

<http://ontologydesignpatterns.org> [ODP] and Evaluation WikiFlow

Enrico Daga, Valentina Presutti, and Alberto Salvati

Semantic Technology Lab, ISTC-CNR

Abstract. We present **ontologydesignpatterns.org** (ODP), a semantic web portal based on semantic wiki technology, that supports the community around best practices of ontology design. ODP provides services for ontology design pattern evaluation, training on ontology design, and a repository of reusable OWL ontologies. We also present **Evaluation WikiFlow**, a (Semantic) MediaWiki extension that we have developed starting from the ODP requirement for pattern evaluation and certification support.

1 Introduction

Ontologydesignpatterns.org (ODP) is a semantic web portal dedicated to ontology design best practices for the semantic web, with particular focus on ontology design patterns (OPs). ODP is targeted at users who are interested in best practices for ontology design and ontology engineering, and OPs are reusable solutions to recurrent modeling problems. As such, they are a preferential route for designing high-quality ontologies. In [16], OPs are classified into different types: logical, content, presentation, correspondence, etc. Currently, ODP supports the lifecycle of *Content* ontology design patterns (CPs) (solutions to content modeling problems, e.g for time, space, biological sequencing, geographic areas, invoicing, etc.). Being integrated with specific modeling problems, CPs are ideally reusable components for building ontologies that are valid with reference to their task and domain, and due to their simplicity, they can be easily exploited as reference vocabularies for publishing linked data on the web. Currently, ODP is under improvement for supporting the lifecycle of other OP types, and it is also used as a resource for enhancing collaborative ontology design with the NeOn Toolkit¹, e.g. by matching, specializing, and composing CPs in specific ontology projects [17]. In this paper we present two contributions.

- ODP: a *semantic* web portal supporting the life cycle of ontology design patterns, from drafting, to evaluation and certification, and that makes them available for download.
- Evaluation WikiFlow: a (Semantic) Media Wiki extension, named *Evaluation WikiFlow*, that supports the workflow for wiki article evaluation and stores a semantic report of the evaluation history, including the rationales expressed in the evaluation reviews.

¹ <http://www.neontoolkit.org>

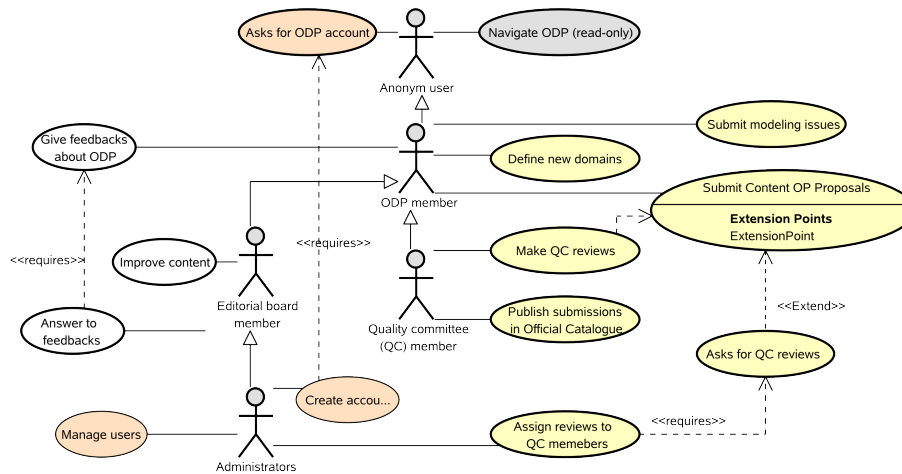


Fig. 1. ODP Use Case Diagram: main functionalities and user roles.

The paper is organized as follows: Section 2 describes ODP organization and main functionalities; Section 3 explains and shows how CPs are presented within ODP; Section 4 depicts Evaluation WikiFlow, and Section 5 reports on some related work. Finally, Section 6 concludes and discusses further developments of this work.

2 ODP Organization

According to the functionality that we have implemented so far, and depicted in Figure 1, ODP features different areas and types of users. Currently, the web portal is organized into the following areas:

Community. This area is meant to receive ODP users' contributions and discussions. It includes a facility for posting modeling issues to the aim of sharing experiences and adopted solutions with the community.

Proposals. This is the area where ODP users can propose OPs². It includes a catalogue of all proposed CPs. There are two alternative procedures for proposing a pattern, both are accessible through the *Propose a Content OP* functionality.

- the user chooses a name for the CP and is guided by a simple form that, once filled and saved, is posted as a new CP proposal;
- the user uploads the owl file of the CP, and the system automatically fills the CP form, that once accepted by the user is posted as a CP proposal. The form can be further edited by the user by the *edit with form* functionality. In order to exploit this facility at its best, the owl file must be annotated in terms of `rdfs:label`,

² Currently, only CP proposals are supported.

`rdfs:comment`, and annotation properties defined by the *CP annotation schema*³. All these annotation properties are mapped to semantic relations into ODP.

The proposed OPs are expected to come from practical and successful experiences of ontology development. All proposed patterns belong to the ODP namespace named `Submissions`.

Reviews. This area collects and publishes the reviews of proposed CPs. Proposed CPs can be reviewed by all registered users i.e. *open reviews*. However, they are eventually reviewed by at least two members of the Quality Committee, formed by ontology experts i.e. *QC reviews*. The aim of reviews is twofold: on one hand, they provide ODP users with explicit rationales behind the evaluation of specific domain solutions. On the other hand, reviews provide the author of a certain CP with guidelines for fixing possible problems in order to get the CP certified.

Open reviews are extremely useful to Quality Committee members while formulating their review, and also for identifying new Quality Committee members from the community.

Catalogue. This is the official CP catalogue. This area collects all CPs that are certified by the ODP Quality Committee. The only difference between certified and proposed CPs is that the formers are guaranteed to be fully described (with regard to ODP specification)⁴, validated by the ODP Quality Committee, and *always* associated with a reusable OWL implementation available for download that has a stable namespace.

Training. This area collects tutorials, publications, and sample modeling exercises.

Feedback. This is the area where ODP users can post feedback entries. Typically, feedback entries identify issues to be addressed in order to improve ODP. The editorial board and the ODP core developers treat such feedback as input/suggestions for raising new development and/or maintenance requirements.

Domain. This page lists all domains that have been defined by ODP users, and allows them to create new ones. Each CP or Modeling Issue is associated with a domain e.g. Organization, Business, Health, Music, etc., by the semantic property `domain`. This semantic property is extremely useful for users who are looking for CPs for a certain domain of knowledge. In fact, they can query ODP about all CPs addressing a certain domain.

Read and write access to ODP areas is handled by a policy based on five different types of users. Types of users are: *AnonymousUser*, *ODPUser*, *EBMember*, *QCMember*, *Administrator*.

- **Anonymous users** (*AnonymousUser*). They have read-only access to the whole portal, and can request an account.

³ <http://ontologydesignpatterns.org/schemas/cpannotationschema.owl>

⁴ See also the cp annotation schema: <http://www.ontologydesignpatterns.org/schemas/cpannotationschema.owl>

- **ODP users** (*ODPUser*). This user type represents everybody who has registered to ODP through the account request page, and hence it is part of the ODP Community. Such users can create and/or edit articles in the Community area, post modeling issues, create and/or edit pages about domains, submit proposals, post feedback entries, make open reviews about proposals, and participate in the discussions (through the discussion page).
- **Editorial board** (*EBMember*). Users in this group answer feedback, propose and contribute to content improvement through a page dedicated to development tasks⁵. The Feedback area is used by editorial board members in order to identify possible development tasks. The development task page is directly used by core developers, and the editorial board members are entitled to manage task priority.
- **Quality committee members** (*QCMember*). These users are entitled to posting *QC reviews* about proposals. Certification of CP that are published in the official catalogue is based on QC reviews, and they are handled in a peer-reviewing like approach. The QC evaluation workflow is managed by the Evaluation WikiFlow extension presented in Section 4.
- **Administrators** (*Administrator*). Users in this group manage account creation, ODP technical issues and upgrades.

3 CP presentation

ODP software is based on Media Wiki⁶, Semantic Media Wiki (SMW)⁷, Semantic Forms (SF)⁸, and other extensions⁹. We have exploited such extensions' features in order to semantically represent as more knowledge as possible about the ontology design patterns. All wiki articles are assigned with a category, and can be related each others through semantic relations. The core of ODP is made of articles that represent ontology design patterns (currently only CPs). According to [16] we have defined a set of semantic properties that comply to the *CP annotation schema*¹⁰. They allow us to specify useful semantic information about CPs, and have guided us as well in defining the visualization template of CP pages.

Consider for example Figure 2. It depicts the ODP page of the CP named *situation*, which is in the catalogue of proposals¹¹. Under the diagrammatic representation of the CP (which is a UML class diagram automatically obtained by Top Braid Composer¹²) on the center-right side of the page there is a box containing CP properties. They include: the person who submitted the proposal; the name of the CP; its intent, consequences, and typically associated competency questions; relations to other CPs, such as its components, its specializing CPs, and CPs that are specialized by it; the URL of

⁵ <http://ontologydesignpatterns.org/index.php/Odp:Development>

⁶ <http://www.mediawiki.org>

⁷ <http://www.semantic-mediawiki.org>

⁸ <http://www.mediawiki.org/Extension:SemanticForm>

⁹ The full list of the extensions can be found at <http://ontologydesignpatterns.org/index.php/Special:Version>

¹⁰ <http://www.ontologydesignpatterns.org/schemas/cpannotationschema.owl>

¹¹ <http://ontologydesignpatterns.org/index.php/Submissions:Situation>

¹² <http://www.topquadrant.com/topbraid/composer/index.html>

the downloadable OWL building block, the source from which it has been extracted, and so on. On the center-left side of the page, all ontology elements defined in the CP are listed and described. Furthermore, each of them has its own wiki article of category `OntologyElement`. On the bottom of the page, there are links to examples of scenarios modeled by using the CP, such examples includes a UML object diagram, a natural language description, and a link to the OWL file that implements it.

Finally, on the very bottom of the page, there are links to possible reviews of the patterns.

Next section describes Evaluation WikiFlow, a SMW extension that is able to support workflows for the evaluation of wiki articles.

4 Evaluation WikiFlow

Evaluation WikiFlow has been released in alpha version as open source software and can be downloaded from the MediaWiki wiki site¹³. The reader can test it on ODP, where it is associated to `ProposedCP` articles.

Evaluation WikiFlow is designed in order to store a semantic representation of the evaluation history of an article. This feature is motivated by the goal of semantically representing the rationales behind an evaluation so as to identify recurrent mistakes and good practices (of ontology design in the case of ODP).

Evaluation WikiFlow is substantiated by a tab, labelled as *evaluation*, which is added on top of the wiki page, as shown in Figure 3. Evaluation WikiFlow's features can be described from two perspectives: configuration and functionality. Configuration features include the following ones.

Selection of article's categories that have to be associated with the evaluation tab.

This feature allows users to activate the evaluation tab and its functionalities (see later in this section) for a set of categories defined by the ontology behind the wiki. E.g. currently ODP activates it for the `ProposedCP` category.

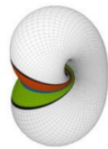
Customization of the semantic form associated to the review page. This feature allows users to define the review schema associated with the correspondent semantic form.

Definition of different semantic review forms associated to different article's categories. This feature allows users to associate different categories (or sets of categories) with semantic forms having different review schemas E.g. in ODP, Logical OPs will be reviewed by using a form different from the one used for reviewing CPs.

Access rights configuration. Evaluation WikiFlow adds five new permissions to the existing ones. They can be configured in order to allow/prevent users to certify articles, and to perform the following actions on reviews: view, ask for, assign, and make. E.g. ODP *QCMember* users have the rights for making a review, while every *ODPUser* can ask for a review.

From the functionality side, Evaluation WikiFlow provides the following features, accessible from the evaluation tab (see Figure 3).

¹³ at <http://www.mediawiki.org/Extensions:EvalWF>



navigation

- Main page
- Modeling Issues
- Proposed Content OPs
- Reviews
- Catalogue
- Feedback
- Domains
- Training

users

- Request an ODP account
- Post a modeling issue
- Propose a Content OP
- Post your Feedback
- Add a Domain
- List of Categories
- List of Properties

quality committee

- Post a Review

content op publishers

- Create a catalogue entry
- Assign review

editorial board

- Development
- Post news

administrator


- Template List
- Form list
- Create a category
- Create a property
- Create a template
- Create a form

search

toolbox

- What links here
- Related changes
- Upload file
- Special pages
- Printable version
- Permanent link
- E-mail user

Submissions:Situation

 If you are a member of quality committee please visit the [evaluation section](#)

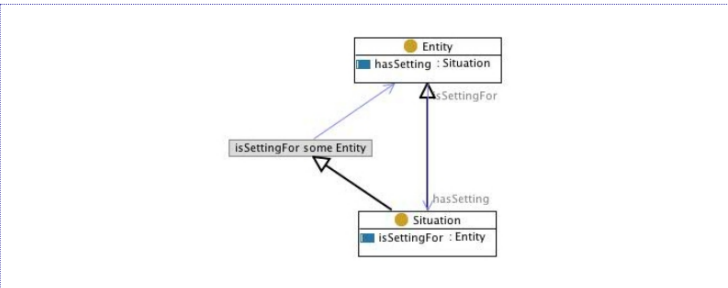
If you are author of this proposal or you want to contribute to this pattern's review, you can:

- ask for a review
- post your open review
- specify if this revision takes in account any of the review(s)
- add a new scenario for Situation

In general, it could be useful to visit the [evaluation section](#) to have informations about the evaluation process of this proposal

Current revision ID: **2262**

Diagram



Elements

The **Situation** Content OP locally defines the following ontology elements:

Entity (owl:Class)

Anything: real, possible, or imaginary, which some modeller wants to talk about for some purpose.

[Entity page](#)

Situation (owl:Class)

A combination of circumstances involving a set of entities. It can be seen as a relational context, relying a relation among the entities involved. In fact, it provides an explicit vocabulary to the *n*-ary relation

[Logical OP](#)

[Situation page](#)

has setting (owl:ObjectProperty)

a relation between entities and situations, e.g. this morning I've prepared my coffee with a new fantastic Arabica (i.e.: (an amount of) a new fantastic Arabica *hasSetting* the preparation of my coffee this morning).

[is setting for](#) is its inverse.

[hasSetting page](#)

is setting for (owl:ObjectProperty)

Inverse property of [has setting](#)

[isSettingFor page](#)

Situation	
Submitted by	ValentinaPresutti
Name	Situation
Also Known As	
Intent	To represent facts, circumstances, observed contexts.
Domains	General
Competency Questions	What entities are in the setting of a certain situation?
Reusable OWL Building Block	http://www.ontologydesignpatterns.org/cp/owl/situation.owl
Consequences	This CP allows the designer to model both a certain situation, and the entities that are involved. It provides designers with a vocabulary for representing n-ary relations.
Scenarios	I prepared a coffee with my heater, 300 ml of water, and an Arabica coffee mix.
Known Uses	
Web References	
Other References	
Examples (OWL files)	http://www.ontologydesignpatterns.org/cp/examples/situation/coffee.owl
Extracted From	http://www.loa-cnr.it/ontologies/DUL.owl
Reengineered From	
Has Components	
Specialization Of	
Related CPs	Submissions:Description

Scenarios about Situation

- I prepared a coffee with my heater, 300 ml of water, and an Arabica coffee mix. >>> [E](#)

Reviews about Situation

There is no review about this proposal. This revision (revision ID **2262**) takes in account the reviews: none

Other info at [evaluation tab](#)

Category: ProposedContentOP

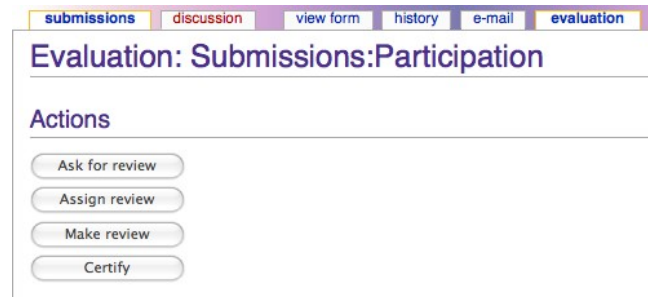


Fig. 3. The WikiFlow evaluation tab on an ODP page.

Ask for Review. This functionality allows users to ask for a review of the current article. Once this action has been performed, the article is automatically assigned to the category `WaitingForReview` that represents the current state of the article.

Assign Review. This functionality allows users to commit another user with the task of reviewing an article. E.g. the ODP *EBMember* users are allowed to perform this action in order to assign the reviewing of a CP proposal to some *QCMember* users.

Make Review. By performing this action, the reviewer faces a review form and can post a review. A new article containing the review is created and named by concatenating the user name of the reviewer and the name of the article under evaluation. If the article under evaluation has the `WaitingForReview` category, this is updated by removing it. E.g. ODP *QCMember* users can perform this action in ODP. A possible future enhancement of this feature would be to make the system aware of additional status of an article associated to an equal number of actions e.g., when an article has received all expected reviews it enters a `WaitingForRevision` status and the system asks the authors to revise the article according to all reviews.

Certify. This functionality allows users to certify an article. Once an article is certified it is *frozen* i.e. it cannot be modified anymore. Furthermore, the evaluation tab of the article reports its evaluation history (see next bullet). The user has the possibility to automatically create a copy of the article. The copy will have a completely separated lifecycle from the previous one e.g. a copy of a certified CP proposal is created in the ODP `Catalogue` namespace and can be modified in order to indicate a persistent identifier for the building block, etc.

Semantic report of evaluation history. The evaluation tab of an article is dynamically updated with its evaluation history. Each review of the article is associated with two article's versions: the one under evaluation and the possible updated version of the article that takes into account the review. The editor of the article can express the association "takes into account" between the revised article and the addressed review, by means of a "one click" functionality. The semantic report of the evaluation history exploits the MediaWiki versioning system. In the context of ODP, our aim is to use the evaluation history as a source for analyzing the rationales behind the evaluation of patterns as well as for extracting good practices and common mistakes e.g. anti-patterns [17] in ontology design.

In order to exemplify the Evaluation WikiFlow extension, we show (see Figure 4) its instantiation in the ODP case. The evaluation tab is used in order to support the certification process of patterns. After posting a CP proposal, an ODP user can submit it to the certification process by asking a review i.e. by clicking in the `ask for review` button in the evaluation tab. The CP proposal is added to the list of CPs that are waiting for reviews (the category `WaitingForReview` is assigned to the CP). Such list is handled by the editorial board.

An *EBMember* user eventually assigns the reviewing task to at least two *QCMember* users. When a review is created, the category of the CP proposal is updated i.e., `WaitingForReview` is removed from the list of the CP's categories, and a link to the review is added in the CP proposal page.

The author of the CP can modify it in order to address the review issues. Once the author believes that a certain version of the CP addresses one or more reviews, (s)he can associate such version to them with the relation `takesIntoAccountReviews`¹⁴.

This process is iterated until reviewers decide agree on the certification of the CP. Certified CPs are copied in the official catalogue i.e the `Catalogue` namespace.

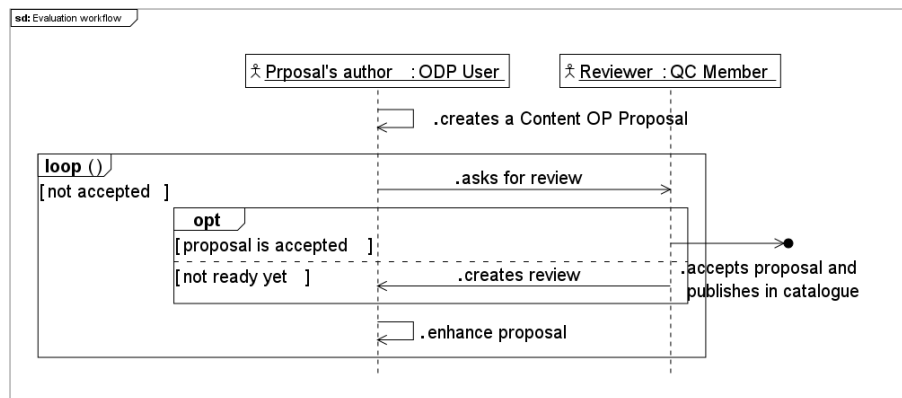


Fig. 4. Evaluation example

5 Related work

In the past few years, there has been lots of research and development effort on good practices of web ontology design. Literature spans from reports of the W3C OEP task force [15, 12], to methodology- and ontology design patterns-oriented books and scientific papers e.g., [16, 14]. Additionally, repositories of reusable components are available on the web¹⁵. In the context of Semantic Web research and applications, ontology

¹⁴ This is a “one click” operation.

¹⁵ E.g. <http://odps.sourceforge.net/>

design patterns (OPs) are now a hot topic. Their notion has been introduced by [5, 18, 19]. Some work [2] has also attempted a learning approach (by using case-based reasoning) to derive and rank patterns with respect to user requirements. The research has also addressed domain-oriented patterns, e.g. for content objects and multimedia [1], software components [13], business modelling and interaction [6, 8], relevance [10] etc. For an historical perspective and a more detailed survey, the reader can refer to e.g. [4, 6, 9, 7].

ODP is about ontology design patterns in the sense of [4] and [16]. It is meant to provide semantic web users with a community-based web portal for training and discussion, associated with a semantic web repository of ontology design patterns. Additionally, evolving requirements of ODP has helped us in developing new functionalities that we release as domain-independent SMW extensions, such as Evaluation WikiFlow, presented here. Wiki-based workflow support is also provided by other existing approaches. For example, a peer-review system has been set for specific works over wikipedia which are going to meet high quality standards¹⁶, in this case the process is not based on semantic features and is not completely automated i.e. the system does not activate actions with respect to certain status, and the focus is not on semantic tracing of rationales. Another tool that supports wiki-based workflows, even if it is not specifically designed for evaluation, is the CICERO SMW extension [3]. It is designed for supporting argumentation based on the argumentation part of the DILIGENT [14] methodology. CICERO might be used as a complementary tool to Evaluation WikiFlow e.g. in case one wants to use the DILIGENT argumentation model in order to support interaction between reviewers i.e. before certify actions in Evaluation WikiFlow.

6 Conclusion and future work

In this paper, we have described ontologydesignpatterns.org (ODP), a semantic web portal supporting the community around best practices of ontology design for the Semantic Web, with particular focus on ontology design patterns (OPs). Starting from ODP requirements of supporting the certification process of patterns, we have developed a SMW extension for supporting the process of wiki articles' evaluation, also presented here and named Evaluation WikiFlow. We have planned a user-based evaluation by monitoring the ODP community activity. ODP ongoing and planned work includes:

- support for new types of ontology design patterns, according to the ontology pattern classification described in [16];
- a keyword-based search service based on Watson [20];
- the ODP repository APIs that will allow to query and select the ODP repository of patterns;
- an OWL/RDF export service;
- augmenting CPs by means of Linking Open Data¹⁷;
- an open rating system for open reviews based on [11];

¹⁶ http://en.wikipedia.org/wiki/Wikipedia:Peer_review

¹⁷ <http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>

- statistical monitoring of CP downloads to be used as a dimension of user-based evaluation of CPs and ODP usage;
- a user-based evaluation of Evaluation WikiFlow based on the ODP certification process.

For additional information on ongoing work on ODP, the reader can refer to the development task page¹⁸.

Acknowledgments We are grateful to the members of the NeOn consortium, who contributed to the NeOn vision being funded by the European Commission 6th IST Framework Programme¹⁹.

References

1. R. Arndt, R. Troncy, S. Staab, L. Hardman, and M. Vacura. Comm: Designing a well-founded multimedia ontology for the web. In *Proceedings of the 4th European Semantic Web Conference (ISCW'07)*, Busan Korea, November 2007. Springer.
2. E. Blomqvist. Fully automatic construction of enterprise ontologies using design patterns: Initial method and first experiences. In R. Meersman, Z. Tari, M.-S. Hacid, J. Mylopoulos, B. Pernici, zalp Babaoglu, H.-A. Jacobsen, J. P. Loyall, M. Kifer, and S. Spaccapietra, editors, *OTM Conferences (2)*, volume 3761 of *Lecture Notes in Computer Science*, pages 1314–1329. Springer, 2005.
3. K. Dellschaft, H. Engelbrecht, J. M. Barreto, S. Rutenbeck, and S. Staab. Cicero: Tracking design rationale in collaborative ontology engineering. In S. Bechhofer, M. Hauswirth, J. Hoffmann, and M. Koubarakis, editors, *ESWC*, volume 5021 of *Lecture Notes in Computer Science*, pages 782–786. Springer, 2008.
4. A. Gangemi. Ontology Design Patterns for Semantic Web Content. In *M. Musen et al. (eds.): Proceedings of the Fourth International Semantic Web Conference*, Galway, Ireland, 2005. Springer.
5. A. Gangemi, C. Catenacci, and M. Battaglia. Inflammation ontology design pattern: an exercise in building a core biomedical ontology with descriptions and situations. In D. M. Pisanelli, editor, *Ontologies in Medicine*. IOS Press, Amsterdam, 2004.
6. A. Gangemi and V. Presutti. Ontology design for interaction in a reasonable enterprise. In P. Rittgen, editor, *Handbook of Ontologies for Business Interaction*. IGI Global, Hershey, PA, November 2007.
7. G. Guizzardi. *Ontological foundations for structural conceptual models*. PhD thesis, University of Twente, Enschede, The Netherlands, Enschede, October 2005.
8. G. Guizzardi and G. Wagner. A unified foundational ontology and some applications of it in business modeling. In *CAiSE Workshops (3)*, pages 129–143, 2004.
9. D. C. Hay. *Data Model Patterns*. Dorset House Publishing, 1996.
10. Juan Gomez-Romero and Fernando Bobillo and Miguel Delgado. An ontology design pattern for representing relevance in owl. In K. Aberer, K.-S. Choi, and N. Noy, editors, *The 6th International Semantic Web Conference and the 2nd Asian Semantic Web Conference 2007*, Busan, Korea, November 2007.
11. H. Lewen. Topic-specific trust and open rating systems: an approach for ontology evaluation. In *Proceedings of WWW'06 4th International EON Workshop Evaluating Ontologies for the Web*, 2006.

¹⁸ <http://ontologydesignpatterns.org/index.php/Odp:Development>

¹⁹ <http://www.neon-project.org>

12. Natasha Noy and Alan Rector. Defining N-ary Relations on the Semantic Web: Use With Individuals. Technical report, W3C, 2005. <http://www.w3.org/TR/swbp-n-aryRelations/> (2004).
13. D. Oberle. *Semantic Management of Middleware*, volume I of *The Semantic Web and Beyond*. Springer, New York, FEB 2006.
14. H. Pinto, S. Staab, and C. Tempich. Diligent: Towards a fine-grained methodology for distributed, loosely-controlled and evolving engineering of ontologies. In *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI 2004)*, Valencia, Spain, 2004.
15. S. W. B. Practices and D. W. Group. Task force on ontology engineering patterns. description of work, archives, w3c notes and recommendations. <http://www.w3.org/2001/sw/BestPractices/OEP/>, 2004.
16. V. Presutti and A. Gangemi. Content Ontology Design Patterns as Practical Building Blocks for Web Ontologies. In *Proceedings of the 27th International Conference on Conceptual Modeling (ER 2008)*, Berlin, 2008. Springer.
17. V. Presutti, A. Gangemi, S. David, G. Aguado de Cea, M. Suarez-Figueroa, E. Montiel-Ponsoda, and M. Poveda. Library of design patterns for collaborative development of networked ontologies. Deliverable D2.5.1, NeOn project, 2008.
18. A. Rector and J. Rogers. Patterns, properties and minimizing commitment: Reconstruction of the galen upper ontology in owl. In A. Gangemi and S. Borgo, editors, *Proceedings of the EKAW*04 Workshop on Core Ontologies in Ontology Engineering*. CEUR, 2004.
19. V. Svatek. Design patterns for semantic web ontologies: Motivation and discussion. In *Proceedings of the 7th Conference on Business Information Systems*, Poznan, 2004.
20. Watson: Semantic web gateway. <http://watson.kmi.open.ac.uk/>.