

Ontology Evaluation and Pattern-based Design

Aldo Gangemi

Semantic Technology Lab, ISTC-CNR, Rome, Italy

aldo.gangemi@cnr.it

@ontolog @ontologysummit

User justifications for exemplary ontologies (from ODP wiki)

- 32 sets of justifications supporting the inclusion of some ontologies in the “Exemplary Ontologies” page of ODP
 - Template designed by Mike Uschold for NeOn ODP
 - <http://ontologydesignpatterns.org/wiki/Ontology:Main>
- Analyzed and classified according to the evaluation framework by Gangemi et al. (2006)
 - See also d’Aquin and Gangemi (2011) on “beautiful” ontologies
 - Classes: syntactic and formal **structure**, conceptual **coverage** and **task**, and **pragmatic** or **social sustainability**
- The most frequent justifications are from the two classes: **task**, and **social sustainability**

Structural justifications

- Reusing foundational ontologies
- Being designed in a principled way
- Being formally rigorous
- Implementing also non-taxonomic relations
- Following strictly an evaluation methodology
- Being modular, or embedded in a modular framework

Conceptual coverage justifications

- Providing important reusable distinctions
- Having a good domain coverage
- Implementing an international standard (e.g., ISO or WHO)
- Providing an organisation to unstructured or poorly structured domains

Conceptual task justifications

- Being oriented at an explicit task
- Having spelled out requirements from scenarios
- Being based on competency questions

Most frequent justifications

Social sustainability justifications

- Being the result of an evolution (many revisions)
- Having wide usage or acceptance
- Having commercial impact
- Being recommended by industry
- Implementing scientific knowledge

Most frequent justifications

Pragmatic sustainability justifications

- Having applications built on top of it
- Having successful personal experience in building apps with it
- Designed for efficient query answering
- Maintaining original expressivity of data, and improving or enriching it
- Able to get rid of clunky constructs or to overcome expressivity limitations
- Being well documented
- Solving other technical aspects

Notions of quality from Ontolog Summit discussion

- What ontology to use?
- How to measure/evaluate?
- Any requirements specified?
- Analogy to QA (requirements, design/production, check against requirements)
- Any (successful) revision story?

Notions of quality from Ontology

Summit discussion -- my contribution

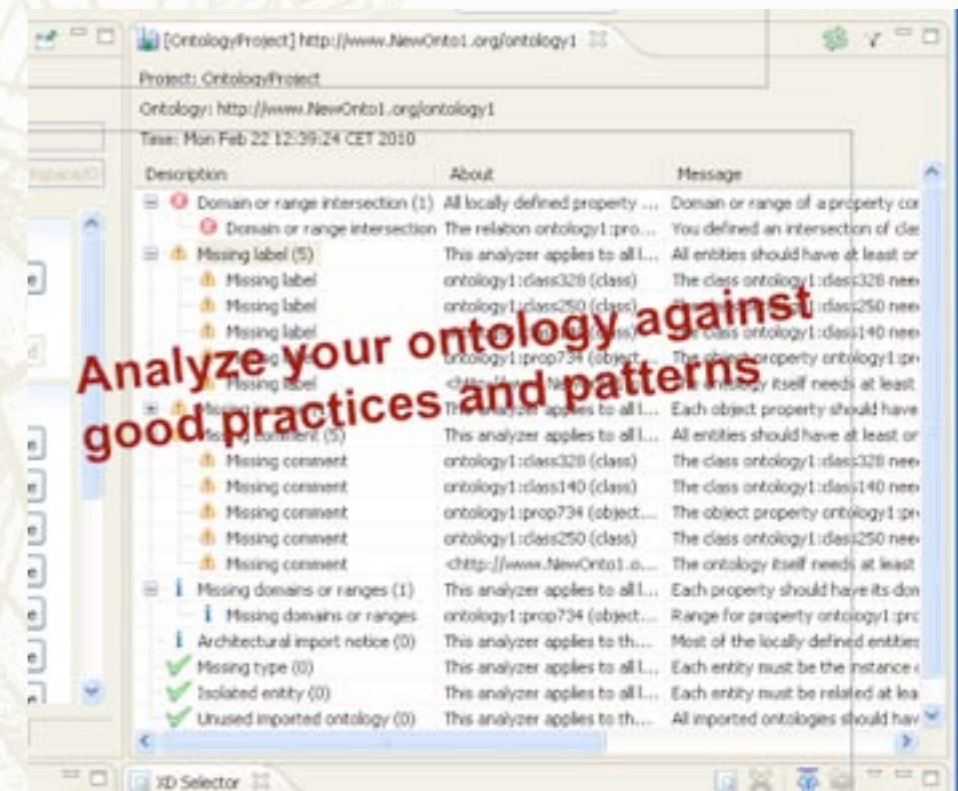
- What ontology to use?
 - Match requirements to repositories (search from Watson, ODP, recommendations, advanced search from XD NeOn Toolkit plugin)
- How to measure/evaluate?
 - Ex-post: decide your principles and metric (QOOD approach)
 - Ex-ante: follow eXtreme Design methods: CQs, patterns, unit tests
- Any requirement specified?
 - Competency questions-based approach
- Analogy to QA (requirements, design/production, check against requirements)
 - Either QOOD, or eXtremeDesign
- Any (successful) revision story?
 - Integration monitoring from eXtreme Design workflow

Quality measure classification

- STLab research from 2004-5: “A formal framework for ontology evaluation and selection”
- Three quality dimensions: Structural-Content-Sustainability
 - Content is the primary dimension
- Content compliance spans Coverage-Task-SelfExplanation
 - Task is the immediately *measurable* aspect
 - Quality is not maximal and abstract, but bound to *context*
 - Partial orders of problems and reusable *solutions*
 - Good practices (*history*)
- Empirical methods for evaluation (measurability)

Examples of structural checking tools

- Graph measures
- Reasoners: HermiT, Pellet, etc.
- LINTs: Pellet, OPPL (custom tests)
 - agghiai-2:pellet-2.2.2 agghiai\$ sh pellet.sh lint -v /Users/agghiai/Workspaces/AllPatterns/dul/DUL.owl
 - No RDF lints found.
 - No OWL 2 DL violations found for ontology <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl>
 - OWL Lints found for ontology <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl>:
 - [EquivalentAndSubclassAxiomPattern: A named concept appears in equivalent axiom(s) and on the left-hand side of a subclass axiom]
 - <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Agent> <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#InformationEntity> <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#InformationRealization> <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#LocalConcept> <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Object> <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Person> <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#PlanExecution> <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#SocialObject> <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#WorkflowExecution>
 - [ExistentialExplosionPattern (MaxTreeSize = 10000): Concepts/Individuals are involved in a large some/min/exact value restrictions tree/loop - maximum recommended number of generated nodes is 10000]
 - - [3.87E10] <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Goal> <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Description> <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Entity> <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Object> <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#PhysicalAgent> <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#SocialObject> ... and 5 more.
- User rank-based measures (e.g. Open Rating System)
- XD Analyzer
 - rule-based anti-pattern tests: typical sources of errors such as domain intersection in properties



Task-oriented ontology design

- Ontologies must match both domain and task
 - Allow the description of the entities (“domain”) whose attributes and relations are concerned because of some purpose
 - *social events* and *agents* as entities that are considered in a *legal case*
 - *research topics* as entities that are dealt with by a *project*, worked on by *academic staff*, and can be topics of *documents*
 - Serve a purpose (“task”)
 - finding entities that are considered *in a same legal case*
 - finding people that *work on a same topic*
 - *matching project topics to staff competencies*, time left, available funds, etc.

Expertise patterns

- Evidence that units of expertise are larger than what we have from average linked data, or worse, ontology learning
 - “Blinking” effects in reacting to events, in evaluating the actions and theories of the others, in understanding context, in interpreting news and ads, etc.
 - Competency questions (Gruninger, 1994) try to convey these units as requirements
 - *Which objects take part in a certain event?*
 - *Which tasks should be executed in order to achieve a certain goal?*
 - *What’s the function of that artifact?*
 - *Does this behaviour conform to a certain rule?*
 - *What norms are applicable to a certain case?*
 - *What norm is superordinated among these ones?*
 - Sometimes exception conditions should be added
 - Task-based ontology evaluation can be performed with unit tests against ontologies trying to satisfy competency questions

Ontology Design Patterns

An ontology design pattern is a reusable successful solution to a recurrent modeling problem

---->

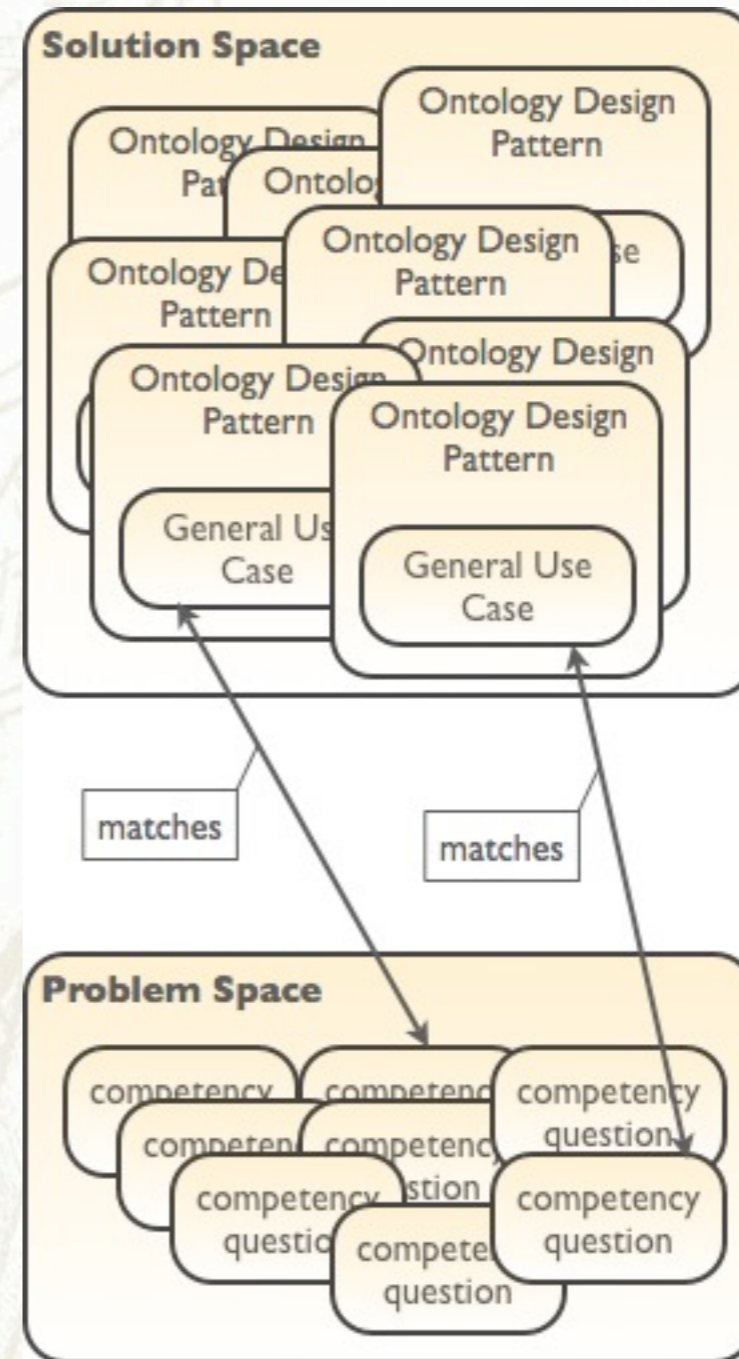
Gruninger on Competency Questions

Alexander on Design Patterns

Clark on Knowledge Patterns

W3C SWBPD WG

ODP movement, cf. ontologydesignpatterns.org



Pattern-based design aka eXtreme Design (XD)

- Pattern-based ontology design is the activity of searching, selecting, composing different patterns, and performing testing and integration of pair-design sessions (see later)
 - Logical, Content (or Knowledge), Reasoning, Architectural, Naming, Correspondence, Reengineering
 - Common framework to understand modeling choices (the “solution space”) wrt task- and domain-oriented requirements (the “problem space”)
 - <http://www.ontologydesignpatterns.org>

Catalogues of ODP 1/2


- ODPs are collected and described in catalogues and comply to a common presentation template
- The ontologydesignpatterns.org initiative maintains a repository of ODPs and a semantic wiki for their description, discussion, evaluation, certification, etc.

Catalogues of ODP 2/2

ontology design patterns . org (odp) [discussion](#) [view source](#) [history](#)

Ontology Design Patterns . org (ODP)








OntologyDesignPatterns.org is a Semantic Web portal dedicated to ontology design patterns (ODPs).
The portal was started under the [NeOn project](#), which still partly supports its development.









What's new

- The 2nd Workshop on Ontology Patterns to be held on November 8, in conjunction with ISWC2010. **Submission deadline extended to September 1st!**
- eXtreme Design camp in Bologna


Navigation

-  **List of Patterns**
You can find lists here, detailing all available ontology design patterns.
-  **Pattern types**
Ontology patterns are of several types. Here are details about pattern types and their taxonomy.
-  **Domains**
Ontology patterns can cover, or be related to, a particular domain. Here is a list.
-  **Modeling Issues**
See all loaded modeling issues. Modeling issues are linked to ontology patterns that solve a defined problem.
-  **Training Area**
Learn about ODPs!
-  **Events**
See a list of events related to ontology design patterns.
-  **Reviews**
Here you can browse both open reviews and quality committee reviews.

Contribute

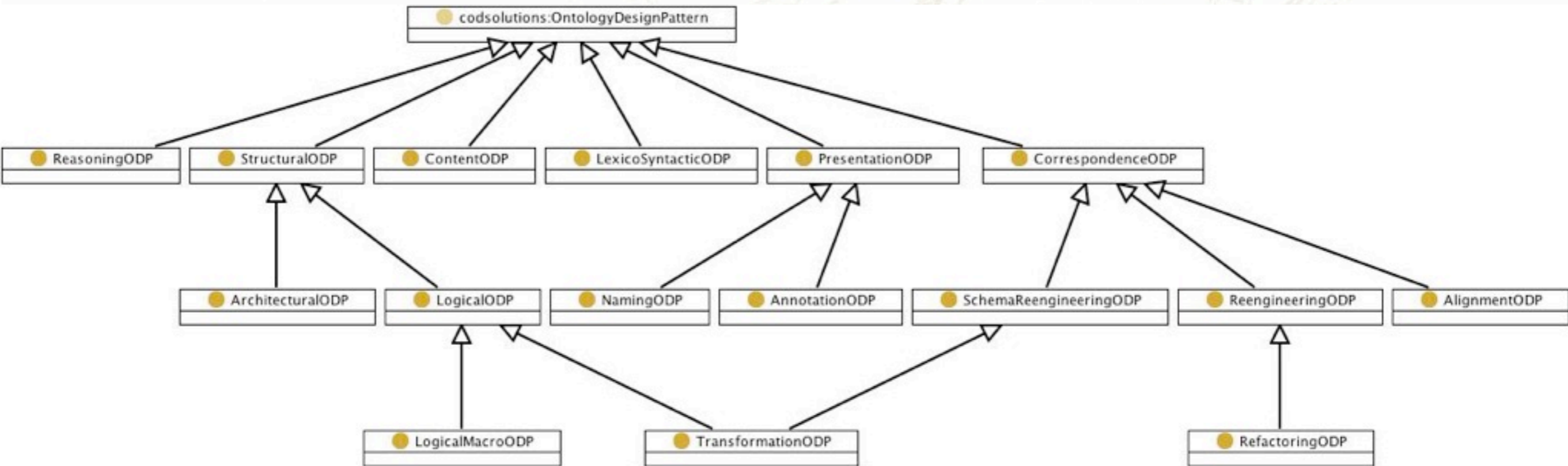
-  **Submit Pattern**
Start here if you want to submit an ontology pattern.
-  **Post Modeling Issue**
If you have an unsolved modeling problem you wish to share with the community, post it here!
-  **Submit an Exemplary Ontology**
Start here if you want to submit an exemplary ontology.
-  **Post Review About a Pattern**
Review a submission to contribute to the certification process.
-  **Post Your Feedback**
If you have issues about the web site, can't find information you need, or simply wish to propose enhancements, you can give feedback here about the ODP portal.
-  **Request Account**
To make changes to the ODP wiki portal, you need to be logged in...

News

 **Latest ODP News!**

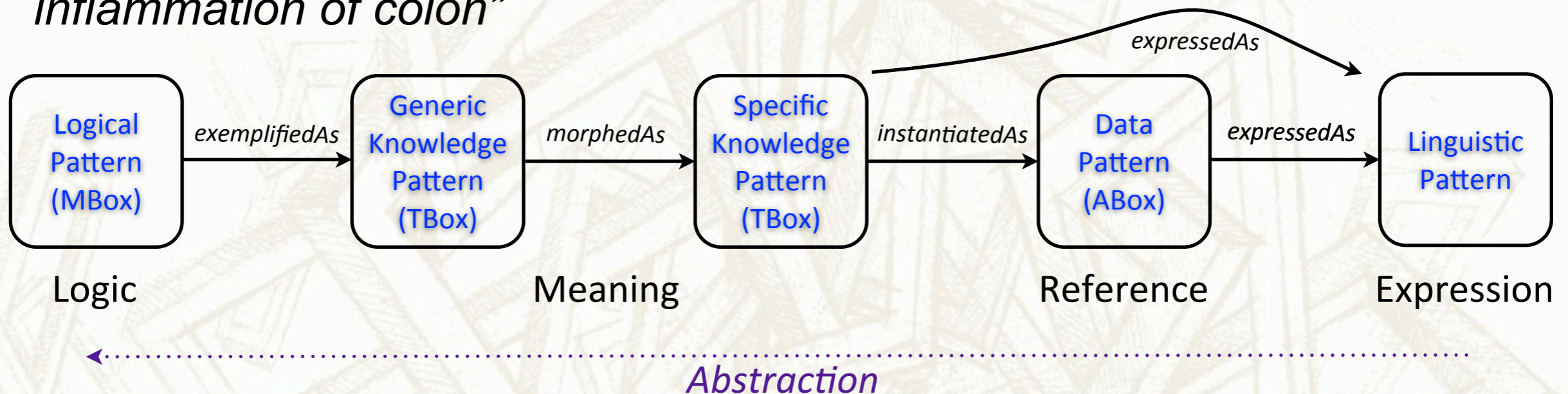
- [2nd Workshop on Ontology Patterns \(WOP\) accepted at ISWC 2010](#)
29 May 2010 12:12:43 - by [EvaBlomqvist](#)
- [VOCamp in Paris - #vocampparis2010](#)
6 April 2010 13:13:28 - by [FrancoisScharffe](#)
- [Collaborative eXtreme Design Camp in Bologna](#)
13 February 2010 13:13:39 - by [AldoGangemi](#)
- [New pattern type: Lexico-syntactic ODPs](#)
8 February 2010 10:10:24 - by [EnricoDaga](#)
- [Vocamp @ Washington D.C. supported by WOP](#)
30 June 2009 15:15:34 - by [ValentinaPresutti](#)
- [Workshop on Ontology Patterns \(WOP\) accepted at ISWC 2009!](#)
23 May 2009 11:11:32 - by [EvaBlomqvist](#)
- [The Loreley of Ontology Design Patterns](#)
2 April 2009 10:10:52 - by [VioletaDamjanovic](#)
- [EvalWF has been released](#)
21 October 2008 12:12:59 - by [EnricoDaga](#)

Types of ODPs



Logical vs. Knowledge patterns

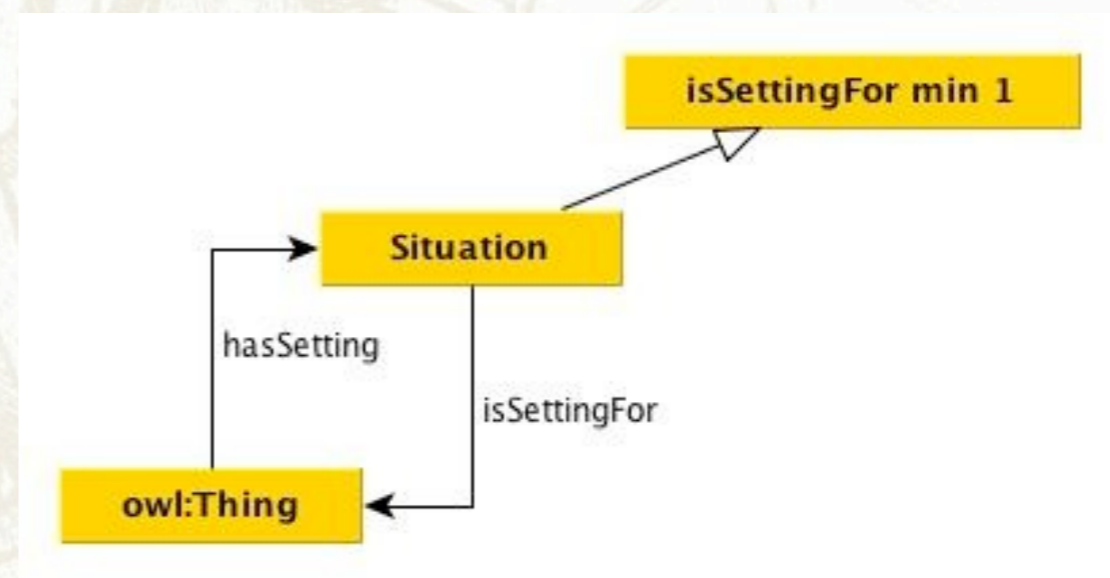
- A Logical ODP describes a formal expression that can be *exemplified, morphed, instantiated, and expressed* in order to solve a domain modelling problem
- **owl:Class:_:x rdfs:subClassOf owl:Restriction:_:y**
- *Inflammation rdfs:subClassOf (localizedIn some BodyPart)*
- *Colitis rdfs:subClassOf (localizedIn some Colon)*
- *John's_colitis isLocalizedIn John's_colon*
- *“John's colon is inflamed”, “John has got colitis”, “Colitis is the inflammation of colon”*



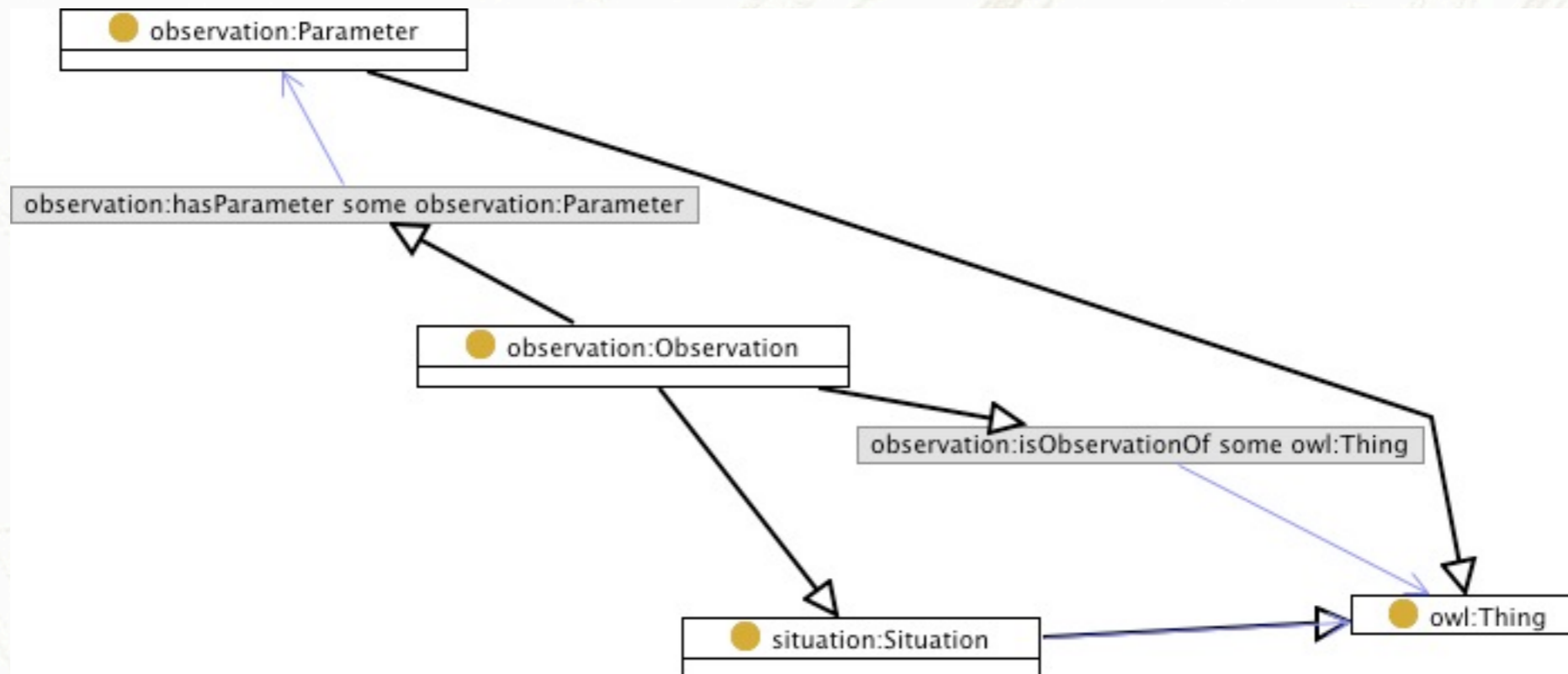
Sample Knowledge Patterns aka Content Patterns

Situation

- A general vocabulary for n-ary relations
- *Situation* abstracts from reified n-ary relations, by defining a top-level relation for all binary projections of the n-ary relation
- A way to conceive a state of affairs, a set of things, a fact
- All time indexed (and place indexed) patterns are (in principle) specializations of *Situation*



The *Observation* knowledge pattern



Continuity between logical and knowledge patterns

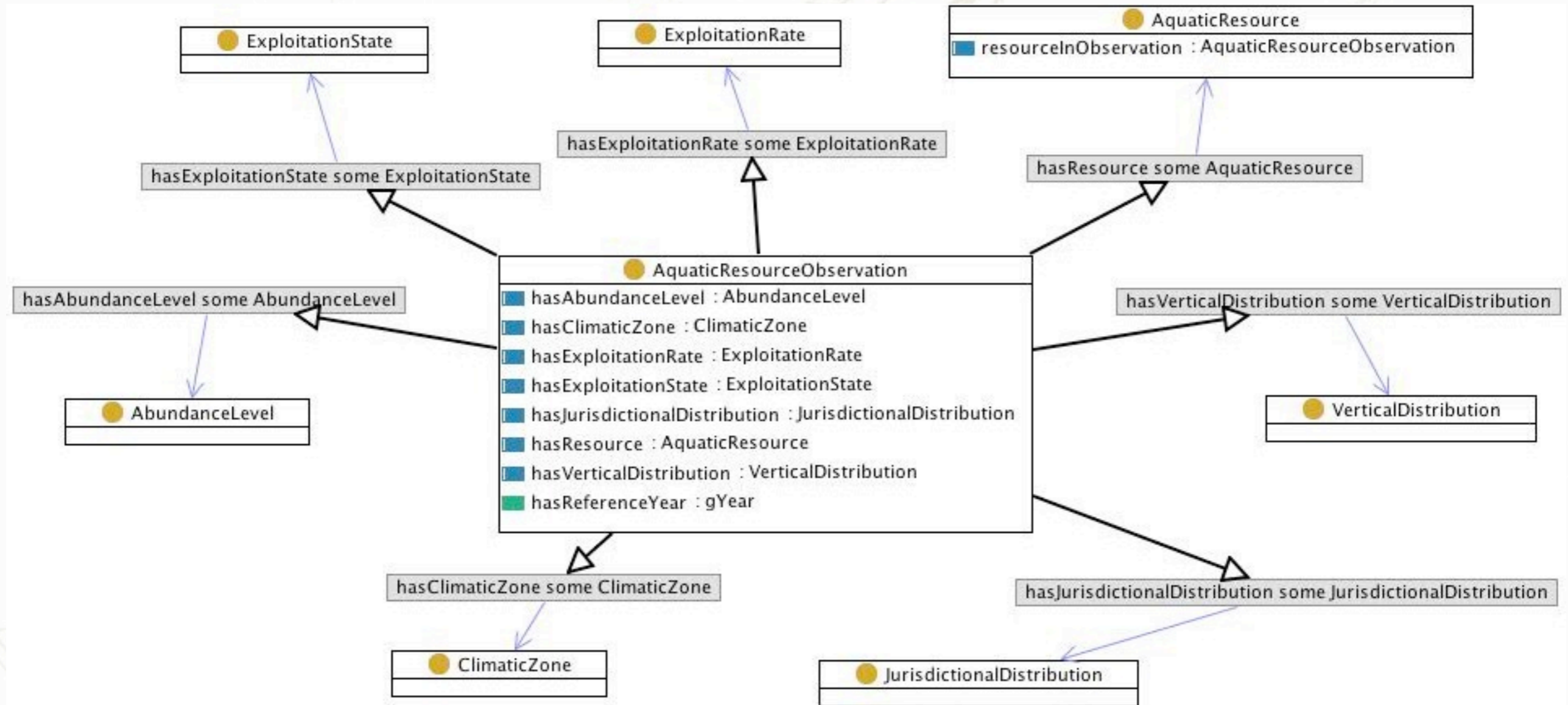
Signature introduction ->

Signature morphism ->

Adaptation ->

Evaluation against competency questions

Aquatic Resource Observation



Time Interval

● TimeInterval
■ hasIntervalDate : date
■ hasIntervalEndDate : date[0..1]
■ hasIntervalStartDate : date[0..1]

Elements

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

Time Interval (owl:Class)

Any region in a dimensional space that represents time.

[TimeInterval page](#)

has interval date (owl:DatatypeProperty)

A datatype property that encodes values from xsd:date for a time interval; a same time interval can have more than one xsd:date value: begin date, end date, date at which the interval holds, as well as dates expressed in different formats: xsd:gYear, xsd:dateTime, etc.

[hasIntervalDate page](#)

has interval start date (owl:DatatypeProperty)

The start date of a *time interval*.

[hasIntervalStartDate page](#)

has interval end date (owl:DatatypeProperty)

The end date of a *time interval*.

[hasIntervalEndDate page](#)

TimeInterval

Submitted by [ValentinaPresutti](#)

Name time interval

Also Known As

Intent To represent time intervals.

Domains [Time](#)

Competency Questions What is the end time of this interval?, What is the starting time of this interval?, What is the date of this time interval?

Reusable OWL Building Block <http://www.ontologydesignpatterns.org/cp/owl/timeinterval.owl>

Building Block

Consequences The dates of the time interval are not part of the domain of discourse, they are datatype values. If there is the need of reasoning about dates this Content OP should be used in composition with the [region](#) Content OP.

Scenarios The time interval "January 2008" starts at 2008-01-01 and ends at and ends at 2008-01-31.

Known Uses

Web

References

Other

References

Examples (OWL files) <http://www.ontologydesignpatterns.org/cp/examples/timeinterval/january2008.owl>

Extracted From

Reengineered

From

Has

Components

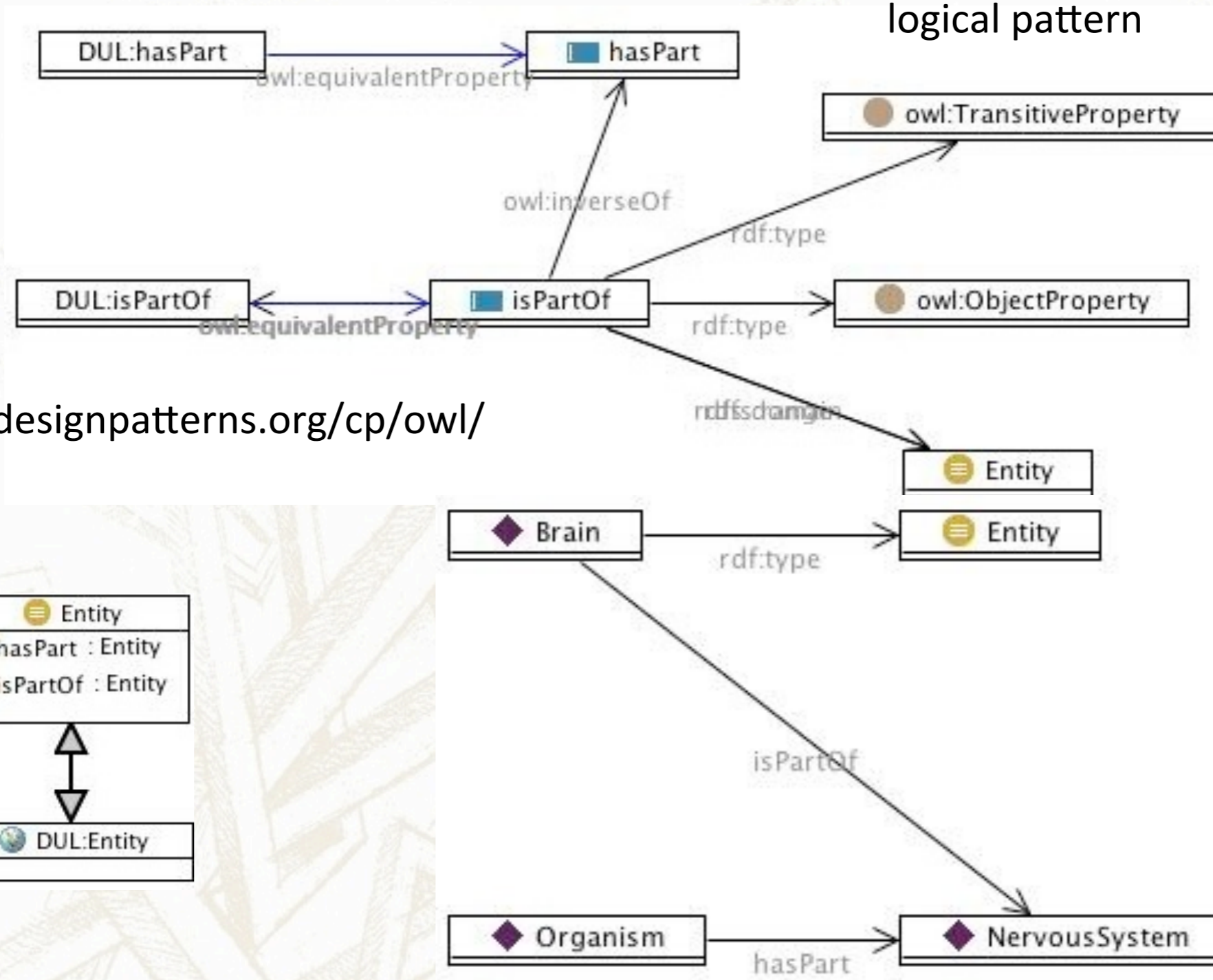
Specialization

Of

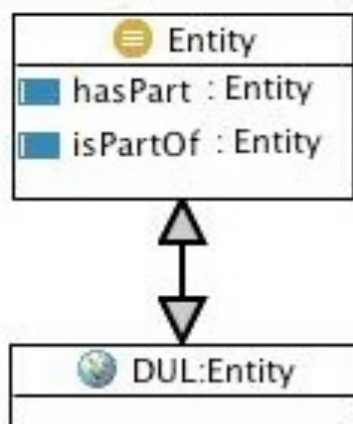
Related CPs

Parthood

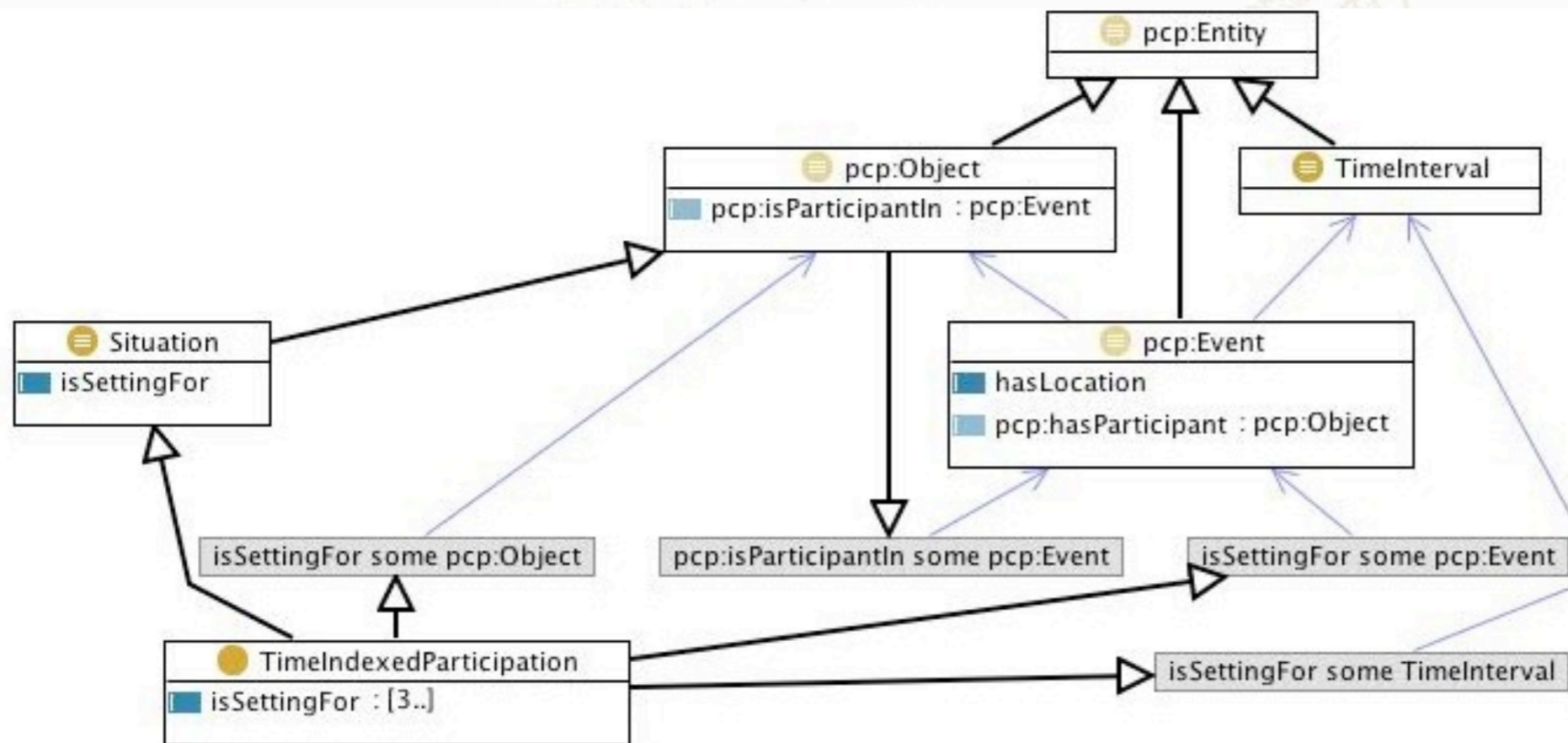
This also uses transitivity reasoning logical pattern



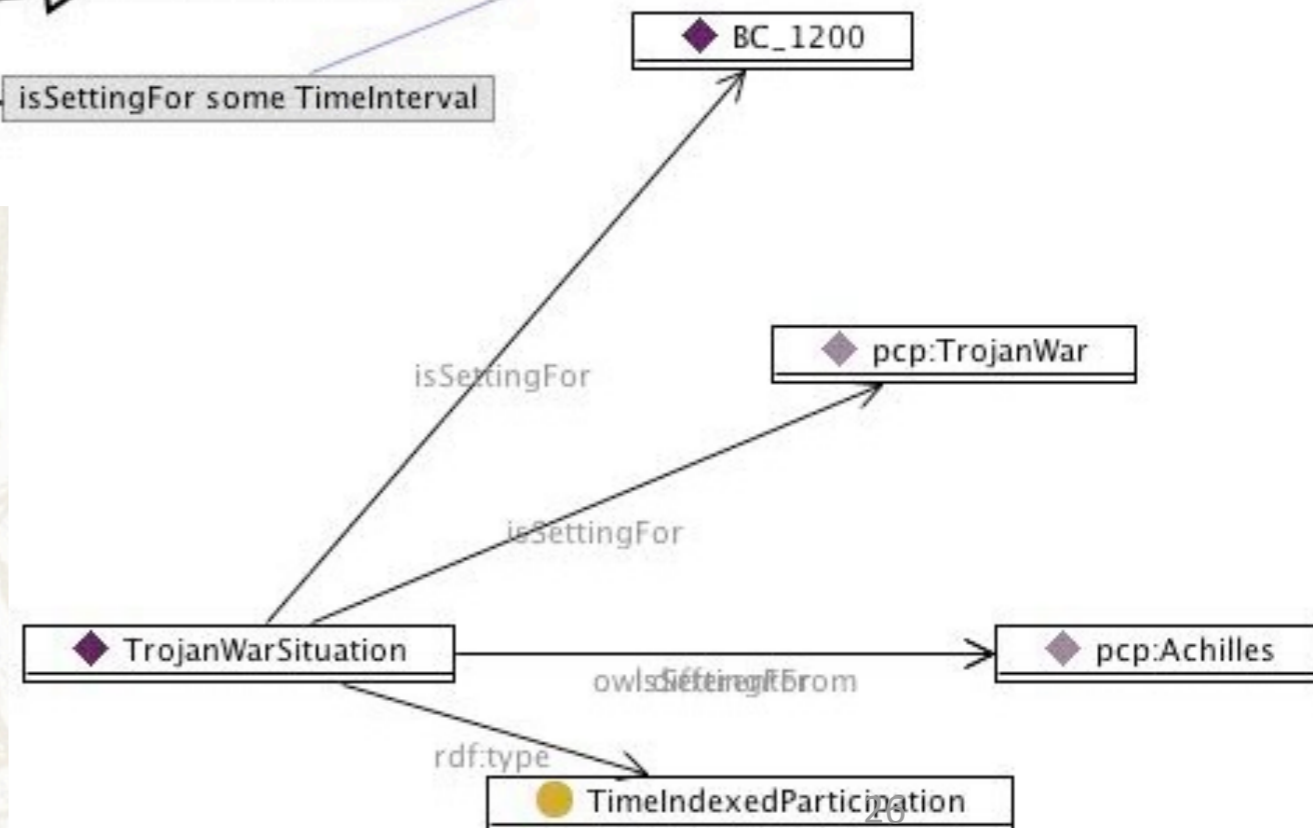
Cf. <http://www.ontologydesignpatterns.org/cp/owl/partOf.owl>



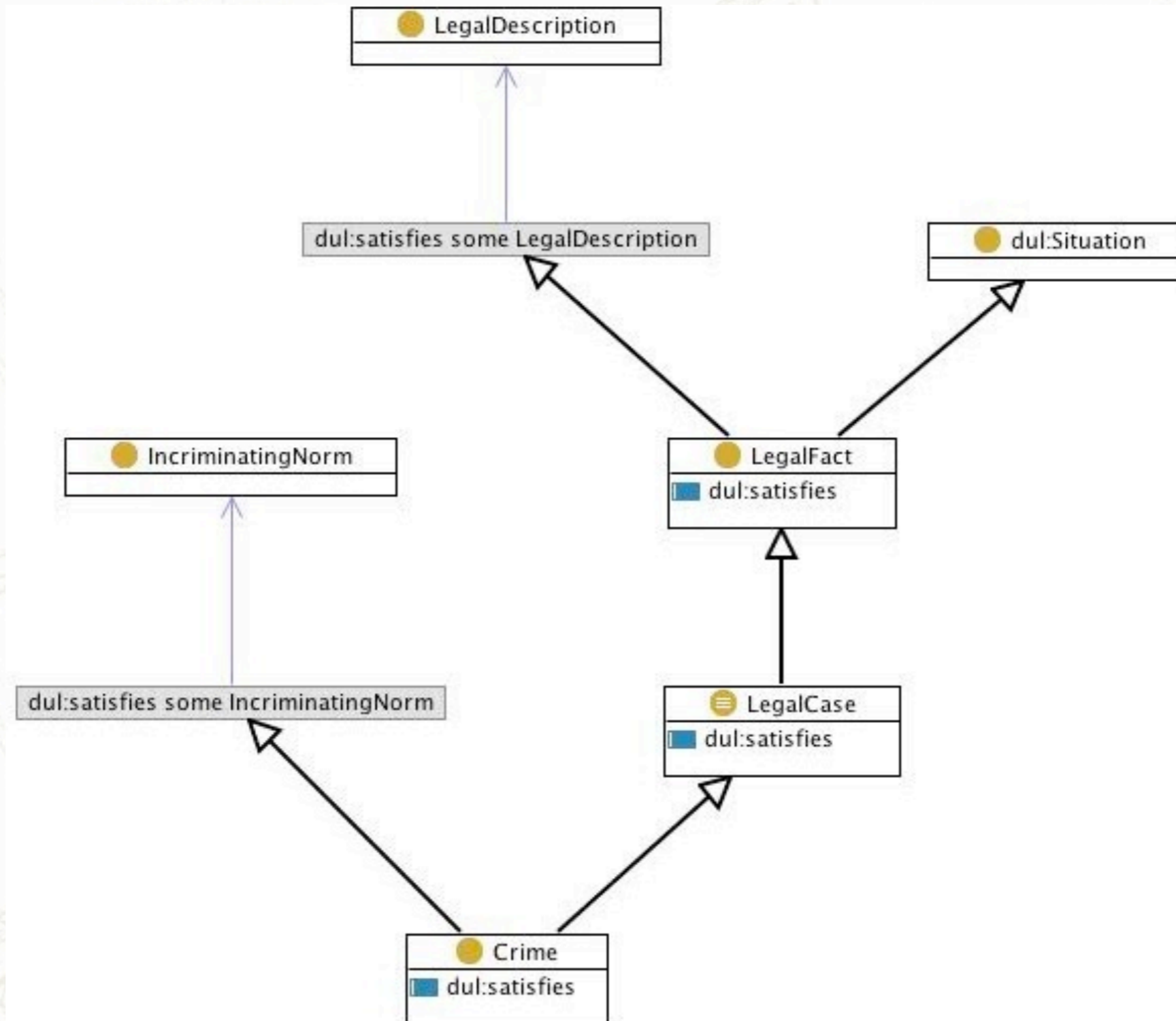
Time-indexed Participation



This also uses N-ary logical pattern



Crime



Anti-patterns (1/2)

- Partonomies or subject classifications as subsumption hierarchies
 - `*City subclassOf Country`
 - `City subclassOf (partOf some Country)`
 - `*City subclassOf Geography`
 - `City broader Geography (e.g. in SKOS)`
- Linguistic disjunction as class disjointness
 - Dead or alive
 - `*Dead or Alive`
 - `Dead disjointWith Alive`
- Linguistic conjunction as class disjunction
 - Pen and paper
 - `*Pen and Paper`
 - `Pen or Paper | Collection subclassOf (hasMember some Paper ; some Pen)`

Anti-patterns (2/2)

- Causality as entailment
 - Kaupthing bank behavior caused Iceland crisis
 - `*KaupthingBankBehavior subclassOf IcelandCrisis`
 - `KaupthingBankBehavior isCauseOf IcelandCrisis`
- Expressions as instances of the class representing their meaning
 - `*dog(word) rdf:type Dog`
 - `dog(word) expresses Dog (with punning)`
- Multiple domains or ranges of properties as intersection
 - `*hasInflammation rdfs:domain Epithelium ; Endothelium`
 - `hasInflammation rdfs:domain (Epithelium or Endothelium)`

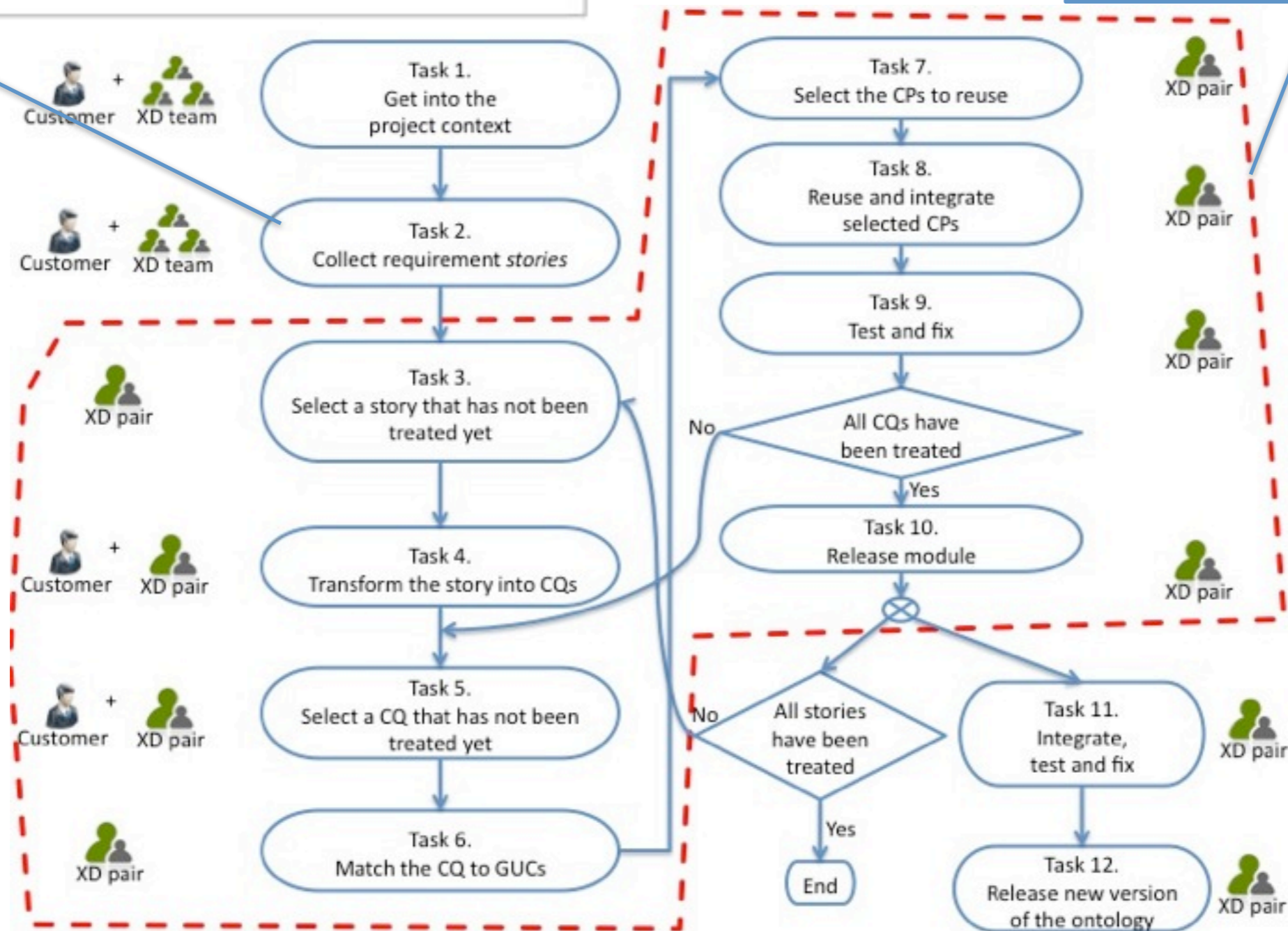


eXtreme Design

XD iteration

Tool support:
matching and selection
of patterns are perceived
to be
the most difficult tasks

Title	Tuna observation
Depends on	Exploitation values, Tuna areas
Description	In 2004 the resource of species "Tuna" in water area 24 was observed to be fully exploited in the tropical zone at pelagic depth.
Priority	High



Method and tool support

- eXtreme Design (XD)
 - a method for developing ontologies with Content Patterns
- XD tool
 - a tool that supports XD method
 - released as both an Eclipse plugin and a NeOn Toolkit plugin
 - We will use it with the NeOn toolkit

<http://neon-toolkit.org/wiki/Download/2.3.2>

Method 1 – Unit tests

- Verifying that the ontology supports retrieval of information corresponding to the CQs
- Write one unit test corresponding to each CQ of the ontology (module)
 - Unit test? – SPARQL query
- What kind of errors?
 - Mainly missing elements
 - Violation of modeling best practices

Method 2 – Performing inferences

- Verifying that the ontology supports the reasoning tasks required
- Create one test case for each inference to be made (according to the requirements)
 - Test case? – Set of “input” facts that should produce the desired output of inference
- What kind of errors?
 - Unfulfilled reasoning requirements
 - Unexpected side-effects
 - Inconsistencies

Method 3 – Performing “stress tests”

- Verify that the ontology enforces the contextual statements and is robust against unexpected or erroneous data
- Create one test case for each contextual statement and “boundary value” to test
 - Test case? – Data that should produce some errors, e.g. inconsistency, in the ontology
- What kind of errors?
 - Unfulfilled or missing contextual statements, i.e. missing axioms
 - Implicit constraints that should be made explicit
 - Unexpected side-effects

Other ways of testing?

- **Inspection**
 - Checking coverage
 - terminology
 - axioms
 - Checking against best practices, e.g. ODPs
- “Peer review”
- ...

eXtreme Design - NeOn Toolkit - C:\Programmi\NeOn\NeOnToolkit-2.3.1-B264-base-win32\NeOnToolkit\workspace

File Edit Navigate Search Project Window Help

Ontology Navigator

- cpannotationschema.owl
- objectrole.owl
- http://www.NewOnto1.org/ontology1
 - Classes
 - <http://www.NewOnto1.org/ontology1/class250
 - class250
 - class328
 - Object Properties
 - <http://www.NewOnto1.org/ontology1/prop734
 - prop734
 - Data Properties
 - Annotation Properties
 - <http://www.ontologydesignpatterns.org/schemas/agentrole.owl#agentrole
 - <http://www.ontologydesignpatterns.org/schemas/objectrole.owl#objectrole
 - <http://www.ontologydesignpatterns.org/schemas/taskrole.owl#taskrole
 - <http://www.ontologydesignpatterns.org/schemas/classification.owl#classification
 - <http://www.ontologydesignpatterns.org/schemas/participation.owl#participation
 - <http://www.ontologydesignpatterns.org/schemas/timeinterval.owl#timeinterval
 - <http://www.ontologydesignpatterns.org/schemas/latentindexing.owl#latentindexing

Entity Properties

OWL Ontology

URI: http://www.NewOnto1.org/ontology1

Location: file:/C:/Programmi/NeOn/NeOnToolkit-2.3.1-B264-base-win32/NeOnToolkit/workspace/O

Imports

http://www.ontologydesignpatterns.org/cp/owl/classification.owl Remove

Add to imports:

ontology to import Add

Imports and Namespaces

Prefix	URI	Edit	Remove
owl	http://www.w3.org/2002/07/owl#	Edit	Remove
xsd	http://www.w3.org/2001/XMLSchema#	Edit	Remove
swrl	http://www.w3.org/2003/11/swrl#	Edit	Remove
swrlb	http://www.w3.org/2003/11/swrlb#	Edit	Remove

[OntologyProject] http://www.NewOnto1.org/ontology1

Project: OntologyProject

Ontology: http://www.NewOnto1.org/ontology1

Time: Mon Feb 22 12:39:24 CET 2010

Description	About	Message
Domain or range intersection (1)	All locally defined property ...	Domain or range of a property cor
Domain or range intersection	The relation ontology1:pro...	You defined an intersection of cla
Missing label (5)	This analyzer applies to all l...	All entities should have at least o
Missing label	ontology1:class328 (class)	The class ontology1:dass328 nee
Missing label	ontology1:class250 (class)	The class ontology1:dass250 nee
Missing label	ontology1:prop734 (object...	The object property ontology1:pr
Missing label	<http://www.NewOnto1.o...	The ontology itself needs at least
Missing inverse	This analyzer applies to all l...	Each object property should have
Missing comment (5)	This analyzer applies to all l...	All entities should have at least o
Missing comment	ontology1:class328 (class)	The class ontology1:dass328 nee
Missing comment	ontology1:class140 (class)	The class ontology1:dass140 nee
Missing comment	ontology1:prop734 (object...	The object property ontology1:pr
Missing comment	ontology1:class250 (class)	The class ontology1:dass250 nee
Missing comment	<http://www.NewOnto1.o...	The ontology itself needs at least
Missing domains or ranges (1)	This analyzer applies to all l...	Each property should have its don
Missing domains or ranges	ontology1:prop734 (object...	Range for property ontology1:pr
Architectural import notice (0)	This analyzer applies to th...	Most of the locally defined entities
Missing type (0)	This analyzer applies to all l...	Each entity must be the instance c
Isolated entity (0)	This analyzer applies to all l...	Each entity must be related at lea
Unused imported ontology (0)	This analyzer applies to th...	All imported ontologies should hav

ODP Registry

Content ODP Submissions

- Submissions
 - AgentRole
 - Biological_Entities
 - Classification
 - Co-participation
 - CollectionEntity
 - Componency
 - Constituency
 - Description
 - GO_Top
 - HasPest
 - Information_realization
 - Invoice
 - Metonymy-species-commodity

ODP Details

URI: http://aims.fao.org/aos/v1.0/

Location: http://www.aims.fao.org/aos#hasPest

Property	Value	Lang
versionInfo	"20091021"	
comment	"Last revised by KCEW 20080508: removed all ..."	en
comment	"Last revised by KCEW 2008: set the correct n..."	en
comment	"Last revised by KCEW 20090730: added some..."	en
comment	"Last revised by KCEW 20090821: " / " URI inst..."	en
comment	"Last revised by KCEW 20090828: edited with ..."	en
comment	"Last revised by KCEW 20090924: changed na..."	en
comment	"Last revised by KCEW 20090929: added hasS..."	en
comment	"Last revised by KCEW 20091002: added all rel..."	en
comment	"Last revised by KCEW 20091021: completed r..."	en
comment	"TODO: remove chemical and geographical con..."	en

XD Selector

Indexing (7)

- /agentrole.owl [http://www.ontologydesignpatterns.org/cp/owl/agentrole.owl]
 - Physical URI: http://www.ontologydesignpatterns.org/cp/owl/agentrole.owl
 - Score: 0.41632402
- /objectrole.owl [http://www.ontologydesignpatterns.org/cp/owl/objectrole.owl]
- /taskexecution.owl [http://www.ontologydesignpatterns.org/cp/owl/taskexecution.owl]
- /taskrole.owl [http://www.ontologydesignpatterns.org/cp/owl/taskrole.owl]
- /classification.owl [http://www.ontologydesignpatterns.org/cp/owl/classification.owl]
- /participation.owl [http://www.ontologydesignpatterns.org/cp/owl/participation.owl]
- /timeinterval.owl [http://www.ontologydesignpatterns.org/cp/owl/timeinterval.owl]
- Latent indexing (37)
 - /agentrole.owl [http://www.ontologydesignpatterns.org/cp/owl/agentrole.owl]
 - /objectrole.owl [http://www.ontologydesignpatterns.org/cp/owl/objectrole.owl]
 - /taskrole.owl [http://www.ontologydesignpatterns.org/cp/owl/taskrole.owl]

Specialize, compose, annotate ODPs and ontologies

Analyze your ontology against good practices and patterns

Browse, search, and get Content ODPs

Experimental evidence (I)

- Content patterns improve the quality of ontologies
 - Experiments with master and PhD students
 - Quality measured in terms of
 - task-coverage
 - error-freedom
 - subjective perception of smooth and good design
- *Published at KCAP09*

Experimental evidence (II)

- eXtreme Design method further improves quality and also improves coverage of the proposed requirements
 - Experiment with 7 designer pairs (PhD students)
- *Published at EKAW2010*

Experimental evidence (III)

- ODP-based ontology learning improves results
- Ontologies are better in terms of cohesion, consistency, functional quality, etc.
- Experiment with OntoCase applied to Text2Onto ontology learning
- *Published at ISWC2009*

Training material on ODP

- <http://ontologydesignpatterns.org/wiki/Training:Main>
- Latest material on eXtreme Design from 2011 PhD course
 - http://ontologydesignpatterns.org/wiki/Training:PhD_Course_on_Computational_Ontologies_%40_University_of_Bologna_2011

References

- Gangemi, A, C. Catenacci, M. Ciaramita, J. Lehmann. Modelling Ontology Evaluation and Validation. Proceedings of ESWC 2006.
- Gruninger, M., and Fox, M.S.: The Role of Competency Questions in Enterprise Engineering. Proceedings of the IFIP WG5.7 Workshop on Benchmarking - Theory and Practice, Trondheim, Norway (1994).
- Clark, P., Thompson, J., Porter, B.: Knowledge Patterns. KR2000 (2000).
- Welty, C.: Semantic Web Best Practices and Deployment Working Group, Task Force on Ontology Engineering Patterns. Description of work, archives, W3C Notes and recommendations available from <http://www.w3.org/2001/sw/BestPractices/OEP/> (2004-5).
- Gangemi, A. Ontology Design Patterns for Semantic Web Content. Musen et al. (eds.): Proceedings of the Fourth International Semantic Web Conference, Galway, Ireland, 2005. Springer.
- Gangemi, A., V. Presutti. Ontology Design Patterns. Staab et al. (eds.): Handbook of Ontologies (2nd Edition), Springer, 2010.
- Blomqvist E., Gangemi A., Presutti V. Experiments in Pattern-based Ontology Design, Proceedings of KCAP09, Los Angeles, ACM Press, 2009.
- Blomqvist E., Gangemi A., Daga E., Presutti V. Experimenting with eXtreme Design. P. Cimiano and S: Pinto (eds.): Proceedings of the Conference on Knowledge Engineering and Knowledge Management (EKAW2010), Springer, 2010.
- Blomqvist E., OntoCase-Automatic Ontology Enrichment Based on Ontology Design Patterns. In: The Semantic Web - ISWC 2009, 8th International Semantic Web Conference, ISWC 2009, Chantilly, VA, USA, October 25-29, 2009. Proceedings. pp. 65-80, Lecture Notes in Computer Science, Springer, 2009
- d'Aquin M., Gangemi A. Is there beauty in ontologies? Applied Ontology, 3, 2011.