Ontology Design Patterns

Sergiu Rafiliu

Department of Computer and Information Science,
Linköping University, Sweden

June 5, 2011

Ontology Design Patterns

Experience in designing ontologies	$\Rightarrow$	Emergence of *patterns*.
Software Engineering Design Patterns	$\Leftrightarrow$	Ontology Design Patterns

Ontology Design Patterns

Goal:

Quickly design ontologies by combining Design Patterns, rather than rethinking everything from ground up.

Software Engineering Design Problem

Slot	Value
Type	UI form
Examples	• Tax forms • Job application forms • Ordering merchandise through a catalog
Context	The user has to provide preformatted information, usually short (non-narrative) answers to questions
Problem	How should the artifact indicate what kind of information should be supplied, and the extent of it?
Forces	• The user needs to know what kind of information to provide. • It should be clear what the user is supposed to read, and what to fill in. • The user needs to know what is required, and what is optional. • Users almost never read directions. • Users generally do not enjoy supplying information this way, and are satisfied by efficiency, clarity, and a lack of mistakes.
Solution	Provide appropriate "blanks" to be filled in, which clearly and correctly indicate what information should be provided. Visually indicate those editable blanks consistently, such as with subtle changes in background color, so that a user can see at a glance what needs to be filled in. Label them with clear, short labels that use terminology familiar to the user; place the labels as close to the blanks as is reasonable. Arrange them all in an order that makes sense semantically, rather than simply grouping things by visual appearance

Ontology Design Problem

Slot	Value
General issue	It is often convenient to put a class (e.g., Animal) as a property value (e.g., topic or book subject) when building an ontology. While OWL Full and RDF Schema do not put any restriction on using classes as property values, in OWL DL and OWL Lite most properties cannot have classes as their values.
Use case example	Suppose we have a set of books about animals, and a catalog of these books. We want to annotate each catalog entry with its subject, which is a particular species or class of animal that the book is about. Further, we want to be able to infer that a book about African lions is also a book about lions. For example, when retrieving all books about lions from a repository, we want books that are annotated as books about African lions to be included in the results.
Notation	In all the figures below, ovals represent classes and rectangles represent individuals. The orange color signifies classes or individuals that are specific to a particular approach. Green arrows with green labels are OWL annotation properties. We use N3 syntax to represent the examples.
Approaches	Approach 1: Use classes directly as property values In the first approach, we can simply use classes from the subject hierarchy as values for properties (in our example, as values for the dc:subject property). We can define a class Book to represent all books.
Considerations	• The resulting ontology is compatible with RDF Schema and OWL Full, but it is outside OWL DL and OWL Lite. • This approach is probably the most succinct and intuitive among all the approaches proposed here. • Applications using this representation can directly access the information needed to infer that Lion (the subject of the LionsLifeInThePrideBook individual) is a subclass of Animal and that AfricanLion (the subject of the TheAfricanLionBook individual) is a subclass of Lion.
OWL code (N3 syntax)	default:BookAboutAnimals a owl:Class ; rdfs:subClassOf owl:Thing ; rdfs:subClassOf [a owl:Class ; owl:unionOf ([a owl:Restriction ;

Describing an Ontology Design Problem

Generic Use Cases – Competency Questions

- Who does **what**, **when** and **where**?
- Which objects **take part in** a certain event?
- What are the **parts** of something?
- What's an object **made of**?
- What's the **place** of something?
- What's the **time** frame of something?
- What **technique, method, practice** is being used?
- Which **tasks** should be executed in order to achieve a certain goal?
- Does this behaviour **conform** to a certain rule?
- What's the **function** of that artifact?
- How is that object **built**?
- What's the **design** of that artifact?
- How did that phenomenon **happen**?
- What's your **role** in that transaction?
- What that information **is about**? How is it **realized**?
- What **argumentation model** are you adopting for negotiating an agreement?
- What's the **degree of confidence** that you give to this axiom?

Conceptual Ontology Design Pattern

Is

- A formal pattern that encodes a Generic Use Case.
- A template to represent and solve a modelling problem.
- Described by
  - An intuitive set of features
  - A minimal semantic characterization and its formal encoding

Should

- Have a compact description.
- Be composable: an element in a design pattern can be expaned with the help of another design pattern.
- Follow best practices in ontology design.

Conceptual Ontology Design Pattern

Is

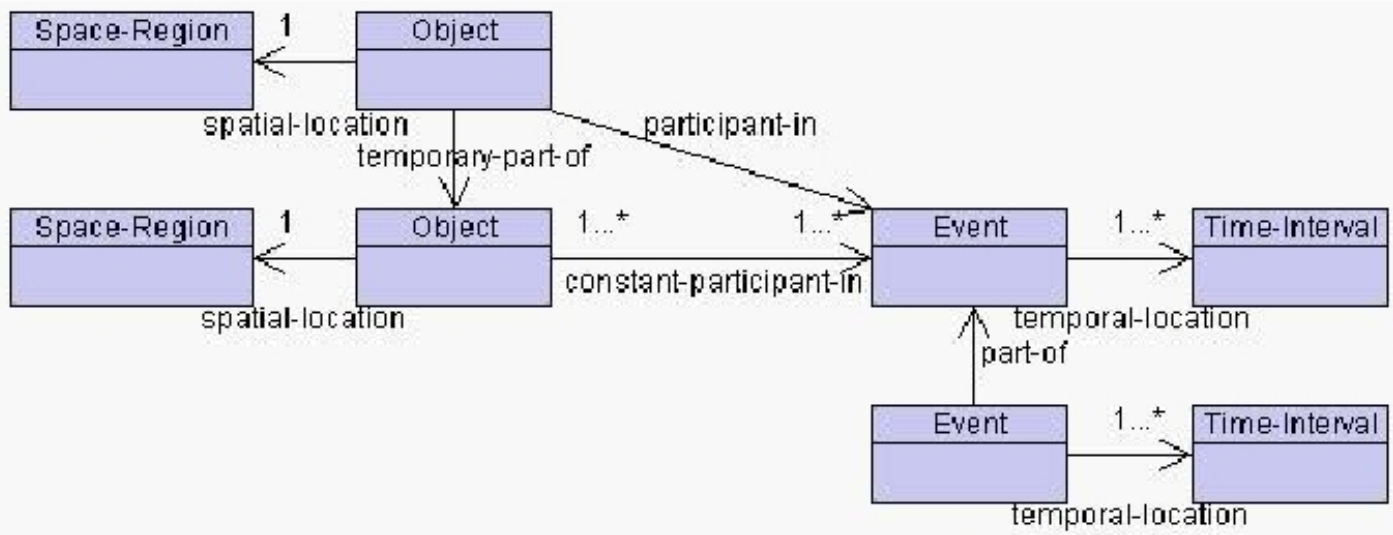
- A formal pattern that encodes a Generic Use Case.
- A template to represent and solve a modelling problem.
- Described by
  - An intuitive set of features
  - A minimal semantic characterization and its formal encoding

Should

- Have a compact description.
- Be composable: an element in a design pattern can be expaned with the help of another design pattern.
- Follow best practices in ontology design.

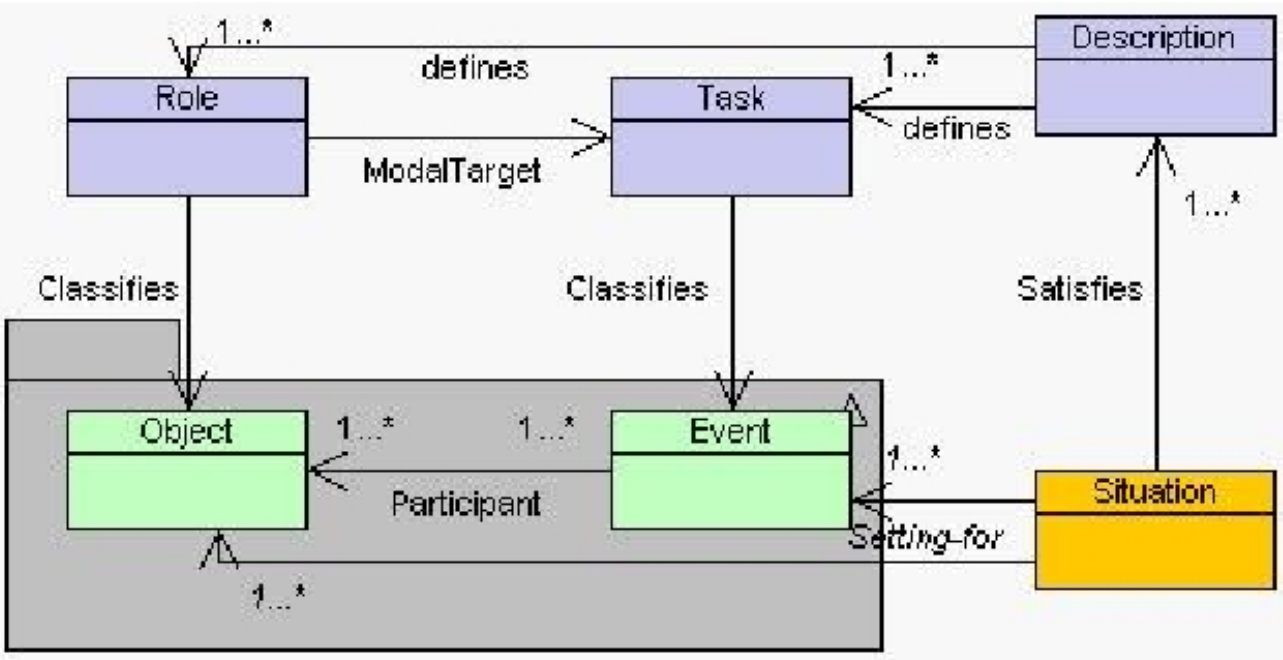
Conceptual Ontology Design Patterns - Example1

Participation at Spacio-Temporal Location pattern



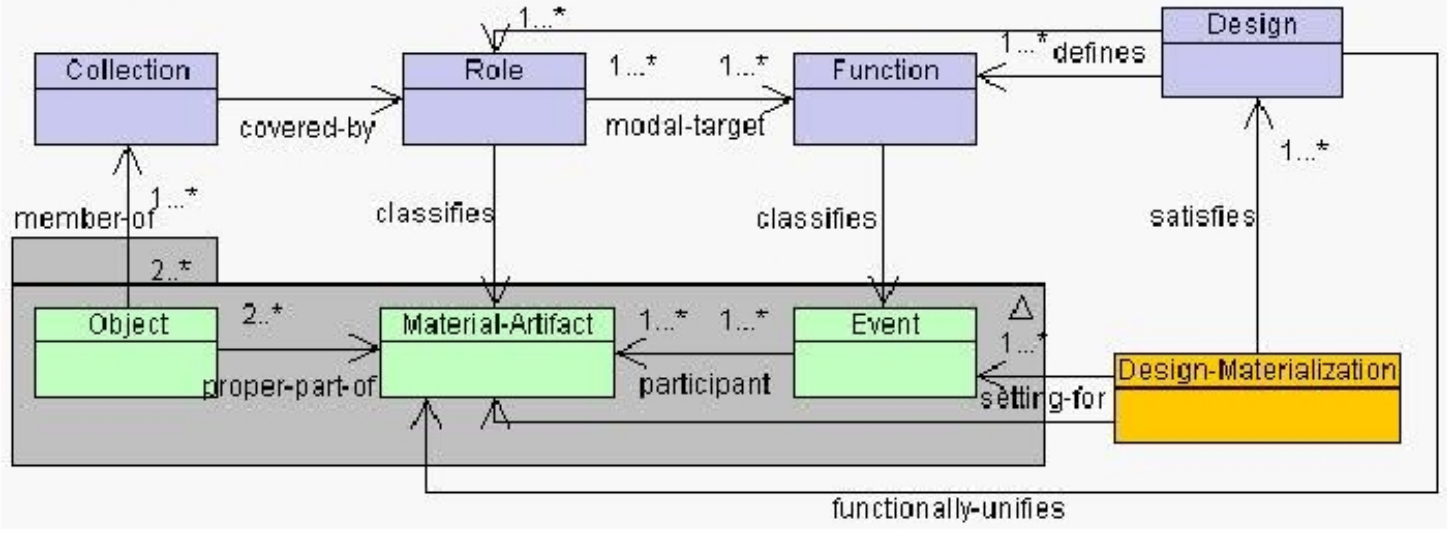
Conceptual Ontology Design Patterns - Example2

Role↔Task pattern



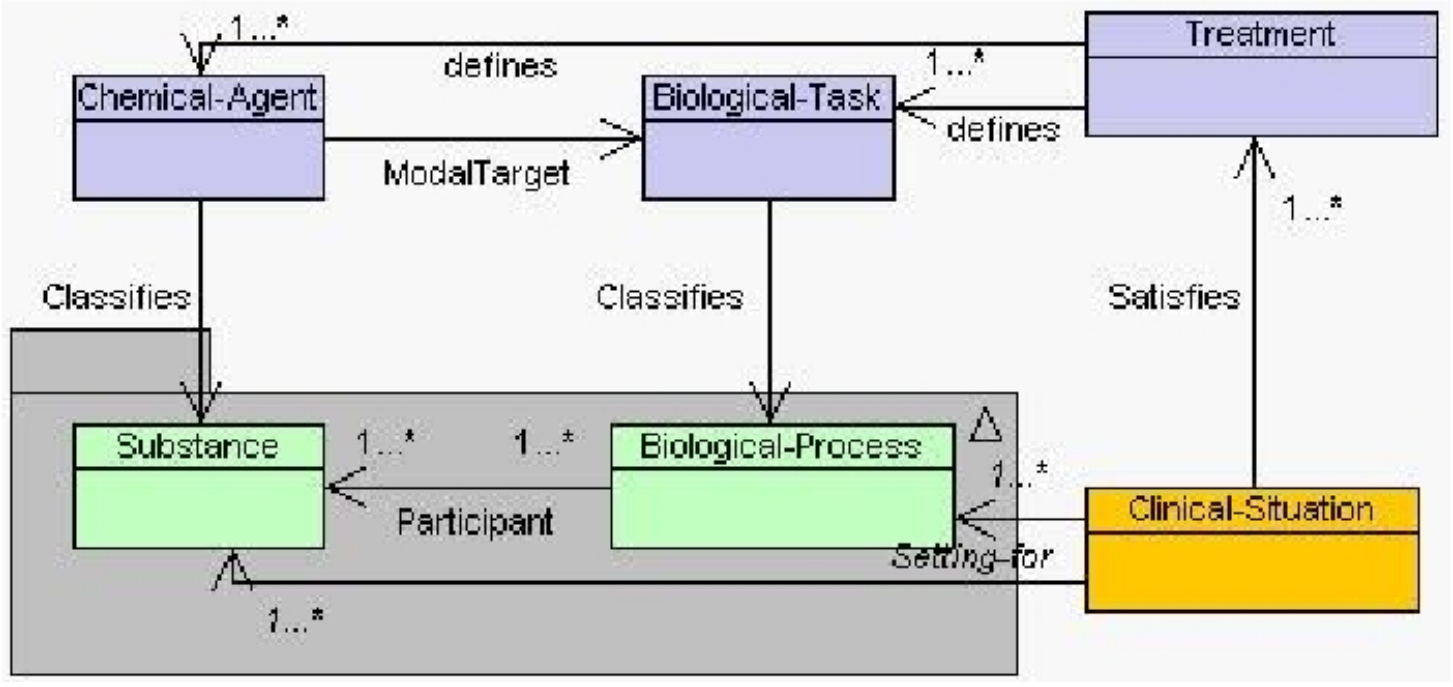
Conceptual Ontology Design Patterns - Example2

Composition of Role↔Task and Collection↔Role patterns



Conceptual Ontology Design Patterns - Example2

Specialization of Role↔Task for chemotherapy.



Types of Ontology Design Patterns

- Structural
- Correspondence
- Reasoning
- Presentation
- Lexico-Syntactic
- Content

Types of Ontology Design Patterns

- Structural:
  - Logical: compositions of logical constructs that solve a problem of expressivity
  - Architectural: composition of Logical Ontology Design Patterns that are used in order to affect the overall shape of the ontology
- Correspondance:
  - Reengineering: transform a conceptual model (possibly a non-ontological resource) into a new ontology
  - Mapping: semantic associations between two existing ontologies

Types of Ontology Design Patterns

- Reasoning: applications of Logical Ontology Design Patterns oriented to obtain certain reasoning results (e.g. classification, subsumption, inheritance, materialization, de-anonymizing,).
- Presentation: usability and readability of ontologies from a user perspective.
- Lexico-Syntactic: linguistic structures or schemas that consist of certain types of words following a specific order, and that permit to generalize and extract some conclusions about the meaning they express.
- Content: encode conceptual, rather than logical design patterns. They solve design problems for the domain classes and properties that populate an ontology.

How Usefull are Patterns?

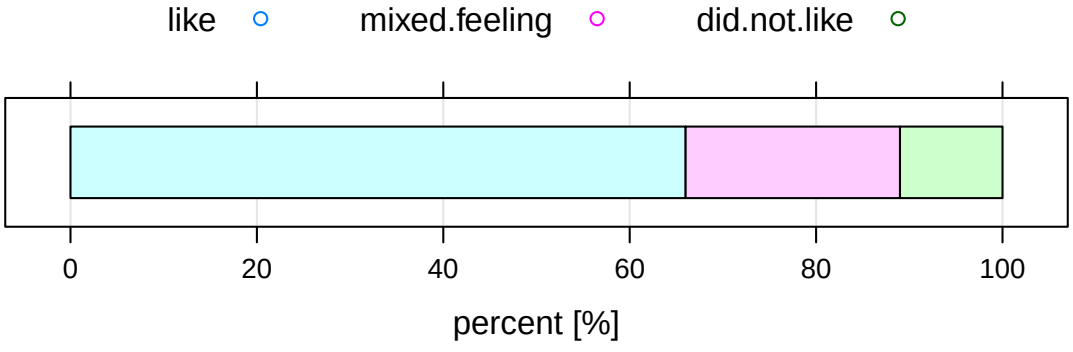
- Are Content Ontology Design Patterns usefull?
- Are ontologies constructed with Content Ontology Design Patterns 'better'?
- Are ontology tasks solved faster with Ontology Design Patterns?

How Usefull are Patterns?

Testing done considering:

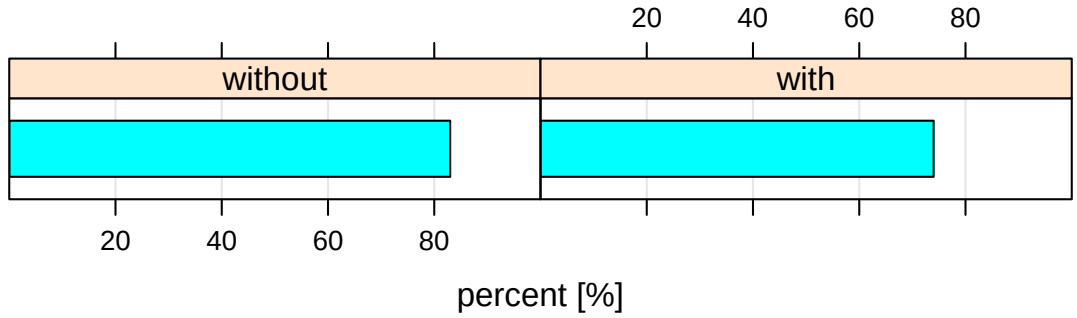
- Cognitive aspects: coverage and understandability of the reuse model.
- Engineering aspects: proof of concept, utility according to some metric

Are Content Ontology Design Patterns Usefull?

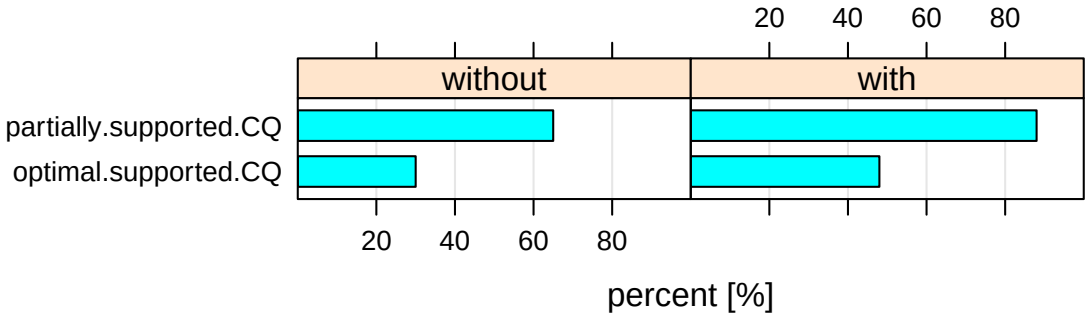


Are Ontologies Constructed with Content Ontology Design Patterns 'Better'?

Terminological Coverage



Functional Coverage



Are Ontology Tasks Solved Faster with Ontology Design Patterns?

Perhaps, if the designer are used to them.

References

1. Introduction to the idea:

   Gangemi, A (2005). **Ontology Design Patterns for Semantic Web Content.** The Semantic Web - ISWC 2005: 4th International Semantic Web Conference.

2. What types of patterns are there?

   Pressuti et al.(2008). **A Library of Ontology Design Patterns: Reusable Solutions for Collaborative Design of Networked Ontologies.** NeOn Deliverable D2.5.1, NeOn Project.

3. What are the effects of using (content) patterns?

   Blomqvist, E; Gangemi, A; Presutti, V (2009). **Experiments on Pattern-Based Ontology Design.** K-CAP '09 - Proceedings of the Fifth International Conference on Knowledge capture.

Questions?