Qood grid:
A metaontology-based framework for ontology evaluation and selection

Aldo Gangemi   Carola Catenacci   Massimiliano Ciaramita   Jos Lehmann
Laboratory for Applied Ontology, ISTC-CNR, Roma (Italy)
{aldo.gangemi, carola.catenacci, m.ciaramita, jos.lehmann}@istc.cnr.it

ABSTRACT

The paper exemplifies the application of two models for ontology description, evaluation and selection. O^2 is a meta-ontology (an ontology that has ontologies and ontology elements in its domain), based on semiotics, which describes four dimensions of ontologies as communication objects: structural, formal, conceptual, and pragmatical. Starting from a large set, measures have been singled out in order to evaluate ontologies. *oQual* is an ontology of ontology selection which provides a formal tool to devise the best set of criteria for an ontology-based project, and to match ontologies to those criteria. A set of criteria is called a *qood*. In the paper we briefly describe the models and provide a small example of applying a qood according to *QoodGrid*, a methodology for ontology evaluation for one's purposes (the basis for selection).

1. INTRODUCTION

A key problem in the field of ontology development and reuse is the current lack of a comprehensive and global approach to ontology evaluation. Although a number of different methodologies have been proposed since the very first days of research in this area [18], a single, unifying framework for tackling the problem at both theory and application level is still wanted. On the other hand, there is a widespread awareness that well-understood notions of evaluation and selection[1], as well as clear principles for the factual application of such notions, are a crucial must for ensuring the success of semantic technologies in industrial and commercial sectors. The more so in the Semantic Web context, where ontologies promise to be essential components of technologies that are able to cope with high interconnection, constant change and incompleteness.

In previous work [5,6], we have presented a formal model for ontology evaluation which is meant to provide exactly such a unifying framework. Based on an analysis of the most relevant methods proposed in the literature, the model consists, in the first palce, of a metaontology – called O^2 – which provides a semiotic foundation to the elements and features that are targeted by evaluation[2]. This metaontology is complemented with an ontology of ontology evaluation and selection – *oQual* – which allows to pick up ontology elements by means of O^2, provides quality-parameters and, when feasible, their ordering functions[3]. These background ontologies are briefly presented in Section 2 of the present paper.

Based on O^2, it is possible to identify three main types of measures for ontology evaluation: *structural measures*, that are typical of ontologies represented as graphs; *functional measures*, that are related to the intended use of an ontology and of its components, i.e. their function; *usability-profiling measures*, that depend on the level of annotation of the considered ontology. For each of these measure types, we have provided definitions, as well as examples of preferential orders [5]. Some definitions and examples from the internal taxonomies of the three types are presented in Section 3.

Evaluation is important, but within a given project, we may want also to *select* an ontology according to the criteria that are relevant for that project. In practice, we may want to define quality parameters that range over some of the attributes obtained from structural, functional, or purely usability-related measurement. In Section 4, we introduce the *qood grid*, i.e. a framework for relating and applying *quality-oriented ontology descriptions* (qoods), *principles* (elementary qoods), *value spaces*, *parameters*, dependencies and preference functions

---

[1] In previous work [5], we have referred to the principled choice of an ontology over others in given contexts as 'validation'; however, in order to avoid a potential ambiguity with the process of 'certification' (not addressed here), we have decided to use 'selection' in its place (following R.M.Sabou, V.Lopez, E.Motta, and V.Uren, personal communication).

[2] The use of metaontologies is becoming increasingly relevant within the Semantic Web (see e.g. [10]), because of their easy integration and shared construction methods with ontologies proper.

[3] O^2 and *oQual* are formalized both in FOL and OWL, in order to provide a semantic-web-compliant component to create annotated ontology libraries, *e*-pinions, and a reasoning base for enhanced ontology services.

between parameters, and the *trade-offs* needed when composing principles with conflicting parameters. We also provide an analytic case for a trade-off.

In Section 5, we draw some conclusions and sketch a picture of our current and future work.

2. THEORETICAL BACKGROUND: O^2 AND *oQual*

2.1 O^2: a semiotic meta-ontology

We consider an ontology to be a semiotic object, i.e. an object constituted by an *information object* and an *intended conceptualization* established within a *communication setting*. The basic intuition behind this proposal is that information is equivalent to any pattern that is used to represent another pattern, whereas that representation is interpretable by some rational agent as an explanation, an instruction, a command, etc (cf. [14]).

This intuition is formalized by applying an ontology design pattern called *Information↔Description* [3], and it originates a new pattern called O^2 (because it is a "meta-ontology"). O^2, in turn, formalizes the following specification:

a) an ontology is information of a special kind; b) its patterns are graph-like structures; c) they *express* intended conceptualizations, i.e. internal representations (by a rational agent) of *entity types*.

In O^2 (Fig.1) an ontology graph has an intended conceptualization and a formal semantic space admitted by the conceptualization. The graph and the conceptualization are 'kept together' by a rational agent who encodes/interprets the graph, while internally representing its intended conceptualization. An agent can also provide a profile containing metadata that express a "description" of the ontology, e.g. a method to measure the structural or functional properties of an ontology graph, its resulting attributes, its possible quality criteria and values, as well as its lifecycle annotations, such as provenance and informal annotations. A good profile typically enhances or enforces the usability of an ontology.

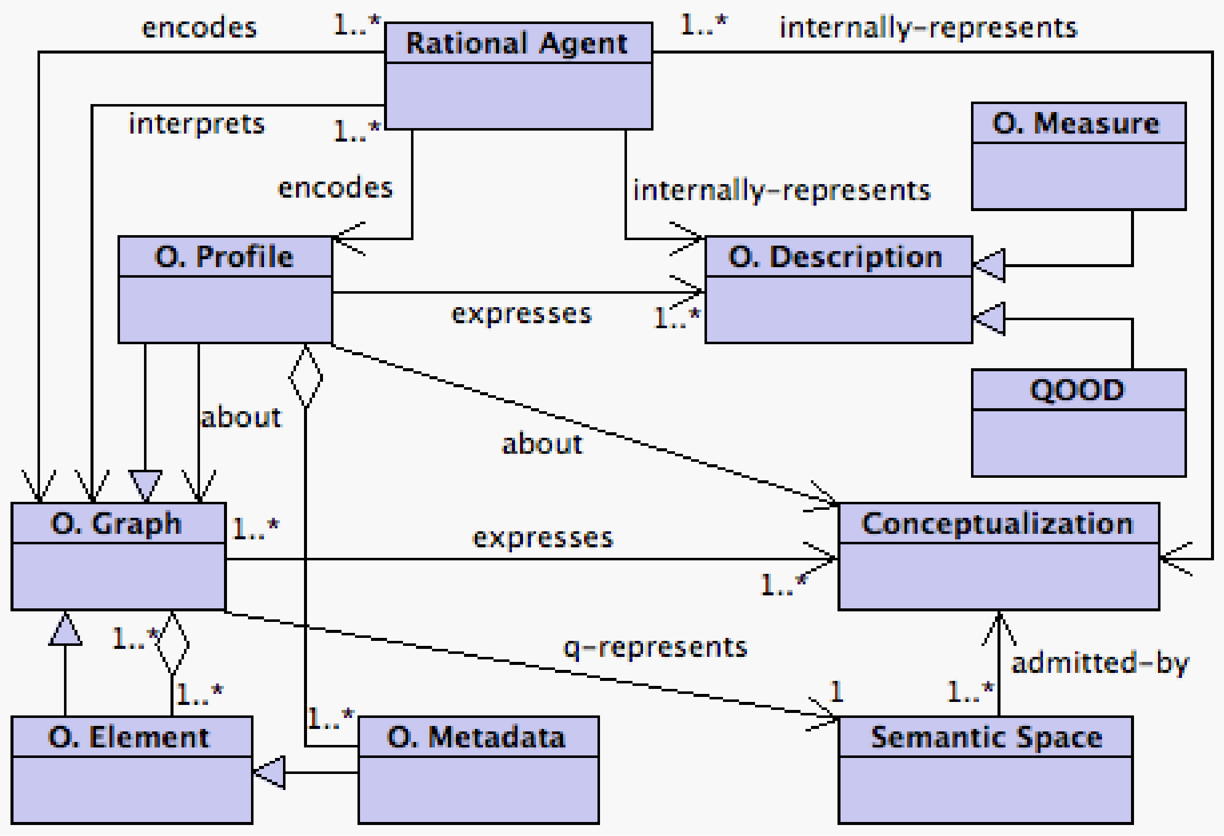

Fig. 1 A UML class diagram depicting the main notions from O^2: ontology graphs, profiles, descriptions, measures, etc.

2.2 *oQual*: an ontology of ontology evaluation and selection

We model ontology evaluation as a *diagnostic task* over ontology elements, processes and attributes. This task is based on ontology descriptions which make explicit those knowledge items that are crucial for evaluating and selecting ontologies. In more detail, the *oQual* ontology involves:

- *Quality-Oriented Ontology Descriptions* (qoods), which are a type of *ontology description* (Fig.1), that provide the *roles* and *tasks* of, respectively, the elements and processes from/on an ontology, and have elementary qoods (called *principles*) as parts. For example, a type of qood is *retrieve*, which formalizes the requirement to be able to answer a certain competency question. In Fig. 2, the *retrieve* type is instantiated as a requirement for the ontology to be able to retrieve the "family history for a condition related to blood cancer", in an ontology project for "blood cancer information service".

- *Value spaces* ("attributes") of ontology elements. For example, the presence of a relation such as: R(*p*,*f*,*c*,*i*), where Patient(*p*), Family(*f*), Condition(*c*), Indicator(*i*).
- *Principles* for assessing the ontology fitness, which are modelled as elementary qoods, and are typically parts of a project-oriented qood. For example, "description of fitness to expertise" is a principle.
- *Parameters* (ranging over the attributes -*value spaces*- of ontologies or ontology elements), defined within a principle. For example, "relation fitness to competency question" is a parameter for the relation R(*p*,*f*,*c*,*i*).
- *Parameter dependencies* occurring across principles because of the interdependencies between the value spaces of the measured ontology elements. For example, the "relation fitness to competency question" parameter is dependent on either "first-order expressiveness" or "presence of a relation reification method" parameters ranging on the logical language of the ontology, because the relation R(*p*,*f*,*c*,*i*) has four arguments and it is not straightforwardly expressible in e.g. OWL(DL).
- *Preferential ordering* functions that compose parameters from different principles. For example, in a "blood cancer information service" project, the "relation fitness to competency question" parameter may be composed with the "computational complexity" parameter.
- *Trade-offs*, which provide a conflict resolution description when combining principles with conflicting parameters. For example, the two abovementioned parameters might be conflicting when the cost of the expressiveness or of the reification method are too high in terms of computational efficiency. A trade-off in this case describes a guideline to simplify the competency question, or a strategy to implement the relation differently.

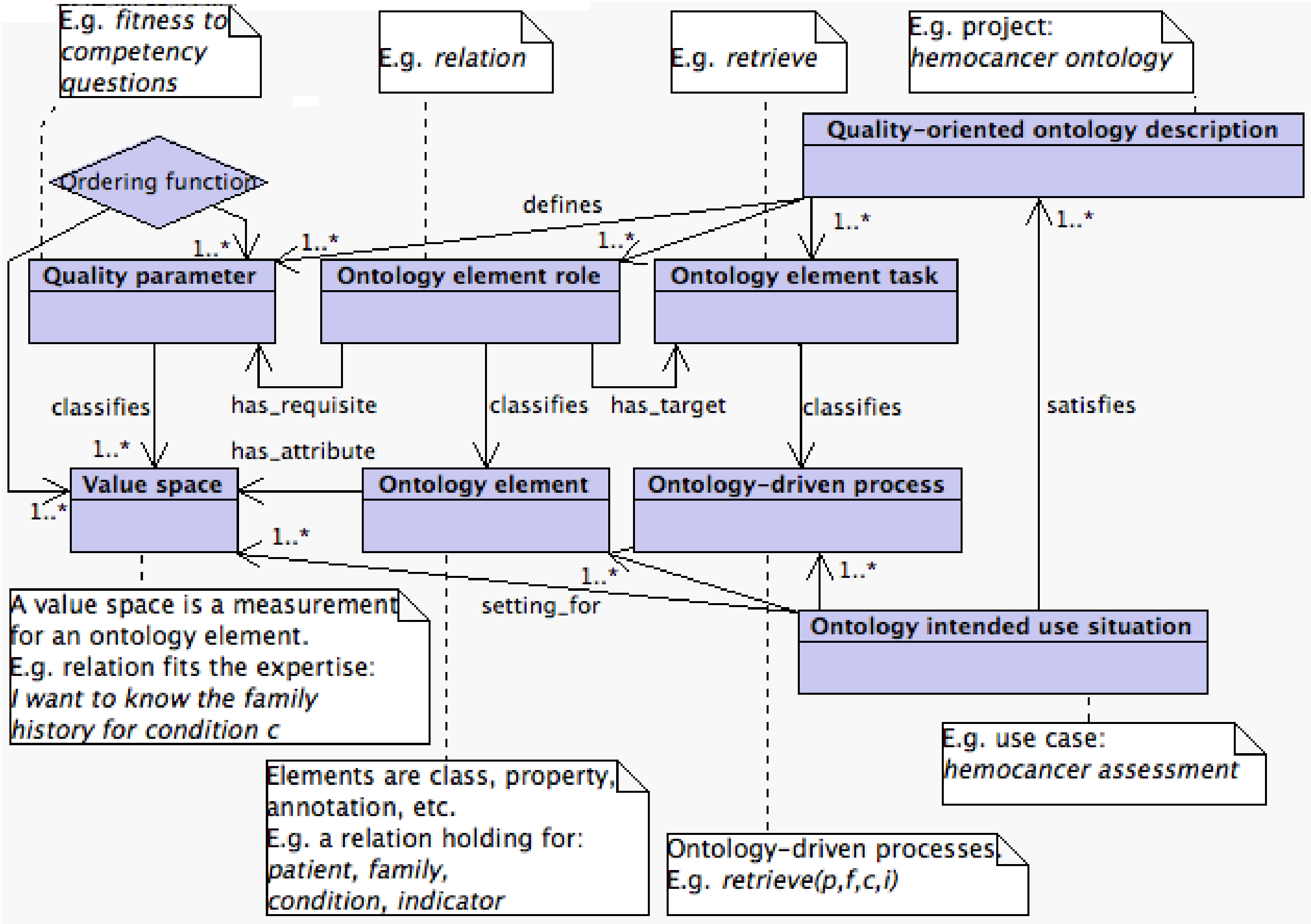

Fig. 2 The *oQual* design pattern applied to a clinical use case. A *qood* based on the *fitness to competency questions* constrains the setting of the intended use for the ontology to be designed in a clinical information project.

The *oQual* formal model, is based on the the *Description*↔*Situation* pattern [4] from the DOLCE ontology library [12], which is integrated with the *Information*↔ *Description* pattern used for O^2.

Ontology descriptions, roles, parameters, and ordering functions are defined on the results of the measurement types that can be performed on an ontology graph, conceptualization, or profile. The results are represented as regions within value spaces. Quality parameters constrain those regions *within* a particular qood. O^2 and *oQual* are currently maintained as OWL models, plugged into the DOLCE ontology library and its design patterns.[4]

---

[4] See http://dolce.semanticweb.org; library can be directly loaded from: http://www.loa-cnr/ontologies/DLP_397.owl.

3. TYPES OF MEASURES

Based on O^2 and *oQual,* and on an analysis of the relevant literature (see e.g. [9,21,11,20,16,15,7,3,13]), it is possible to identify three main types of measures for ontology evaluation: *structural*, *functional*, and *usability-profiling* measures.

The *structural* dimension of ontologies focuses on syntax (e.g. graph structure) and formal semantics. An ontology shows its structural dimensions when represented as a graph. In this form, the *topological*, *logical* and *meta-logical* properties of an ontology may be measured by means of a metric. The existence of these structural dimensions, however, can be seen as independent from the metric that is used. Among the measured topological dimensions are, e.g., *breadth* (related to the cardinality of levels - or "generations" - in a graph), *depth* (related to the cardinality of paths in a graph), *tangleness* (related to multihierarchical nodes of a graph), and *fan-outness* (related to the "dispersion"of graph nodes). Logical adequacy, on the other hand, is measured e.g. through *consistency*, *anonymous classes* and *cycle ratios*. Meta-logical adequacy, finally, is measured through *qualified density* (i.e. presence of meaningful conceptual-relation 'dense' areas, or 'patterns').

The *functional* dimension is coincident with the main purpose of an ontology, i.e. specifying a given conceptualization, or a set of contextual assumptions about an area of interest. An appropriate evaluation strategy should involve finding ways of measuring the extent to which an ontology mirrors a given expertise [17], competency [19], or task. Since these sort of things are in the experience of a given community and are not necessarily represented in their entirety in the available documents, it is probable that no automatised method will ever suffice and that intellectual judgement will always be needed. However, both automatic and semi-automatic techniques can be applied that make such evaluation easier, less subjective, more complete and faster (cf. [3]). The functional measures we have provided in [5] are variants of the *precision, coverage,* and *accuracy* measures introduced by [8], which are in turn based on an analogy with the *precision* and *recall* measures widely used in information retrieval (cf. [1]). They range over *competence adequacy* (e.g. inter-subjective agreement, task adequacy, task specifity, and topic specificity); *NPL adequacy* (e.g. compliance with lexical distinctions), and *functional modularity* (e.g. stratification of foundational/core/domain modules, or granularity).

Finally, *usability-profiling* measures focus on the ontology profile, which typically addresses the communication context of an ontology (i.e. its pragmatics). An ontology profile is a set of ontology annotations: the metadata about an ontology and its elements. Three basic levels of usability profiling have been singled out: *recognition* annotations (e.g. user-satisfaction, provenance and versioning information), *efficiency* annotations (e.g. application-history information), and *interfacing* annotations (e.g. organizational-design information). *Presence, amount*, *completeness*, and *reliability* are the usability measures ranging on annotations.

Notice that the three dimensions of measurement (structural, functional, and usability-profiling) follow a partition into *logical types*: structurally, we look at an ontology as an (information) *object*; functionally, we look at it as a *language* (information object+intended conceptualization), and from the usability viewpoint, we look at its *meta-language* (the profile about the semiotic context of an ontology). Therefore, the dimension types correspond to the constituents of an ontology as a semiotic object.

The complete list of the currently identified measures, with their internal taxonomy and related measurement functions or methods, is presented in [5].

4. THE QOOD-GRID: A FRAMEWORK FOR ONTOLOGY SELECTION

Evaluation is important, but within a given project, we may want also to *select* an ontology according to the criteria and goals that are relevant for that project. In practice, we may want to define quality parameters that range over some of the attributes obtained from structural, functional, or purely usability-profiling measurement. In section 2.2 we have already introduced the distinction between *qoods* (quality-oriented ontology descriptions), *principles* (elementary qoods), *value spaces*, *parameters*, dependencies and preference functions between parameters, and *trade-offs.* In this section, we give a still initial, but more detailed presentation of principles, some of their typical parameters, and an analytic case for a trade-off, showing how the 'qood-grid' actually works.

4.1 Some principles and parameters

Principles are defined here as structured descriptions of the quality of an ontology (*qoods*): they are considered *elementary* qoods because they usually define a limited set of parameters constraining ontology properties in order to support a common *goal*. Principles should also *lack conflicting parameters*.

Here is a list of some qoods emerged in the practice of ontology engineering:

- Cognitive ergonomics
- Transparency (explicitness of organizing principles)
- Computational integrity and efficiency
- Meta-level integrity
- Flexibility (context-boundedness)
- Compliance to expertise
- Compliance to procedures for extension, integration, adaptation, etc.
- Generic accessibility (computational as well as commercial)
- Organizational fitness

The parameters defined by principles can be complex, but at the current state of research, they are usually simple scalars ranging on the measurement value spaces associated with the measures mentioned in the previous Section (see [5] for the extensive list).

Here is a list of parameters defined by the principles introduced above: for an easier understanding, each parameter is presented with the name of measure on which it ranges, preceded by a + or – sign to indicate the scalar region constrained within the value space:

Cognitive ergonomics. Intuition: this principle prospects an ontology that can be easily understood, manipulated, and exploited by final users. Parameters:
-depth
-breadth
-tangledness
+class/property ratio
+annotations (esp. lexical, glosses, topic)
-anonymous classes
+interfacing

+patterns (dense areas)

Transparency. Intuition: this principle prospects an ontology that can be analyzed in detail, with a rich formalization of conceptual choices and motivations. Parameters:
+modularity
+axiom/class ratio
+patterns
+specific differences
+partitioning
+accuracy
+complexity
+anonymous classes
+modularity design

Computational integrity and efficiency. Intuition: this principle prospects an ontology that can be successfully/easily processed by a reasoner (inference engine, classifier, etc.). Parameters:
+logical consistency
+disjointness ratio
-tangledness
-restrictions
-cycles

Meta-level integrity. Intuition: this principle prospects an ontology that respects certain ordering criteria that are assumed as quality indicators. Parameters:
+meta-level consistency
-tangledness

Flexibility. Intuition: this principle prospects an ontology that can be easily adapted to multiple views. Parameters:
+modularity
+partitioning
+context-boundedness

Compliance to expertise. Intuition: this principle prospects an ontology that is compliant to one or more users' knowledge. Parameters:
+precision
+recall
+accuracy

Compliance to procedures for mapping, extension, integration, adaptation. Intuition: this principle prospects an ontology that can be easily understood and manipulated for reuse and adaptation. Parameters:
+accuracy(?)
+recognition annotations (esp. lexical)
+modularity
-tangledness(?)

Organizational fitness. Intuition: this principle prospects an ontology that can be easily deployed within an organization, and that has a good coverage for that context. Parameters:
+recall
+organizational design annotations
+commercial/legal annotations
+user satisfaction
+organizational design annotations

Generic accessibility. Intuition: this principle prospects an ontology that can be easily accessed for effective application. Parameters:
+accuracy (based on task and use cases)
+annotations (esp. policy semantics, application history)
+modularity
-logical complexity

4.2 Preference and trade-offs

Due to partly mutual independence of principles, the need for a preferential ordering of quality parameters required by different principles often arises, e.g. because of a conflict, or because two parameters from different principles are unsustainable with existing tools or resources. OntoMetric [11] is an example of a tool that supports measurement based on a preferential ordering.
A preferential ordering can either define the *prevalence* of a set of parameters from a principle p_1 over another principle p_2, or it can define a composition of the two sets of parameters from p_1 and p_2. A compositions is the result of a *trade-off*. Both prevalence and trade-off descriptions are based on meta-parameters, e.g.: *available resources*, *available expertise*, *business relations*, *tools*, etc.

4.3 Qood-grid: an example in legal ontologies

We explain with a simple example how a trade-off is applied to principle composition.
Transparency and *compliance to expertise* principles usually require *content ontology design patterns* (cf. [4]), involving *hub nodes* (classes with several properties, cf. [13]), then those principles require a *high rate of dense areas* parameter. But dense areas often need the definition of sets of (usually existential) axioms that potentially induce complex (in)direct cycles. Consequently, *high rate of dense areas* depends on a *high complexity* parameter (cf. [2] for the complexity of description logic ports of UML models).
The content design pattern for the *LimitViolation* pattern is an example of such a case (Fig.3).

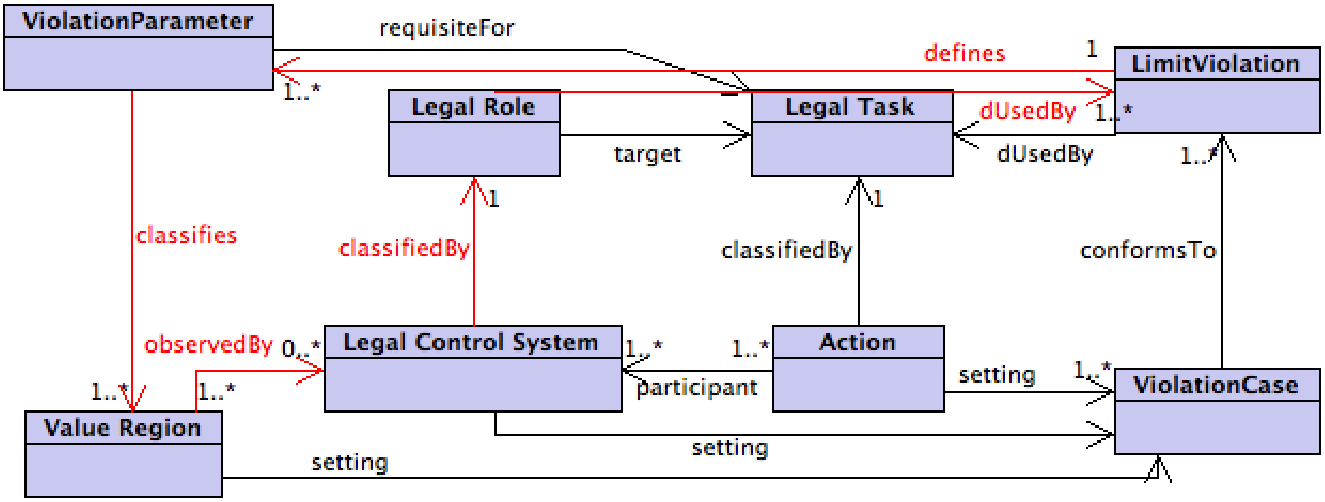

Fig. 3 The *LimitViolation* pattern in UML, showing a potential indirect cycle: a description of limit violation defines violation parameters ranging on some value space (e.g., speed), also assigning (legal) roles and tasks to legally-relevant entities: control systems, vehicles, persons, actions, etc. A violation case conforms to the description if legally-relevant entities and values are classified by parameters, roles, and tasks.

The *LimitViolation* pattern contains the following axioms (restrictions) that constitute a cyclical path, encoded here in OWL abstract syntax (corresponding to the red path in Fig.3):

```
Class(LimitViolation partial restriction(defines
someValuesFrom(ViolationParameter)))
Class(ViolationParameter partial restriction(classifies
someValuesFrom(ValueRegion)))
Class(ValueRegion partial restriction(observedBy
allValuesFrom(LegalControlSystem)))
Class(LegalControlSystem partial restriction(classifiedBy
someValuesFrom(LegalRole)))
Class(LegalRole partial restriction(d-used-by
someValuesFrom(LimitViolation)))
Class(LimitViolation partial restriction(defines
someValuesFrom(ViolationParameter)))
Class(ViolationParameter partial restriction(classifies
someValuesFrom(ValueRegion)))
Class(ValueRegion partial restriction(observedBy
allValuesFrom(LegalControlSystem)))
Class(LegalControlSystem partial restriction(classifiedBy
allValuesFrom(LegalRole)))
Class(LegalRole partial restriction(d-used-by
someValuesFrom(LimitViolation)))
```

If an ontology project using the limit violation axioms is based on a *qood* that aims at both a *transparency* principle, and a *computational efficiency* principle, and we already know (cf. [5]) that computationally efficiency requires a *low rate of cycles parameter*, then we get a conflict of parameters (Fig.4). Therefore, a trade-off may be needed in an ontology project that uses the limit violation axioms. The trade-off can be applied by following two approaches.

The first approach defines a *preference ordering* over the parameters, as mentioned in Section 2.2, which in the example leads either to accept the complexity, or to dismiss the pattern. The pattern is in this case essential to the ontology, then, if the *low rate of cycles* is also required because of e.g. available computational resources, we must resort to the second approach: *relaxation of parameters*.

The possible methods to relax the parameters should act on either the reasoning algorithm, or the axioms. Since the first cannot be changed easily in most ontology projects, the best practice is to modify the model according to some *tuning practices* e.g. involving generalization over restrictions, which in our example can be done on one of the following axioms by substituting the class in the restriction with its superclass:

```
Class(ValueRegion partial restriction(observedBy
allValuesFrom(ControlSystem)))
Class(LegalControlSystem partial restriction(classifiedBy
allValuesFrom(Role)))
```

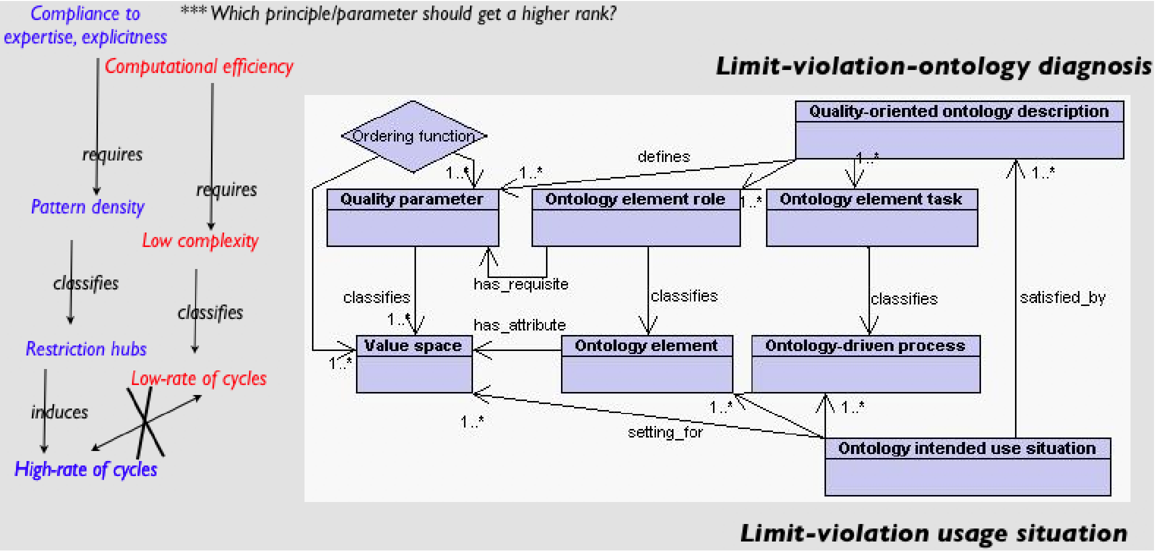

Fig. 4 A *qood* (a diagnosis of an ontology project using the limit violation pattern) that composes two principles requiring conflicting parameters.

5. CONCLUSIONS AND FUTURE DIRECTIONS

O^2 and *oQual* are ontologies (represented in both FOL and OWL, on top of DOLCE-Lite-Plus and its design patterns) that characterise an ontology as a communication object, and allow to make a parametric design of both evaluation and selection (diagnostic) tasks. Ontologies are analyzed in their graph and formal elements, functional requirements, and annotation profile. Therefore our approach results in parametric specifications that address varied measures, ranging from graph properties to logical consistency, precision/recall, intersubjective reliability, etc. These measures, however, do not suggest *prescriptive* selection of an ontology, but rather a process of interactive, distributed selection against well-understood tasks.

Current work is focusing on the empirical assessment of the *oQual* evaluation ontology, the qood-grid and related methods, by measuring existing ontologies, comparing the quality of distinct ontologies that represent the same domain, creating correlations between user-oriented and structural measures, and creating tools to assist ontology evaluation and selection in large industry- and organization-scale projects (until now, the ontology and the method have been tested on fragments of large ontologies, and only thoroughly in the context of the Italian OntoDev project, featuring a mid-size lightweight ontology repository). Collaboration with the Oyster, Onthology, and KnowledgeZone projects[5] are being established in order to harmonize ontology metadata semantics with tools, and to include/extract evaluation annotation in/from current metadata vocabularies.

An extension of our approach to the further process of ontology certification will also be explored.

6. REFERENCES

[1] Baeza-Yates R. and Ribeiro-Neto B., 1999, *Modern Information Retrieval*, Addison Wesley.

[2] Berardi D., Calvanese D., De Giacomo G., 2001, "Reasoning on UML Class Diagrams using Description Logic Based Systems", *Proceedings of the KI'2001 Workshop on Applications of Description Logics*.

[3] Daelemans W. and Reinberger M.L., 2004, "Shallow Text Understanding for Ontology Content Evaluation", *IEEE Intelligent Systems*: 1541-1672.

[4] Gangemi A., 2005, "Ontology Design Patterns for Semantic Web Content", in Motta E. and Gil Y. (eds.), in *Proceedings of the Fourth International Semantic Web Conference*.

[5] Gangemi A., Catenacci C., Ciaramita M., and Lehmann J., 2005, "Ontology evaluation and validation: an integrated formal model for the quality diagnostic task", *Technical Report*, available at http://www.loa-cnr.it/Publications.html.

[6] Gangemi A., Catenacci C., Ciaramita M., and Lehmann J., 2006, "Modelling Ontology Evaluation", to appear in Y. Sure and J. Domingue (eds), *Proceedings of ESWC2006*.

[7] Gómez-Pérez A., 2003, "Ontology Evaluation", in *Handbook on Ontologies*, S. Staab and R. Studer (eds.), Springer-Verlag, pp. 251–274.

[8] Guarino N., 2004, "Towards a Formal Evaluation of Ontology Quality", *IEEE Intelligent Systems*: 1541-1672.

[9] Hartmann J., Spyns P., Giboin A., Maynard D., Cuel R., Suárez-Figueroa M.C., and Sure Y., 2004, "Methods for ontology evaluation", *Knowledge Web Deliverable D1.2.3*, v. 0.1.

---

[5] See http://oyster.ontoware.org; http://www.onthology.org; http://smi-protege.stanford.edu:8080/KnowledgeZone/.

[10] Hartmann J., Palma R., Sure Y., Suárez-Figueroa M.C., and Haase P., 2005, “OMV–Ontology Metadata Vocabulary”, paper presented at the Ontology Patterns for the Semantic Web (OPSW) Workshop at ISWC2005, Galway, Ireland: http://www.research.ibm.com/people/w/welty/OPSW-05/.

[11] Lozano-Tello, A. and Gómez-Pérez A., 2004, “ONTOMETRIC: A method to choose the appropriate ontology”, *J. of Database Management*, 15(2).

[12] Masolo, C., A. Gangemi, N. Guarino, A. Oltramari and L. Schneider, 2004, “WonderWeb Deliverable D18: The WonderWeb Library of Foundational Ontologies”, available at http://www.loa-cnr.it/Publications.html/.

[13] Noy, N., 2004, “Evaluation by Ontology Consumers”, *IEEE Intelligent Systems*: 1541-1672.

[14] Peirce Ch.S., 1931-1958, *Collected Papers*, vols. 1-8, C. Hartshorne, P. Weiss and A.W. Burks (eds), Cambridge, MA: Harvard University Press.

[15] Porzel R. and Malaka R., 2004, “A Task-based Approach for Ontology Evaluation”. *Proeedings. of ECAI04*.

[16] Spyns P., 2005, “EvaLexon: Assessing triples mined from texts”, *Technical Report 09*, STAR Lab, Brussel.

[17] Steels L., 1990, “Components of Expertise”, *AI Magazine*, 11, 2: 30-49.

[18] Sure Y. (ed.), 2004, “Why Evaluate Ontology Technologies? Because It Works!”, *IEEE Intelligent Systems*: 1541-1672.

[19] Uschold U. and Gruninger M., 1996, ”Ontologies: Principles, Methods, and Applications”, *Knowledge Eng. Rev.*, vol. 11, no. 2: 93–155.

[20] Welty C., Guarino N., (2001), “Supporting ontological analysis of taxonomic relationships”, *Data and Knowledge Engineering*, vol. 39, no. 1, pp. 51-74.

[21] Yao H., Orme A.M., and Etzkorn L., 2005, “Cohesion Metrics for Ontology Design and Application”, *Journal of Computer Science*, 1(1): 107-113.