

ONTOLOGY: A SUPPORT STRUCTURE FOR A V-LABS NETWORK, THE EURONET - LAB.

Raúl Cordeiro Correia
Escola Superior de Tecnologia
de Setúbal
Instituto Politécnico de Setúbal
raul.correia@estsetubal.ips.pt

José Manuel Fonseca
Faculdade de Ciências e Tecnologia
Universidade Nova de Lisboa
Caparica, Portugal
jmf@uninova.pt

Andrew Donnellan
IT Tallaght, Dublin
Ireland
andrew.donnellan@ittdublin.ie

Abstract — This paper presents work on the establishment of a virtual laboratories network, based on the concept of Virtual Closet that will contain all the elements and parts required to build the different experiences available in a v-labs network (that we call Euronet-Lab).

To build this complex network it is necessary to find an adequate support. The large database of v-labs and independent elements that can integrate this structure allows the “recycling” of elements for purposes diverse from those they were originally created. This means to “re-use” the same element several times in different scenarios. In order to do this, it is necessary to have a structure that allows multiple instances of the same component.

This paper will discuss the ontology essential to the establishment of such a network in order to standardize concepts and create uniform components description that supports the interaction between heterogeneous systems using shared parts or circuits.

Index Terms— V-labs, Virtual Labs, E-learning, CMS, Ontology, Remote Laboratory.

I. INTRODUCTION

The first requirement for the establishment of an open network of heterogeneous virtual laboratories is the definition of an information structure capable of describing its different components without ambiguity.

This structure must support the classification of all the components and parts used in experiences, and allow to get these elements from a “Virtual Closet”. This must be build and defined, prior to anything else. We defend that an Ontology, that defines the concepts and standardizes the nomenclature, is essential.

The “Virtual Closets” will be the base for the construction of experiments on our system. The structure that supports these “closets” must allow the definition of multiple instances of the same object to be used simultaneously in different experiments.

The work already done to build this system was the project and drawing of the system, and also the definition of the main branches and elements of the ontology to be used.

The *Protégé Software* was used to draw the structure of our ontology. All the hierarchy, including super-classes and sub-classes, is already built. The concepts for each class and the properties of its concepts (also called “restrictions”) are already defined.

The contribution of this paper is to explain and justify why, in our opinion, the better solution to implement this structure is an Ontology, and also to define, and build all the objects and elements that we need in our system.

The paper starts by making an introduction, where it is explained how to build an ontology, and what are the rules and steps that we must follow to correctly build the ontology that it is proposed to implement. Then we present and defend our solution to the Euronet-Lab ontology, and we finalize the paper presenting the direction to the development of future works and the implementation of a experimental prototype to the system.

II. DEFINING AN ONTOLOGY

An Ontology is defined as:

“An Ontology defines a common vocabulary for researches and someone who need to share information in a domain”. Ref. [1]

If necessary, an ontology can include machine-interpretable definitions of basic concepts in the domain as well as relations among them. It is also possible to define a common vocabulary for several users who needs to share information in a domain. An Ontology is therefore essential to share information among heterogeneous users or software agents that share the same system and structure.

The main advantages of using an ontology are to:

- Share common understanding of the information structure users or software agents
- Enable the reuse of domain knowledge
- Make domain assumptions explicit
- Separate domain knowledge from the operation knowledge
- Analyze domain knowledge

Considering our goals, an ontology can be seen as a formal explicit description of concepts in a domain of discourse. The main elements of an ontology are:

- Classes - sometimes called concepts
- Slots - properties of each concept describing various features and attributes of the concept. Sometimes called roles or properties
- Facets - restrictions on slots, or even properties of slots, or restrictions of slots

The Ontology, together with a set of individual instances of classes, constitutes a **knowledge base**. In reality, there is a fine line where the ontology ends and the knowledge base begins. Developing an Ontology includes:

- Defining the classes of the Ontology
- Arranging the classes in a taxonomic (sub-class – superclass) hierarchy
- Defining slots and describing allowed values for these slots
- Filling in the values for slots for instances

We can then create a knowledge base by defining individual instances of each classes filling in specific slot value information and additional slot restrictions.

To design correctly an ontology we must respect the following rules. These rules may seem rather dogmatic, unless we have a strong and rational base structure, but they can help to make correct design decisions in most of the cases where ontologies can be applied:

1. There is no one correct way to model a domain of knowledge – there are always several alternatives. The best way to implement our ontology depends on the application that we have in hands, and all the extensions of it that was possible to us to anticipate.
2. Continuous ontology development process is necessary, and is an iterative process
3. Concepts in ontology should be very close to objects (physical or logic) and also close from the relationship that exist in the domain where we define the ontology.
4. Probably the most common is to define nouns (objects) or verbs (relationships) in sentences that describe the domain.

In a most detailed way, we can say that there are some steps that we must follow to define our ontology:

Step n. 1: Determine the domain and scope of our ontology.

To do this we must essentially to respond to the following questions:

- What is the domain that the ontology will cover ?
- For what we going to use the ontology ?
- For what types of questions the information on the ontology should provide answers?
- Who will use and maintain the ontology ?

Step n. 2: Consider reusing existing ontologies:

This involves a detailed analysis of existing ontologies, for the purpose of refining and extending existing sources for our particular domain and task. Reusing existing ontologies may be a requirement if our system needs to interact with other applications that have already committed to particular ontologies or controlled vocabularies.

Step n. 3: Enumerate important terms in the Ontology:

It is useful it write down a list of all terms we would like either to make statements about or to explain to a user.

Step n. 4: Define the classes and the class hierarchy:

These are several possible approaches in developing a class hierarchy:

- Top-down development process:
Starts with the definitions of the most general concepts in the domain and subsequent specialization of concepts.
- Bottom-up development process:
Starts with the definition of the most specific classes, the leaves of the hierarchy, with subsequent growing of these classes in more general concepts.
- Combination development process: This is really a combination of top-down and bottom-up approaches: we define the most salient concepts first and then generalize and specialize them appropriately.
None of these three methods is inherently better than any of the others. The approach to take depends strongly on the personal view of the domain and the situation in particular. If a developer has a personal top-down view of the domain, then it may be easier to use the top-down approach. However, the combination approach is often the easiest way for many ontology developers, since the concepts “in the middle” tend to be the more descriptive concepts in the domain [2]

Step n. 5: Define the properties of class-slots:

The classes done will not provide enough information to answer the competency questions.

Once we have defined some of the classes, we must describe the internal structure of concepts.

In general, there are several types of object properties that can become slots in an ontology:

If we take as example a Ontology about wines, we must consider the following:

- “Intrinsic” properties (ex: flavor of a wine)
- “Extrinsic” properties (Name of the wine and area of production)
- “Parts” if the object is structures, these can be both physical and abstract “parts” (Indicated dishes to drink with)
- “Relationships to other individuals,” these are relationships between individual members of the class and other items (maker of the wine, type of grape)

Note: All the subclasses of a class inherit the slot of a class.

Step n. 6: Define the facets of the slots:

Slots can have different facets describing the value type, allowed values, the number of values (cardinality), and other features of the value the slot can take.

Some normal common facets are:

Slot Cardinality: Defines how many values a slot can have some details:

- Some systems distinguish between single cardinality (Allowing at least one value) and multiple cardinality (allowing any number of values)
- Some systems allow the specification of a maximum and a minimum cardinality to describe the number of slots more that a slot must have at least N Values.

Slot value Type, they have some possible types that corresponds to the common variable data types:

- String
- Number (Float or integer)
- Boolean (yes-no flag`s)
- Enumerated (list of specific allowed values)
- Instance (instance-type slots allows the definition of relationship between individual.
- Instance must also define a list of allowed classes from which the instances can come.

The classes to which a slot is attached or the classes which property a slot describes, are called the **domain of the slot**.

We can define the **range of a slot** as the allowed classes for slots of type instance.

Some systems allow restricting the range of a slot when the slot is attached to a particular class.

Step n. 7: Create instances:

The last step to create an ontology is creating individual instances of classes in the hierarchy.

To accomplish this step we must do the following “sub-steps”:

1. Choosing a class
2. Creating an individual instance of that class
3. Filling the slot values

To develop a model that is accepted by the end users, it is essential for the model to be easily understood and well structured and to provide an adequate representation of the real world situation. Ref [1]

However several technical problems stand in the way of shared and reusable knowledge. Ref. [2]

Knowledge base systems impose special requirements to complete interoperability, to communicate.

They must use statements always expressed in a formal knowledge representation, like happens in the specification of environments by ontologies. Also they use queries and must receive answers. They take also “background knowledge” as input data.

For this high level of communication using ontology is necessary to define and respect convention at three levels:

1. Representation language format
2. Agent communication protocol
3. Specification of the content of shared knowledge.

These specifications are also assumptions and models about a world and environment to describe, so in this case ontologies can play a very important role in software specification[3].

One of the objectives of this work is to build and define libraries of reusable knowledge components, (like RLO reusable learning objects in SCORM specification) and also Knowledge – based services that can be invoked over networks; to achieve this objective the most indicated structure to define and describe all the elements of a virtual laboratory as parts and components is an ontology, by the above reasons exposed.

This paper presents the way how an Ontology can be used to describe the environment of a virtual laboratory, very specifically a “virtual-closet”, where all the parts, components, and elements of a virtual laboratory can be “stored” , used and re-used when necessary for the various experiments to develop.

Also an ontology allows that several equal elements are used in several and different experiences. So the best way to achieve this purpose is to find a structure that allows to define several instances of the same component or element.

Also we can say that an ontology permit to describe in a very detailed way the components / elements of the “virtual closet”, with all its details and features.

Thus according to these reasons referred above we think that an Ontology is the best solution to our problem of describing all the elements of a virtual laboratory.

But we need also to build a strong and very consistent conceptualization, that is a way of formally represent knowledge in this specific area.

A conceptualization is a structure of organized knowledge where the objects, concepts and other entities are described.

A conceptualization also describes the relationships between the different objects and components and how it is possible to define the details of that relationship.

If an object or entity has some data fields with specific characteristics, the conceptualization also indicates us what are the fields that allow us to establish this relationship. So we can say that a conceptualization is an abstract, simplified view of the world and environment that we want to represent for some purpose.

So we can consider that “An Ontology is an explicit specification of a conceptualization”. Ref [3]

The set of objects described by the Ontology, and its relationships, are reflected in the vocabulary tree, in the same way that a knowledge-base program represents knowledge.

In an Ontology the definitions that its contents include, associate the names of entities in its universe (classes, relations, functions, or any kind of object) with a description of what these names mean, with the formal axioms that constrain the interpretation of these terms in the present context and environment.

So formally we can say that an Ontology is the statement of a logical theory.

In a pragmatic way we can say that an Ontology defines the vocabulary with which queries and assertions are exchanged between systems that communicate in the v-labs network.

Ontological commitments constitute agreements that should be used as shared vocabulary in a coherent and consistent way.

Design criteria for ontologies:

Formal ontologies are designed when we choose how to represent something in an ontology, and in this act we are making design decisions.

If we want to have an ontology with the purpose of knowledge sharing and interoperability among programs and systems based on a shared conceptualization it is

necessary to follow and improve the following rules, to build our ontology:

1. **Clarity:** the definitions should be very objective because an ontology should effectively communicate very clearly the intended meaning of defined terms.

The definitions should be independent of social and computational context.

All the definitions should be complete definitions, every predicate should be defined with all the necessary and sufficient conditions. All the definitions should be documented with natural language.

2. **Coherence:** An Ontology should be the most coherent possible, this means that an ontology should sanction inferences that are not consistent with the definitions.
3. **Extendibility:** An ontology should be designed to anticipate the uses of the shared vocabulary. This ontology should allow define new terms for special uses based on existing vocabulary, in a way that does not require the revision of the existing definitions.
4. **Minimal Encoding:** Bias should be minimized, because knowledge-sharing systems over the network may be implemented in different representation systems and styles.
5. **Minimal Ontological Commitment:** An ontology should make as few claims as possible about being modeled, allowing the parties committed to the ontology freedom to specialize and instantiate the ontology as needed.

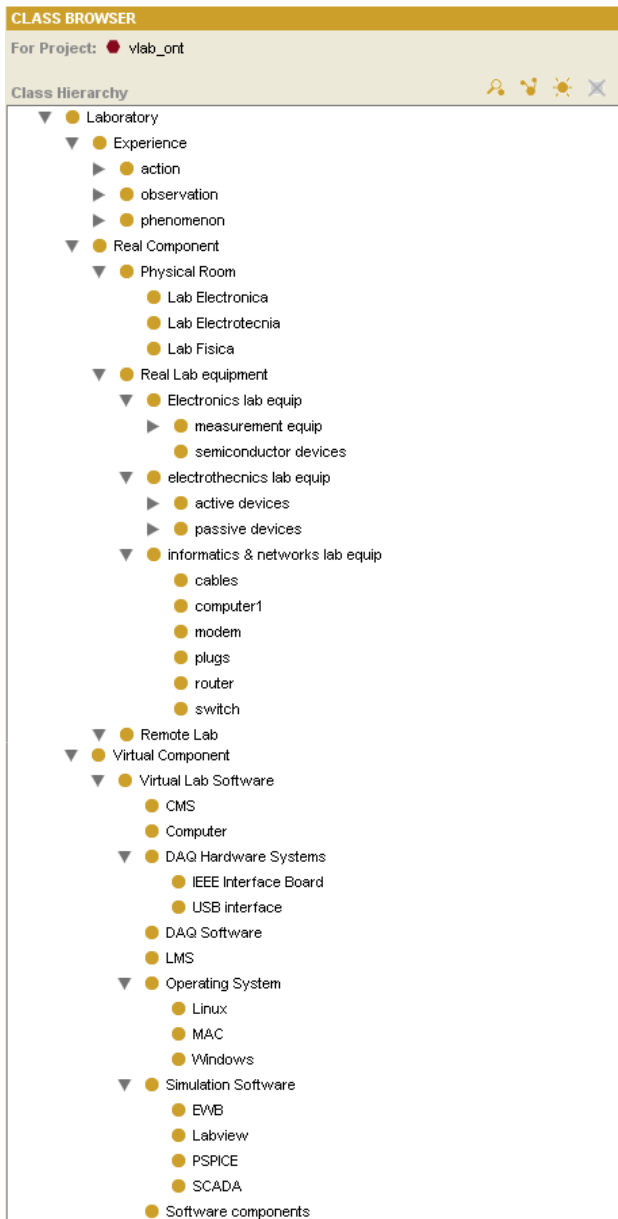
In conclusion, When designing ontologies, it should be noted, that “an ontology is a particular system of categories accounting for a certain vision of the world. This system does not depend of a particular language. A shared ontology need only describe a vocabulary for talking about a domain, whereas a knowledge base may include the knowledge needed to solve a problem or answer arbitrary about a domain”.

The Proposed Ontology:

To implement and build the ontology the software “Protégé” is used. It is a tool specially developed to build ontologies.

Protégé is a software that allows easily to build ontologies respecting all the rules that we define in our system.

In Protégé we define what will be the classes and also we can define all the relations between them. In our particular case the “root” or Master-class of our ontology is “LABORATORY”:



From there we define three main classes:

- Experiment
- Real Component
- Virtual Component

From these three main classes we define several sub-classes.

In particular for “Experiment”, we consider three sub-classes: phenomena, action and observation. All this are actions that happens normally in real laboratories.

Phenomena is something that occurs naturally in a laboratory and can be divided in several types of phenomena: Electric, Electronic, electrostatic. Theoretically, a phenomena will occur as consequence of some action over the system.

Essentially the ontologies used in this field of knowledge using semantic web technologies.

We can define, according to the W3C, "The Semantic Web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries."

Tim Berners-Lee defines the Semantic Web as "a web of data that can be processed directly and indirectly by machines."

One of the first and most difficult steps to start using Semantic Web technologies is to design a domain specific ontology. An ontology is a generalized formal representation of the domain. This model that we purpose consists of several properties and vocabularies representing different relationships between different types of resources in a generic model. Several layers of the Semantic Web stack provide a potential framework to model a domain specific ontology. Ref. [1]

To develop a model, which is acceptable to the end users, it is essential that the model can be easily understood and well structured in way to provide an adequate representation to the real world situation.

For defining a new model, several key issues must be taken in consideration, including the effective use of existing and well-established standards that might fit the domain specific requirements.

In our study we can consider the following classification of laboratories:

- Remote Laboratory: A remote laboratory is an online laboratory that provides real experiments. This definition implies the control of real hardware and the conduction of real measurements remotely.
- Virtual Laboratory: A virtual laboratory is an online laboratory that provides software simulations or applications.
- Hybrid Laboratory: A hybrid laboratory is an online laboratory that combines virtual laboratory and remote laboratory technologies. It also provides real hardware experiments and software simulations.

All the laboratories are composed by experiments, that we can define as:

“An experiment is the smallest enclosed unit of an online laboratory. It provides also the execution of virtual or real experiments to observe the behavior and output of a

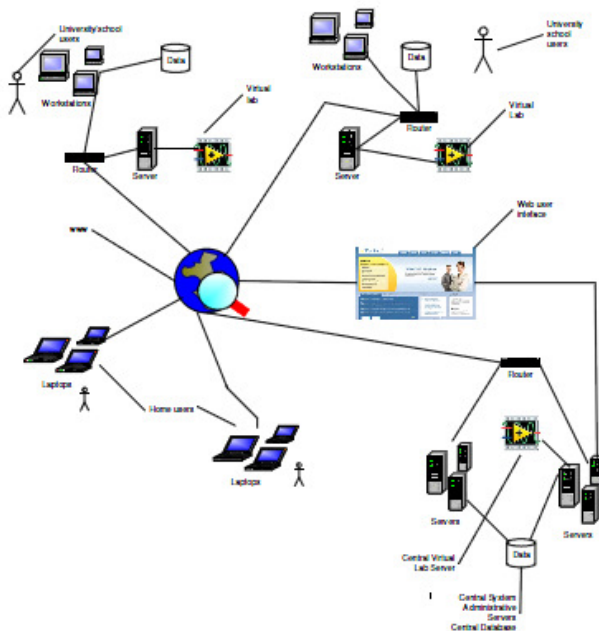
system. An online laboratory consists of one or more experiments in different fields of science and engineering. Ref. [1]

Also normally, associated to an online laboratory, there must be provided to the students, other learning resources like a laboratory tutorial and lecture notes to provide the theoretical background necessary to carry out an experiment. Therefore, it is necessary to provide a variety of additional documents and references. Ref. [1]

So is very important to have as support of all this system. A CMS or a e-learning platform that interconnect the build ontology with the v-labs proposed in the “virtual closet”.

The CMS , LMS or a e-learning platform allows the existence and organization of all this pedagogical and technical pedagogical contents and supports.

The proposed network as the following block-diagram:



In this network we have three main actors:

- Teachers
- Students and Researchers
- Administrative staff

A login interface, forms and database must be created using ASP or PHP that allows students to:

- Schedule an experiment in a certain lab;
- Verify available labs and in which universities or institutes are located;
- Verify the experiments they have done, their grade and comments from their teachers;

- Read or review pedagogical contents that support the different experiments.

The teachers should be able to:

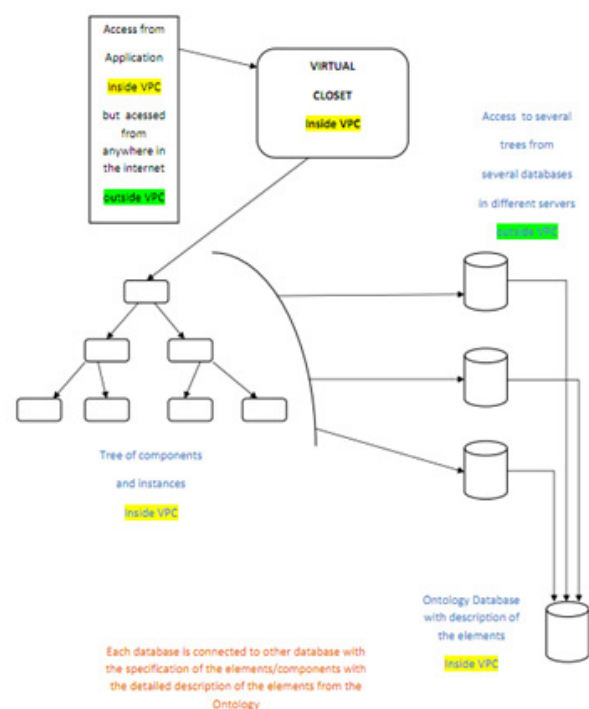
- Send pedagogical contents;
- Review contents and materials;
- Evaluate the students' Works;
- Communicate with the students using email, chat, video-conference in order to give orientations and clarify subjects.

The administrative staff should support all the non-technical issues and administrative issues derived from the communication between universities, institutions, teachers, researchers and students.

This is the way that we think that this V-labs network , the “Euronet – Lab” should work using is on Ontology. Ref. [5,6].

FUTURE DEVELOPMENTS

The following action to take in this project is to build a prototype system that interconnects the build ontology with the v-labs network and the LMS databases that support the administrative parts of the system as shown in the next figure:



ACKNOWLEDGMENTS

Instituto Politécnico de Setúbal – Escola Superior de Tecnologia de Setúbal

REFERENCES

- [1] Abul K.M. Azad et al., *Internet Accessible Remote Laboratories: Scalable E-Learning Tools for Engineering and Science Disciplines*, Engineering Science Reference (an imprint of IGI Global) E-mail: cust@igi-global.com Web site: <http://www.igi-global.com>.
- [2] Gruber, T.R., 1993a. Toward Principles for the Design of Ontologies Used for Knowledge Sharing N. Guarino & R. Poli, eds. *International Journal of Human-Computer Studies*, 43(5-6), pp.907-928. Available at: <http://linkinghub.elsevier.com/retrieve/doi/10.1006/ijhc.1995.1081>.
- [3] Gruber, T.R., 1993b. Toward Principles for the Design of Ontologies Used for Knowledge Sharing N. Guarino & R. Poli, eds. *International Journal of Human-Computer Studies*, 43(5-6), pp.907-928. Available at: <http://linkinghub.elsevier.com/retrieve/doi/10.1006/ijhc.1995.1081>.
- [4] Noy, N.F. & McGuinness, D.L., 2000. Ontology Development 101□: A Guide to Creating Your First Ontology. *Development*, 32(1), pp.1-25. Available at: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.136.5085&rep=rep1&type=pdf>.
- [5] Cordeiro, R., Fonseca, J. M., & Donellan, A. (n.d.). A Cloud Based Laboratory Environment, pp 33-37 short papers section, CISTI 2010, Santiago de Compostela, Spain.
- [6] Cordeiro R., Passos, H., ed. *Virtual Labs in the E-Learning Context as Tools of Collaboration Work*.Session: Virtual Learning Environments (VLEs) & online/virtual labs, EDULEARN 09. 2009: Barcelona, Spain.

AUTHORS

Raúl Cordeiro Correia is with the Instituto Politécnico de Setúbal – Escola Superior de Tecnologia de Setúbal, Largo Defensores da Republica nº 1, 2910-470 Setúbal, PORTUGAL
Email:raul.correia@estsetubal.ips.pt / cinel.raul@gmail.com

José Manuel Fonseca is Professor in the Electrical Engineering Department of the Universidade Nova de Lisboa, Campus da FCT/UNL, 2829-516 Caparica, Portugal
Email: jmrff@fct.unl.pt

Andrew Donnellan is a lecturer in the Department of Electronic Engineering, Institute of Technology, Tallaght, Dublin 24, Ireland.
Email: andrew.donnellan@ittdublin.ie

This work was partially supported by IPS – Instituto Politécnico de Setúbal – Escola Superior de Tecnologia de Setúbal