual Reality: Conceptual Descriptions and Models for the Realization of Virtual Environments) and the OntoBasis project (Ontology-based Design of Virtual Web Environments).

REFERENCES

[1] Carey, R. and Bell, G., 1997. *The annotated VRML 2.0 Reference Manual*. Addison Wesley.

[2] Chittaro, L. and Ranon, R., 2002, New Directions for the design of Virtual Reality Interfaces to E-commerce Sites. *In Proceeding of AVI 2002.* Trento, Italy

[3] Fencott, C., 1999. Towards a Design Methodology for Virtual Environments. *Workshop on user centered design for virtual environments*. York, England

[4] Gruber, T. R., 1993. A translation approach to portable ontologies. *In Journal of Knowledge Acquisition*, Vol. 5, No. 2, pp. 199-220

[5] Guarino, N. and Giaretta, P., 1995. Ontologies and knowledge bases: towards a terminological clarification. *In Towards Very Large Knowledge Bases: Knowledge Building Knowledge Sharing*, ION Press, pp. 25-32

[6] Lassila, O. and Swick, R., 1999. Resource Description Framework: Model and Syntax Specification. *W3C Recommendation*, World Wide Web Consortium, Cambridge.

[7] McGuinness, D. L. and van Harmelen, F., 2004, OWL Web Ontology Language Overview, *W3C Recommendation 10 February 2004*. World Wide Web Consortium, Cambridge.

[8] Murdock, K. L., 2003. *3ds max 5 Bible*. Wiley Publishing, Inc., Indianapolis, Indiana.

[9] Raymaekers, C. et al., 2004. High-level Interaction Modelling to Facilitate the Development of Virtual Environments, *In Proceedings of the Virtual Reality International Conference 2004, (VRIC04)*. Laval, France

[10] Sowizral, H. et al., 2000. *Java3D API Specification*. Addison Wesley.

[11] Tanriverdi, V. and Jacob, R.J.K., 2001. A Design Model and Methodology for Developing Virtual Reality Interfaces. *In Proceedings of ACM Symposium on Virtual Reality Software and Technology 2004*. Alberta, Canada, pp. 175-182

[12] Walsh, A.E. and Sévenier, M., 2001. *Core Web3D*. Prentice Hall, Upper Saddle River, USA.

[13] Willans, J. S., Smith, S.P. and Harrison, M. D., 2001. A toolset supported approach for designing and testing virtual environment interaction techniques. *In International Journal of Human-Computer Studies,* Vol. 55, No. 2, pp. 145-165.

[14] http://www.unrealtournament.com, Accessed 13 April 2004

[15] http://www.sgi.com/software/cosmo/worlds.html, Accessed 12 April 2004

[16] http://www.daml.org/2000/12/daml+oil.daml, Accessed 13 April 2004

[17] http://opende.sourceforge.net/, Accessed 14 April 2004