Open access • Book Chapter • DOI:10.1007/3-540-48775-1_9

Ontological Reengineering for Reuse — Source link

Asunción Gómez-Pérez, Dolores Rojas-Amaya

Institutions: Technical University of Madrid

Published on: 26 May 1999 - Knowledge Acquisition, Modeling and Management

Topics: Ontology (information science), Process ontology, Upper ontology, Ontology-based data integration and Suggested Upper Merged Ontology

Ontological Reengineering for Reuse

Asunción Gómez-Pérez, Mª Dolores Rojas-Amaya
Facultad de Informática
Universidad Politécnica de Madrid
Campus de Montegancedo s/n
Boadilla del Monte, 28660. Madrid. Spain.
Tel: (34-1) 336-74-39, Fax: (34-1) 336-74-12
Email: {asun, mrojas}@delicias.dia.fi.upm.es

Abstract

This paper presents the concept of *Ontological Reengineering* as the process of retrieving and transforming a conceptual model of an existing and implemented ontology into a new, more correct and more complete conceptual model which is reimplemented. Three activities have been identified in this process: reverse engineering, restructuring and forward engineering. The aim of *Reverse Engineering* is to output a possible conceptual model on the basis of the code in which the ontology is implemented. The goal of *Restructuring* is to reorganize this initial conceptual model into a new conceptual model, which is built bearing in mind the use of the restructured ontology by the ontology/application that reuses it. Finally, the objective of *Forward Engineering* is output a new implementation of the ontology. The paper also discusses how the ontological reengineering process has been applied to the `Standard-Units` ontology [18], which is included in a `Chemical-Elements` [12] ontology. These two ontologies will be included in a `Monatomic-Ions` and `Environmental-Pollutants` ontologies.

1 Introduction

The concept of reengineering is commonly used in Software Engineering and started to move into the field of Knowledge Engineering a few years ago. When we try to define the term reengineering, other very closely related concepts emerge, such as reverse engineering, restructuring and forward engineering. The term *reverse engineering* is used to denote the process of analyzing a system to identify its components and relations and/or represent a system in another manner [5]. Therefore, the reverse engineering process could be defined as the analysis of a system/program in an attempt to create a representation of the program at a higher level of abstraction than source code. This is what Pressman refers to as design retrieval [20]. There are several definitions of the term *restructuring* [6, 8, 21]. The most representative definition was made by Chikofsky who defined restructuring as system transformation to pass from one representation to another at the same level of abstraction, conserving functionality and semantics [8]. He also defines the concept of *forward engineering* as the traditional process leading from a high level of abstraction, which is independent of implementation design, towards the physical implementation of a system [8]. Accordingly, the term *reengineering* refers to the process in which design information about the existing software is retrieved and this information is then used to alter or reconstruct the existing system in an attempt to improve overall quality [20]. The software outputted by reengineering mostly reimplements the function of the existing system; however, at the same time, the developer adds new functions and/or improves overall performance.

D. Fensel and R. Studer (Eds.): EKAW'99, LNAI 1621, pp. 139-156, 1999.

There are no papers on reengineering related to the field of ontological engineering, although the paper by Barley et al. [3], who show how knowledge on stillended panel layout implemented in ICAD code has been manually analyzed and transformed into production rules, which have been formalized in KIF [13] and SLANG [19], could be construed as a kind of reengineering. So, this paper presents how we have done ontological reengineering of the `Standard-Units` [18] ontology, which is included in a `Chemicals-Elements` [12] ontology. Both ontologies are reused by a `Environmental-Pollutants` ontology. This paper is organized as follows: Section 2 presents the need for environmental ontologies and section 3 the scope of the problem; sections 4 and 5 describe the ontological reengineering method applied to the `Standard-Units` ontology, and, finally, section 6 reviews the `Chemical-Elements` ontology.

2 Need for environmental ontologies

Specialists from different fields, such as biologists, geologists, computer scientists, chemists, lawyers, etc., are involved in the environmental sciences. Each expert uses his own vocabulary, there being no common terminology or standard to ensure that each term is used accurately. There are numerous reasons for building ontologies in the environmental field: (1) The existence of synonyms (for example, the terms "contamination" and "pollution" are used as synonyms in reference to air pollution, as are "bleaching" and "leaching" in the case of soil treatment and problems); (2) One term can be used in different sciences, where it may have a similar but not an identical meaning (for example, in the geological domain, the word "contamination" refers to the process in which the chemical composition of the magma changes due to the assimilation of rocks and, in the microbiological domain, it is defined as the biological process of bacterial alteration); and (3) there are terms that are closely related within the same science that present slight differences of meaning, (for example, within the biological sciences, "contamination" is the term used in microbiology and "pollution" is the term used in ecology).

There are a lot of possibilities for building environment-related ontologies, but we are going to center on environmental pollutants ontologies. An ontology of this type has to study the methods of detecting the different pollutants components of various media: water, air, soil, etc., and the maximum permitted concentrations of these components, taking into account all the legislation in effect (European Union regulations, Spanish, German, US legislation, etc.). Moreover, the elements that are part of compounds are ionic. Ions are, therefore, the entities to be considered when performing environmental-pollutants-related studies, as they are possible indicators of pollution, deterioration, etc. Previous knowledge about elements in their pure state and their properties, as well as the units of measure of some properties, are required to represent knowledge about ionic concentration. The `Environmental-Pollutants` ontology seeks to produce a unified, complete and consistent terminology that can be used consistently, precisely, unambiguously and concisely in environmental applications that employ the maximum permitted concentration of ions to detect alterations in such media.

3 Problem scope

Before developing the ontologies on monatomic-ions and environmental-pollutants, we looked for other ontologies that had already been developed to check whether any of the knowledge they contain could be reused. We looked for ontologies related to periodic system elements and containing Système International (SI) units of measure. Accordingly, we searched the ontologies in the Ontology Server[1] [9] and the Cyc[2] ontology server. Useful ontologies, like `Standard-Units`[3], which defines a series of base units of measure, and `Chemical-Elements`, which defines the chemical elements of the periodic system, were found at the Ontology Server. Definitions of some units of measure and chemical entities (atom, ion, molecule and radical) were found at the Cyc server. As the ontology describing the units of measure at the Ontology Server includes a natural language definition, physical dimension and factors of conversion to other units of the same dimension for each unit and Cyc´s ontologies only include a natural language definition, we decided to use the Ontology Server ontology as it was more complete. Moreover, as `Chemical-Elements` was developed by our work group, we opted to take the Ontology Server ontologies as a starting point, using the Cyc ontologies as a reference point. Then, we evaluated (verify[4] and validate[5]) the `Chemical-Elements` and `Standard-Units` ontologies to assure that they were correct and complete and thus guarantee that these ontologies provided a solid basis on which new ontologies could be developed incrementally. These ontologies were analyzed bearing in mind its future use of this ontology by the `Monatomic-Ions` ontology. The initial analysis of `Chemical-Elements` revealed that: (1) it addresses the elements in their pure state, (2) it needs to be updated with new knowledge that addresses elements from the environmental viewpoint, and (3) it includes attributes (i.e., atomic-weight) that have associated SI units of measure.

In pursuit of the above-mentioned objectives, we are going to develop a new ontology on `Monatomic-Ions` which will be later included in an ontology on `Environmental-Pollutants`.

The starting point of the new ontology will be the monatomic ion, both anionic and cationic, addressed from the viewpoint of inorganic chemistry and, also, analyzed with a view to standardization in the soil and waterfields within the physical environment and in terms of human health. On the other hand, as the ontology under development covers such an extensive field, the development of an ontology of polyatomic ions has been postponed. Figure 1 shows how all these

[1] http://www-ksl.stanford.edu:5915 and its european mirror site at http://www-ksl-svc-lia.dia.upm.es:5915.

[2] http://www.cyc.com.

[3] The Standard-Units used to develop this work was available at the Ontology Server in December 1997.

[4] Verification concerns with analyzing the completeness, consistency, conciseness, expandability, and robustness of the definitions and axioms that are explicitly stated in the ontology, and the inferences that can be drawn from those axioms [16].

[5] Validation refers to whether the meaning of the ontology definitions really represent the real world for which the ontology was created [16].

ontologies will be integrated in a hierarchical and distributed architecture. The ontologies at the top of this hierarchy should be interpreted as including the lower-level ontologies. Note that this hierarchical architecture facilitates ontology design, maintenance and understanding by the future user. The description of the `Monatomic-Ions` and `Environmental-Pollutants` ontologies are out of the scope of this paper.

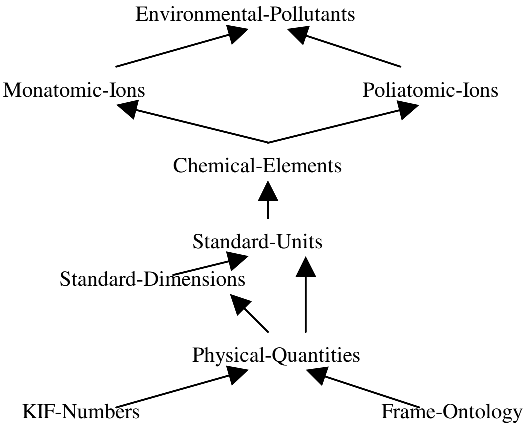

Figure 1. Relationship between the ontologies involved.

4 Ontological reengineering: method

The method for reviewing the `Standard-Units` ontology at the knowledge level is presented in Figure 2 and adapts Chikofsky's software reengineering schema [8] to the ontology domain. In this paper, we define *ontological reengineering* as "the process of retrieving and transforming a conceptual model of an existing and implemented ontology into a new, more correct and complete conceptual model, which is reimplemented". The ontological reengineering process should be carried out bearing in mind the use of the existing ontology by the system (ontology/software) that reuses it. Therefore, several ontological reengineering processes could be performed on the same ontology. If this were the case, configuration management would be required to keep a record of ontology evolution, as would strict change control.

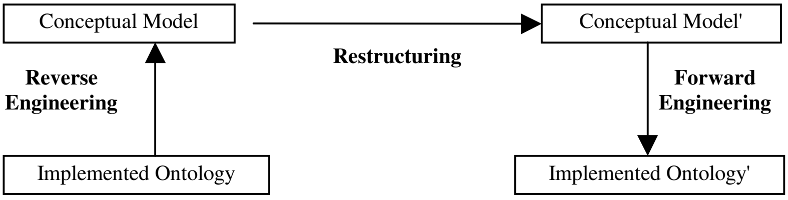

Figure 2. Ontological reengineering process.

Three activities were identified in the ontological reengineering process: reverse engineering, restructuring and forward engineering. Figure 3 pictures an organizational chart showing the activities performed during the reengineering process and the documents generated in each step.

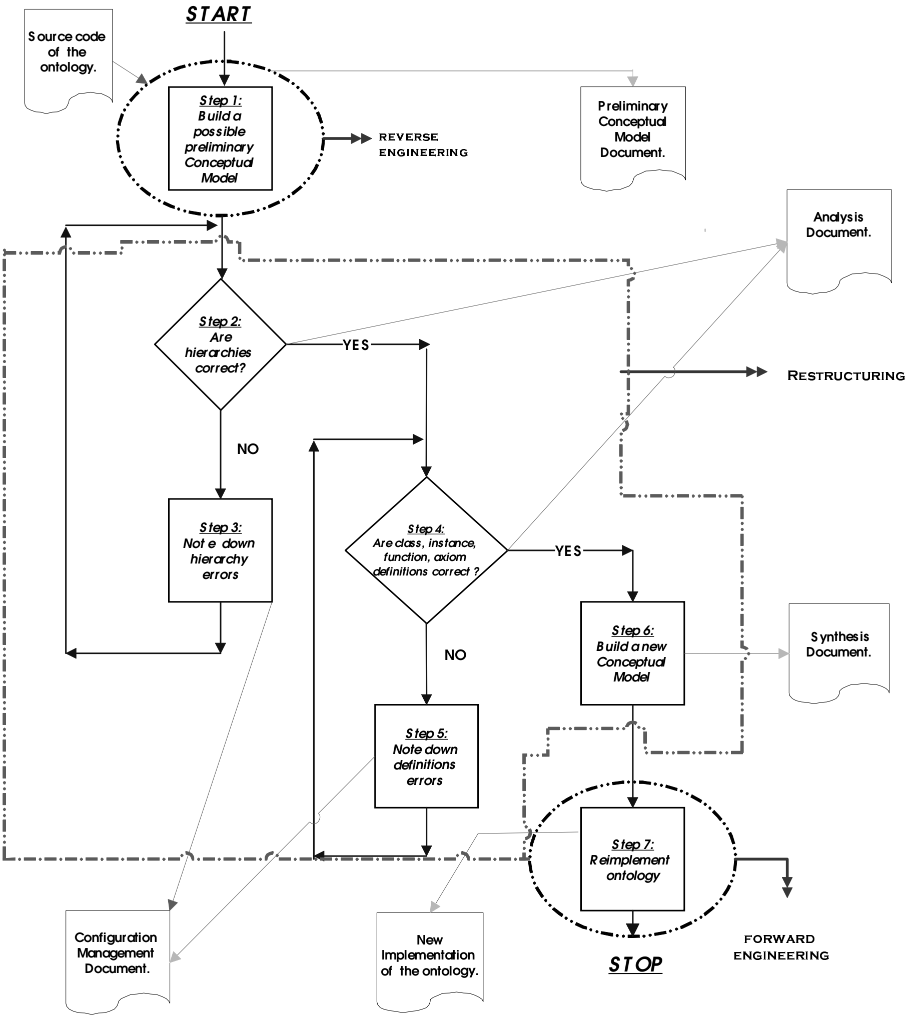
START
Source code of the ontology.
Step 1: Build a possible preliminary Conceptual Model
REVERSE ENGINEERING
Preliminary Conceptual Model Document.
Analysis Document.
Step 2: Are hierarchies correct?
YES
NO
RESTRUCTURING
Step 3: Note down hierarchy errors
Step 4: Are class, instance, function, axiom definitions correct ?
YES
NO
Step 6: Build a new Conceptual Model
Synthesis Document.
Step 5: Note down definitions errors
Step 7: Reimplement ontology
FORWARD ENGINEERING
Configuration Management Document.
New Implementation of the ontology.
STOP

Figure 3. Ontological Reengineering activities.

Reverse Engineering**:** Its objetive is to output a possible conceptual model on the basis of the code in which the ontology is implemented. For the purpose of building a conceptual model, the set of intermediate representations proposed by the methodology named METHONTOLOGY [11, 12, 15] are used.

Step 1: Draw the hierarchies and taxonomic relations between concepts and instances, "ad hoc" relations between concepts, instances and between concepts and instances of the same or another hierarchy. Identify the functions and axioms of the ontology. Generate a document reflecting the preliminary conceptual model outputted by this step.

Restructuring**:** Its objective is to correct and reorganize the knowledge contained in an initial conceptual model, and detect missing knowledge. This restructuring is guided by the ontology that is to reuse the knowledge, which means that there is no way of assuring that the restructured ontology will be a hundred per cent valid for ontologies that reuse the restructured knowledge. We distinguish two phases: analysis and synthesis. The analysis phase goal (steps 2 to 5 of figure 3) is to evaluate the ontology technically [14], that is, to check that the hierarchy of the ontology and its classes, instances, relations and functions are complete (contain all the definitions required for the domain of chemical substances), consistent (there are no contradictions in the ontology and with respect to the knowledge sources used), concise (there are no explicit and implicit redundancies) and syntactically correct. The synthesis phase (step 6 of figure 3) seeks to correct the ontology after the analysis phase and document any changes made.

Step 2: Check the correctness and completeness of each hierarchy [14]. Analyze: a) whether the taxonomic relations between concepts are correct; b) whether the concepts present in the original hierarchy should be further specified or generalized; c) that all the concepts/instances required appear in the original hierarchy; d) if necessary, add/delete from the original ontology any concept/instance.

Step 3: Note down the errors. This will allow change control to be performed as part of configuration management process.

Step 4: Having checked that the hierarchies are correct, analyze the correctness and completeness of the definitions of classes, instances, properties, relations, functions and axioms. The ontologist will analyze the initial conceptual model attached to the code in which the ontology is implemented. Specialized material for this purpose (such as books, dictionaries, handbooks, etc.) will be required, as will the help of an expert in the domain defined in the ontology.

Step 5: Note down the errors detected in step 4 in order to perform change control as part of configuration management process.

Step 6: Having reviewed and corrected an original conceptual model, design a new conceptual model including all the above-mentioned changes, building the correct and complet hierarchies and outputting the correct and complete definitions for their later implementation. The ontologist will draw up a synthesis document specifying the actions carried out and the design criteria governing restructuring.

A series of documents will be generated, which can be divided into three groups: (1) analysis document, including a list of anomalies (problems, errors, omissions, ambiguities, etc.) encountered and detected in steps 2 and 4; (2) configuration management document, which includes reports related to the changes made in steps 3 and 5 on the basis of the set of errors identified in the analysis document. This document includes: description, need and effects of the change, possible alternatives, justification of the selected alternative, date of the change, etc.; and (3) synthesis documents, including the actions taken and criteria observed in step 6.

Forward Engineering**:** The objective of this step is to output a new implementation of the ontology on the basis of the new conceptual model.

Step 7: Reimplement the ontology on the basis of the new conceptual model, including all the recorded changes. This will output a document containing the code of the new ontology implementation.

The proposed work method is a sound initial approach to carrying out the above-mentioned process, although it could be improved in later studies using more complex ontologies. In order to increase the reusability of the ontology to be reengineered, guidelines and criteria to achieve a higher degree of reusability are needed in the restructuring process. Other open issue regards the relationship between the ontology that is being reengineered and top-level ontologies, if any.

5 A case study: Reengineering `Standard-Units`

5.1 The need of reviewing `Standard Units`

The `Standard-Units` ontology defines a series of SI units of measurement and other commonly used units that do not belong to the SI units. It includes the `Standard-Dimensions` ontology, which defines a series of physical dimensions (i.e., mass, time, length, temperature and electrical current) for different quantities. It also includes other dimensions, derived from the above five, including pressure, volume, etc. Depending on the system of units used, the physical quantities defined at the `Standard-Dimensions` ontology can be expressed in different units using the vocabulary of the `Standard-Units` ontology; for example, length can be expressed in meters, miles, inches, etc. Both the `Standard-Units` and the `Standard-Dimensions` ontologies include `Physical-Quantities` (see Figure 1), which defines the basic vocabulary for describing physical quantities in a general form, making explicit the relationship between quantities of various orders, units of measure and physical dimensions. A quantity is a hypothetically measurable amount of something. For example, the term meter, defined in the `Standard-Units` ontology, is an instance of the class Unit-Of-Measure defined in the `Physical-Quantities` ontology.

We came to revise the `Standard-Units` ontology because it was included in `Chemical-Elements`. We needed to check that the units of measurement of certain attributes in `Chemical-Elements` befitted the knowledge and usual practice of experts. One example of the type of check that the experts carried out was that an attribute (Semidisintegration-Period) of a concept (Elements) was filled in with a particular value type which was associated with a unit of measurement (Year). After the experts had drawn up the inspection document setting out the properties to be checked, each query was transformed into the vocabulary of the ontology. For example, check that the Semidisintegration-Period of the concept Elements of the ontology Chemical-Elements is filled in with a value type Time-Quantity and its unit of measurement is Year. This is illustrated in Figure 4.

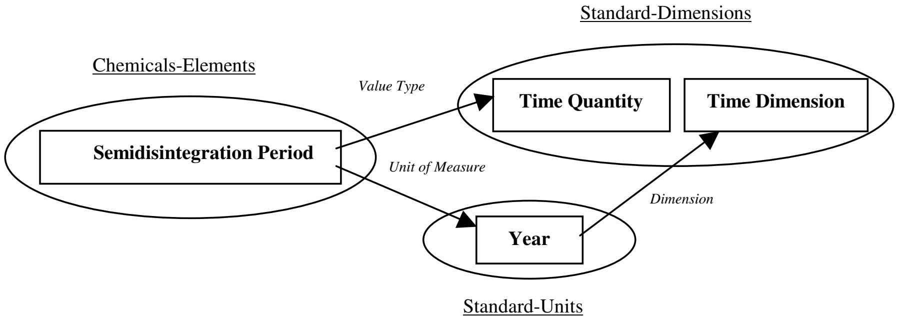

Figure 4. Relation between the Standard-Units and Standard-Dimensions ontologies.

When reviewing all the units of measure present in Chemical-Elements, we checked that they all appeared in Standard-Units. Any that were missing were added. Basically, there were two manners of reviewing the Standard-Units ontology: (1) Review the Ontolingua code of the ontology at the symbolic level, which means that the ontology has to be analyzed using Ontology Server facilities. This option was rejected as domain experts do not understand formal ontologies codified in ontology languages [1]. So, they could neither validate nor formalize knowledge without an ontologist's help; and (2) Review the ontology at the knowledge level using the work method described in section 4. This is the approach taken in this paper. The following describes how the work method was applied to the Standard-Units ontology.

5.2 Reverse engineering

The Standard-Units ontology was analyzed on the basis of its Ontolingua implementation. Figure 5 shows a preliminary conceptual model that possibly originated such implementation. It is important to note that this ontology contains neither relations, functions nor axioms. The hierarchy illustrates that there are two classes: Unit-Of-Measure and System-of-Units, both defined in the Physical-Quantities ontology. In this manner, a series of units of measure which are instances of the class Unit-Of-Measure are defined in Standard-Units, as well as a class, Si-Unit, which groups all the SI units. Additionally, Si-Unit is defined as an instance of the class System-of-Units, as there could be other systems grouping another series of units, which are also, instances of units of Unit-Of-Measure.

In the Standard-Units ontology, all the units have a property that indicates the dimension of the aforesaid unit. These dimensions are defined in the Standard-Dimensions ontology, which has two hierarchies. The hierarchy representing the definition of dimensions is shown in figure 6. In this ontology, there is a class, called Physical-Dimension, which is also defined in the `Physical-Quantities` ontology, of which all the dimensions defined are instances.

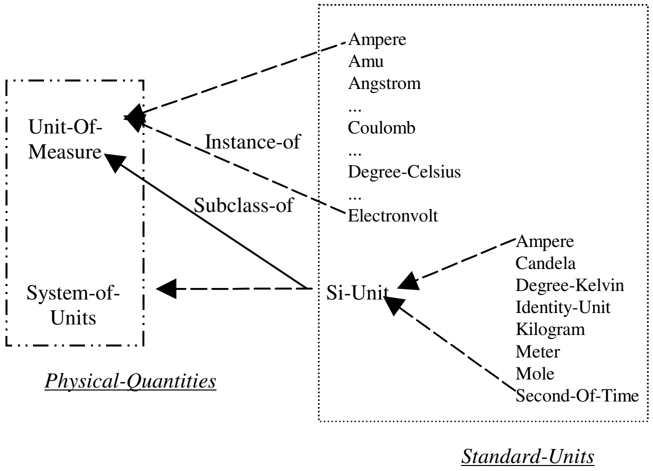

Figure 5. Preliminary hierarchy of the Standard-Units ontology.

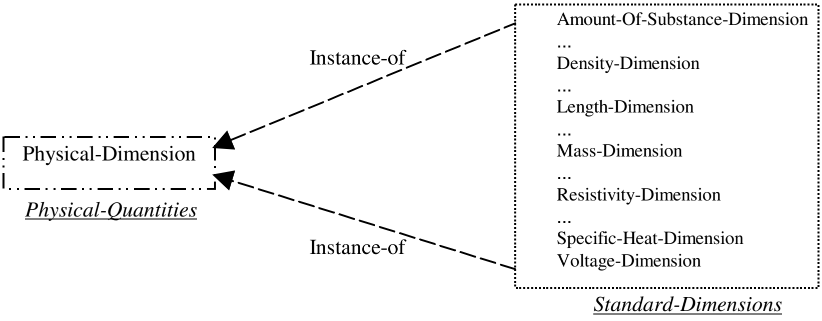

Figure 6. One of the hierarchies of the `Standard-Dimensions` *ontology.*

5.3 Restructure to create a new conceptual model

Here we summarize some design criteria and a set of principles that have proved useful in the development of ontologies. Gruber [17] identified five design criteria: *Clarity and Objectivity*, which means that the ontology should provide the meaning of defined terms by providing objective definitions and also natural language documentation; *Completeness*, which means that a definition expressed by a necessary and sufficient condition is preferred over a partial definition (defined only by a necessary or sufficient condition); *Coherence*, to permit inferences that are consistent with the definitions; *Maximize monotonic extendibility,* which means that new general or specialized terms should be included in the ontology in a such way as does not require the revision of existing definitions; and *Minimal ontological* *commitments*[6], which means making as few claims as possible about the world being modeled, giving the parties committed to the ontology freedom to specialize and instantiate the ontology as required. When building taxonomies, the Ontological Distinction Principle [7] proposes that classes in an ontology should be disjoint. The criterion used to isolate the core of properties considered to be invariant for an instance of a class is called the Identity Criterion.

This section presents the process used to restructure `Standard-Units` ontology. This ontology was restructured bearing in mind its future use by the `Chemical-Elements`, `Monatomic-Ions` and `Environmental-Pollutants` ontologies. It also provides a set of guidelines, which can be used, for building ontologies.

5.3.1 Analysis

Taking into account figures 5 and 6, and the `Standard-Units` and `Standard-Dimensions` Ontolingua code, the most prominent problems and faults found are:

1. There is no taxonomic organization identifying the general concepts that divide into other more specific concepts all the way down to instances. By contrast, there is a single class to which all the instances are subordinated. This is not really correct. First, the instances cannot be classified by similar characteristics. Second, part of the inference power allowing some concepts to inherit properties from more general concepts in a properly diversified hierarchy is lost. It would be more beneficial to build branched taxonomies using an identity criterion to take advantage of the above-mentioned benefits.

2. Definitions that should be made in the same manner, as they refer to similar concepts, are made differently in the implemented Ontolingua code. For example, the SI base unit of measure called Ampere was defined as follows:

```
(Define-Frame Ampere
          : Own-Slots
          (( Documentation “Si electrical current unit.”)
           (Instance-Of Unit-Of-Measure)
           (Quantity.Dimension Electrical-Current-Dimension))
          : Axioms
          ((= (Quantity.Dimension Ampere) Electrical-Current-Dimension)))
```

However, the following instance definition was used to define Meter, which is another SI base unit:

```
(Define-Instance Meter (Unit-Of-Measure)
          “SI length unit. No conversion is given because this is a standard.”
          : Axiom-Def
          (And (= (Quantity.Dimension Meter) Length-Dimension)
               (Si-Unit Meter)))
```

[6] “Ontological commitments are an agreement to use the shared vocabulary in a coherent and consistent manner. They guarantee consistency, but not completeness of an ontology” [18].

It would be advantageous to use the same pattern to make sibling definitions, thus improving ontology understanding and making it easier to include new definitions. This would improve the clarity of the ontology and its monotonic extendibility.

3. The choice of names for the different instances does not comply with a fixed standard. For example, the different multiples and divisors of Ampere are called: Milli-Amp, Nano-Ampere and Pico-Ampere. To ease ontology understanding and improve its clarity, the same naming conventions should be used to name related terms. Therefore, the above-mentioned names should be standardized and denoted as follows: Milli-Ampere, Nano-Ampere and Pico-Ampere.

4. The multiples of the base units do not appear to have been chosen systematically. For instance, Kilo-ohm and Milli-meter are omitted. Incompleteness is a fundamental problem in ontologies [14]. In fact, we cannot prove either the completeness of an ontology or the completeness of its definitions (an omission can always be found), but we can prove both the incompleteness of a definition or the incompleteness of an ontology, if at least one definition is missing with respect to the established framework of reference. When restructuring the `Standard-Units` ontology, our framework was the set of units of interest for the `Chemical-Elements`, `Monatomic-Ions` and `Environmental-Pollutants` ontologies. As Kilo-ohm and Milli-meter will not be used in these ontologies, we can say that the `Standard-Units` ontology is complete in this framework of reference.

5. The ontology includes factors of conversion between different units of the same dimension. However, this conversion is not always made from one particular unit to the unit that is considered as the base unit in the SI. For example, taking the base unit of time Second, each definition of its multiples (minutes, hours, etc.) and its submultiples (millisecond, microsecond, etc.) should contain the appropriate factor of conversion to seconds. However, definitions appear in the `Standard-Units` ontology with factors of conversion as follows:

```
(Define-Frame Day
: Own-Slots
( (Documentation “one day, i.e. 24 hours”)
  (Instance-Of Unit-Of-Measure)
  (Quantity.Dimension Time-Dimension) )
: Axioms
( (= Day (* 24 Hours)) (= Year (* 365 Day) ) ) )
```

In this definition, the factors of conversion of the unit Day are established in relation to non-base units (hours and years), but not to the base unit (seconds). The following factor of conversion should be added to the formal definition:

```
( (= Day (* 86400 Second-Of-Time)))
```

The conversion should always be made to the base unit to improve the clarity of the ontology. Other commonly used factors of conversion between units can also be added, but the conversion to the base unit should never be missing.

6. Some definitions have quite a poor informal language description, which provides the user with no information. This is the case of the natural language definition of Meter, which states: "SI length unit. No conversion is given because this is a standard." An extreme example is Kilometer, for which no informal definition is given at all. A natural language definition should be included whenever possible to give a better understanding of the more formal definition made later. In the example, " A Meter is 1650763.73 wave lengths in vacuo of the unperturbed transition $2p_{10}$ - $5d_5$ in ^{86}Kr."

7. The vocabulary of the `Standard-Dimensions` ontology has not been used in a standardized manner either. Thus, for example, the dimension Megapascal is defined as a Pressure-Dimension:

```
(Quantity.Dimension Megapascal Pressure-Dimension)
```

whereas the dimension Pascal is said to be:

```
(= (Quantity.Dimension Pascal)
(* Force-Dimension (Expt Length-Dimension -2)))
```

As the Pressure-Dimension definition exists in the `Standard-Dimensions` ontology, which is used by Standard-Units, it would be more rational and clearer to define all the units of pressure using this dimension, instead of using its equivalent in units of length and force. Therefore, the dimension Pascal should be defined as follows:

```
(Quantity.Dimension Pascal  Pressure-Dimension)
```

8. In the `Standard-Units` ontology, the number Pi (π) is defined as an instance of the real numbers because a factor of conversion between angles and radians appears in the definition of Angular-Degree.

(Define-Instance **Angular-Degree** (Unit-Of-Measure)
"Angular measurement unit."
:= (* Radian (/ The-Number-Pi 180))
:Axiom-Def (= (Quantity.Dimension Angular-Degree) Identity-Dimension[7]))

As this is an ontology of units of measure, definitions that have nothing to do with the above units must not be included. This problem could be solved in two ways: one possible solution would be to delete the definition of the number π in the factor of conversion and enter the real number 3.1415926535897936. However, a better and modular solution is to include the real number π in the KIF-Numbers ontology, which could be included in the `Standard-Units` ontology and thus this definition could be used.

When an ontology is restructured, a series of criteria must be established beforehand to assess why the new ontology outputted is of higher quality than its predecessor. The following criteria were established when `Standard-Units` was

[7] Identity-Dimension is the identity element for * operator on physical-dimensions. This means that the product of identity-dimension and any other dimension is the other dimension.

restructured: (1) establish the framework of reference against which to prove the completeness of the ontology; (2) model the knowledge of the domain using the ontological distinction principle; (3) build taxonomies that allow property inheritance to be applied; (4) define terms uniformly, using the same patterns to define similar terms, which improves the clarity of the ontology, its understanding by future users and its monotonic extendibility; (5) the documentation accompanying each definition must be clear, useful and give a better understanding of the formal definition of the term; and (6) increase the information contained in the original ontology. If the original ontology was found not to contain enough domain knowledge, new classes, instances, relations, functions and axioms should be added to the new implementation.

5.3.2 Synthesis

As mentioned above, the `Standard-Units` ontology was analyzed because it is used in `Chemical-Elements`, which is used in the `Monatomic-Ions`, which is included in the `Environmental-Pollutants` ontology. After analyzing the `Standard-Units` Ontolingua code and obtaining a possible underlying conceptual model of the ontology and after considering the problems explained above, we modified the conceptual model of the `Standard-Units` ontology as follows:

Standardize naming conventions. We gave standard names to the new classes and instances. The names of the classes in the `Standard-Units` ontology were chosen taking into account the type of units represented and the names of the dimensions found in the `Standard-Dimensions` ontology.

Specialization of concepts. The goal was to identify general concepts that are specialized into more specific and disjoint concepts down to domain instances. The identity criterion used for specialization was to group units according to the base unit. Therefore, we can say that the restructured ontology complies with the Ontological Distinction Principle. For example, all the units for measuring length are grouped within the same class. The name of this class is Length-Unit, and its instances are: Meter (SI base unit), Angstrom, Centimeter, Foot, Inch, Kilometer and Mile. In this case, the new conceptual model includes one class for each type of SI base. We have created 19 new classes, and all these classes are disjoint. We also have maximized the monotonic extendibility of the `Standard-Units` ontology because the inclusion of classes and instances does not require the revision of existing definitions.

Branched taxonomies. Whenever possible, the hierarchy must be sufficiently branched by similar characteristics to increase the power provided by inheritance mechanisms between classes and instances. Figure 7 illustrates the new hierarchy, which should be interpreted as: all the concepts of the first branch are subclasses of the Unit-Of-Measure class, and the terms represented in boxes are instances of the concept to which they are linked by an arrow.

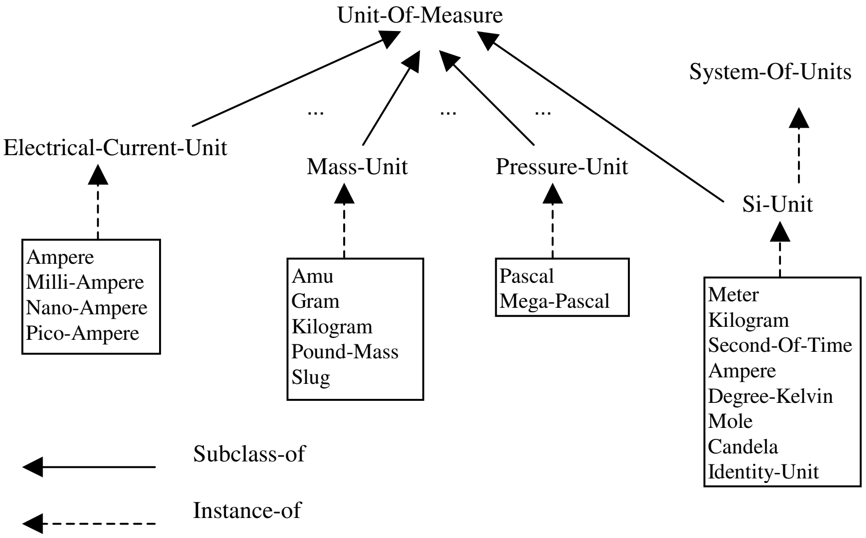

Figure 7. Taxonomy for the Modified Conceptual Model.

Inclusion of new properties and changes to existing properties. In this ontology, the property Abbreviation was added to each unit defined for the purpose of extending the use of this international standard for this attribute, ruling out widespread, though not absolutely correct, uses. In this manner, all the people who use this ontology will be accustomed to using the same standard abbreviation.

Minimize the semantic distance between sibling concepts [2]. Similar concepts are usually grouped and represented as subclasses of one class and should be defined using the same set of primitives, whereas concepts which are less similar are presented further apart in the hierarchy. All the terms in the restructured ontology have been defined using the same pattern in order to give a clearer understanding of the ontology. In this case, the factors of conversion have been expressed from any unit to its SI base unit. For any units that are not part of the SI base, the unit most commonly used by the international scientific community was chosen as the base unit used for the purpose of conversion. It is important to note that all the factors of conversion between units of the same type could be included, if considered useful, as this would not increase ontology complexity.

5.3.3 Configuration management: `Standard-Units`

In Software Engineering, configuration management has three objectives [20]: (1) establish and maintain the integrity of the products generated during a software development project and throughout the entire product life cycle; (2) evaluate and control the changes made to products, that is, control the evolution of the software system; and (3) ease the understanding of product evolution. Therefore, configuration management applied to the ontological engineering field can be considered as ameansof assuring thequalityof theontologies andcan,therefore,be included to supplement validation and verification activities [15].

For the purpose of assuring information about the evolution of the `Standard-Units` ontology, a rigorous change control has been performed throughout the restructuring phase. The goal is to have all the changes documented, detailing the changes made, their causes and effects. It is important to perform proficient change control of both definitions and taxonomies. In this manner, any ontologist who needs to use part of or the entire ontology can easily understand its evolution. Even if an ontology has not been fully developed, provided it is well documented, it could be finished off by another developer using the existing documentation. The configuration management documents can rule out incorrect decision making, if they state the courses of action to be taken at any time, and justify the choice of one rather than another. Change control also helps end users to determine which version of the ontology they require for their system or for the new ontology they are to develop.

Change control starts with a petition for change, followed by the classification and registration, approval or initial rejection and evaluation of the change petition, submission of the change report to the Change Control Committee, performance of the change and certification that the change was made correctly. It ends when the result is reported to the person who proposed the change. Figure 8 presents an example of a control report for a change made to the `Standard-Units` ontology.

***Description of the Change*:** Modify the hierarchy of the *Standard-Units* ontology shown in figure 2, as it does not include intermediate classes that represent each type of SI base unit. In this model, all the instances of the ontology depend on one class.

***Need for Change*:** It is not technically correct to have a class from which all the instances of the ontology hang. This structure prevents concepts being classed by similar characteristics, and some of the inference power allowing concepts to inherit properties from other more general concepts in a properly diversified hierarchy is lost.

***Effects of the Change*:** The hierarchy has been satisfactorily branched, as shown in figure 7. In this case, one class has been created for each type of SI base unit. This change affects all the instances of the ontology, as the Unit-Of-Measure class has to be replaced in the formal definition of the instances by the new class representing the SI base unit to which they belong.

***Alternatives*:** None.

Date of change: 27/03/98.

Change made: Changes are shown in figure 7.

Figure 8. Change control report.

5.4 Forward engineering: implementation of the new ontology

The new conceptual model of the `Standard-Units` ontology was reimplemented in Ontolingua using the Ontology Server editor. The new ontology has also been evaluated. In fact, (1) The ontology is syntactically correct, as it successfully passes the Ontology Server Analyze tests; (2) the ontology is complete for its use in `Chemical-Elements`, `Monatomic-Ions` and `Environmental-Pollutants` ontologies. The experts checked that it is possible to specify the units of measure of the properties identified in these ontologies. They also verified with ontologists that the checks identified in the inspection document have been made; (3) the ontology is internally consistent and the knowledge formalized has been checked against the above-mentioned sources of knowledge; and (4) the ontology is concise, as there is no redundant knowledge.

6 Review of Chemical-Elements

The need to use the Chemical-Elements ontology in the Monatomic-Ions and Environmental-Pollutants ontologies led us to review this ontology, as we had done for Standard-Units. The result of the review process showed that the different versions of the ontology needed to be merged to output a new unified and corrected ontology which could be extended before being included in the Monatomic-Ions ontology. The Chemical-Elements review process was divided into three clearly separate types of activities: technical evaluation, merging and configuration management.

Technical evaluation. The knowledge present in the conceptual model of the ontology was technically evaluated [14] (verified and validated) with chemical and environmental experts for the purpose of ascertaining whether the knowledge represented was correct and complete and detecting any missing knowledge. As with Standard-Units, we decided to review the conceptual model of the Chemical-Elements ontology at the knowledge level using a series of intermediate representations proposed by METHONTOLOGY. For this purpose, the conceptual model of the ontology was given to the chemical and environmental experts, along with an explanation of the meaning of the intermediate representations. The experts and ontologists verified and validated the model in 6 hours and reached the following conclusions: (1) add properties that are useful from the viewpoint of both the chemical element in its pure state and the environment; (2) retain any properties that, although they are not useful for the monatomic ions ontology, can be used to represent elements in their pure state; (3) adapt the names chosen; (4) check the values of the class and instance attributes for correctness using the sources of information recommended by the chemical and the environmental experts; and (5) validate (experts) that the definitions represented formally correspond with the knowledge that they were supposed to represent contained in books, handbooks, etc.

Merging. Development of the Chemical-Elements ontology commenced in June 1995, and a first stable version was produced in December 1996 [10]. Since then, different versions of this ontology have been created and used: (1) to extend the intermediate representations used at the conceptualization phase of METHONTOLOGY; (2) to test the usefulness and validity of the new intermediate representations proposed; (3) by the Ontogeneration system [1], which allows Spanish users to consult and access the knowledge contained in the Chemical-Elements ontology in their own language. A unified conceptual model was built merging all the releases of this ontology, and includes all the improvements.

Configuration Management was carried out according to the guidelines described in section 5.3.3 to make this new version of Chemical-Elements easier to understand for users. As a result, a Chemical-Elements configuration management document was outputted that includes a series of change control reports related to the terms modified in this ontology.

Conclusions

Although the concept of reengineering is well established in Software Engineering, the field of reengineering is totally new in the Ontological Engineering field. Therefore, the main contributions of this paper are to start up research into a process that allows any ontology to be reengineered, and configuration management and change control to be carried out on the ontology as a result of this reengineering activity. The main contributions can be summarized as a preliminary method was proposed for Ontological Reengineering, which includes three activities: Reverse Engineering, Restructuring and Forward Engineering. The *reverse engineering* activity produces a preliminary conceptual model of the ontology from its code. The *restructuring* activities involve: (1) performing a technical evaluation of the initial conceptual model with the expert; (2) reorganizing and extending the initial conceptual model to output a new conceptual model according to a series of criteria (standardize naming conventions, specialize concepts, branch taxonomies, minimize the semantic distance between sibling concepts, etc.) established beforehand. The restructuring process is carried out bearing in mind the use of the restructured ontology by the ontology/application that reuses it; (3) keep records of the changes performed; and (4) build a new, more correct conceptual model, accepted by the experts. The *forward engineering* activity produces a document containing the implementation of the new conceptual model, including the suggested changes. The reegineering process includes the evaluation of both the original and the resulting ontology, and performing configuration management to keep records of ontology evolution, the changes made, their causes and effects.

Future work will include primarily: (1) addressing in more depth the theoretical foundations of ontology reengineering, (2) extending the work method proposed after reengineering more complex ontologies that include relations, functions and axioms, apart from taxonomies of concepts and instances, and (3) developing flexible tools to automate the reengineering process.

Acknowledgements

We would like to express our thanks to the following persons: Almudena Galán and Rosario García, for their knowledge of chemistry and the environment; Mariano Fernández, for providing `Chemicals` and all the documentation required to understand its construction and evolution; Sofía Pinto, for her help in `Chemical-Elements` configuration management; and Juan Manuel García-Pinar and Mariano Fernández again, for converting the `Chemicals` ontologies from the earlier ODE tool format to the current version. We also thank the anonymous reviewers for their comments.

REFERENCES:

[1] Aguado, G.; Bañón, A.; Bateman, J.; Bernardos, S.; Fernández, M.; Gómez-Pérez, A.; Nieto, E.; Olalla, A.; Plaza, R.; Sánchez, A. *"ONTOGENERATION: Reusing domain and linguistic ontologies for Spanish text generation."* Workshop on Applications of Ontologies and Problem Solving Methods. ECAI-98. Brighton (UK). 1998. **The 13th European Conference on Artificial Intelligence.**

[2] Arpírez, J.; Gómez-Pérez. A.; Lozano, A.; Pinto, H.S. *"(ONTO)^{2}Agent: An Ontology-Based WWW broker to select ontologies."* ECAI-98. Brighton (UK). 1998. Workshop on Applications of Ontologies and Problem Solving Methods. **The 13th European Conference on Artificial Intelligence.**

[3] Barley, M.; Clark, P.; Williamson, K.; Woods, S. *"The neutral Representation Project."* Boing Research and Technology. **Ontological Engineering. AAAI-97** Spring Symposium Series. March 97. Stanford University. California. 1997.

[4] Blázquez, M.; Fernández, M.; García-Pinar, J.M.; Gómez-Pérez, A. *"Building Ontologies at the Knowledge Level using the Ontology Design Environment."* **KAW98**. Banff, Canada. 1998.

[5] Blum, B. *"Software Engineering. A holistic view."* Oxford University Press. 1992.

[6] Böhm, C.; Jacopini, G. *"Flow diagrams, Turing machinesd, and languages with only two formation rules."* Communications of the ACM, May 1996. PP:366-371.

[7] Borgo, S.; Guarino,N.; Masolo, C. *"Stratified Ontologies: The case of physical objects"*. **Workshop on Ontological Engineering. ECAI96**. Budapest. PP: 5-15

[8] Chikofsky, E.J.; Cross II, J.H. *"Reverse Engineering and design recovery: A taxonomy."* Software Magazine. January 1990. PP:13-17.

[9] Farquhar, A.; Fikes, R.; Pratt, W.; Rice, J. *"A collaborative ontology construction for information integration."* Technical Report KSL-95-63. Knowledge Systems Laboratory. Stanford University. 1995.

[10] Fernández, M. *"CHEMICALS: Ontología de elementos químicos."* Proyecto Fin de Carrera. Facultad de Informática de Madrid. UPM. December 1996.

[11] Fernández, M.; Gómez-Pérez, A.; Juristo, N. *"METHONTOLOGY: From Ontological Art Towards Ontological Engineering"*. **Ontological Engineering. AAAI-97**. Spring Symposium Series. Stanford 1997. PP:33-40.

[12] Fernández, M.; Gómez-Pérez, A.; Pazos, A.; Pazos, J. *"Building a Chemical Ontology using METHONTOLOGY and the Ontology Design Environment."* IEEE Intelligent Systems. Special Issue on Uses of Ontologies. January/February 1999.

[13] Genesereth, M.R.; Fikes, R. *"Knowledge Interchange Format. Version 3.0 Reference Manual"* Tech Report Logic-92-1. Computer Science, Stanford University (CA)1992.

[14] Gómez-Pérez, A. *"A framework to verify knowledge sharing technology"*. Expert Systems with Application. Vol.11, N.4. 1996 . PP:519-529).

[15] Gómez-Pérez, A. *"Knowledge Sharing and Reuse"*. The Handbook of Applied Expert Systems. Edited by Liebowitz. CRC. 1998.

[16] Gómez-Pérez, A.; Juristo, N.; Pazos, J. *"Evaluation and Assessment of the Knowledge Sharing Technology."* Towards Very Large Knowledge Bases. N.J.I. Mars, Ed. IOS Press, 1995.

[17] Gruber, T. *"Towards Principles for the Design of Ontologies"*. Ksl-93-04. Knowledge Systems Laboratory. Stanford University. 1993.

[18] Gruber, T.;Olsen, G. *"An ontology for Engineering Mathematics"*. **Fourth International Conference on Principles of Knowledge Representation and Reasoning.** Doyle, Torasso y Sandewall (eds.) Morgan Kaufmann. 1994.

[19] Julling, R.; Srinivas, Y.V.; Blaine, L.; Gilham, L.M.; McDonald, J.; Waldinger, R. "Specware language manual." Tech Report, Kestrel Institute. 1995.

[20] Pressman, R.S. *"Ingeniería del Software. Un enfoque práctico."* Mac-Graw Hill. 1993.

[21] Yourdon, E. *"RE-3. Re-engineering, restructuring and reverse engineering."* American Programmer Magazine, Vol.2, N°4, April 1989. PP:3-10.