

Formal Models of Design Patterns II

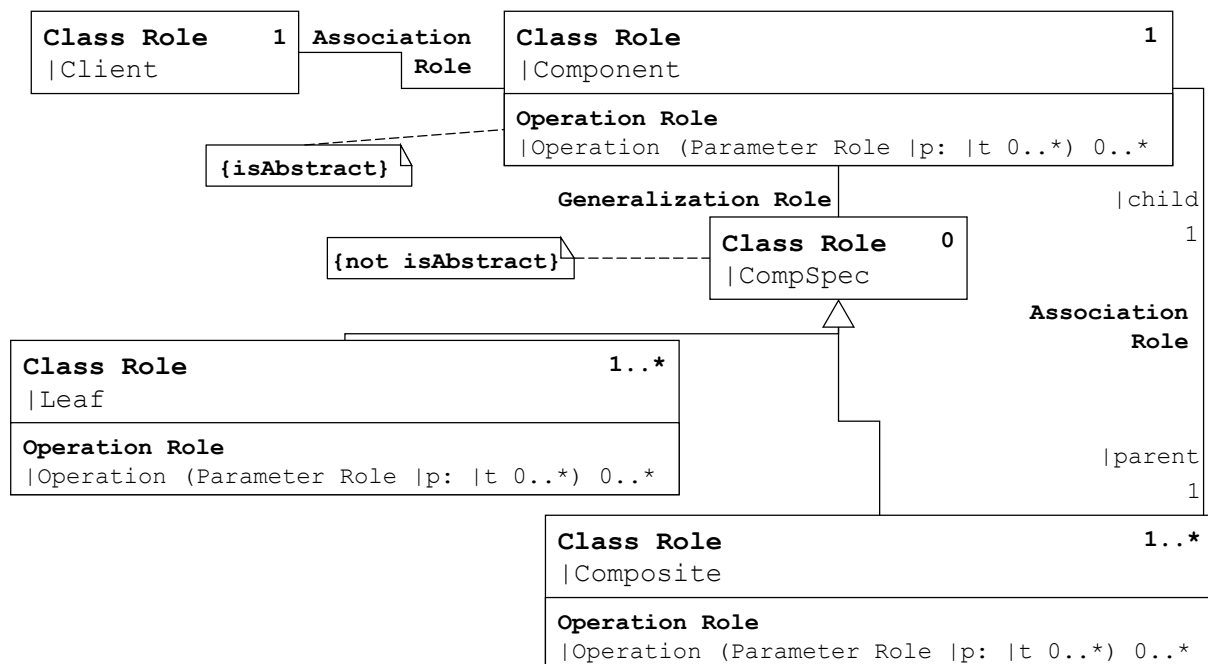
Task 9.1: Composite in RBML

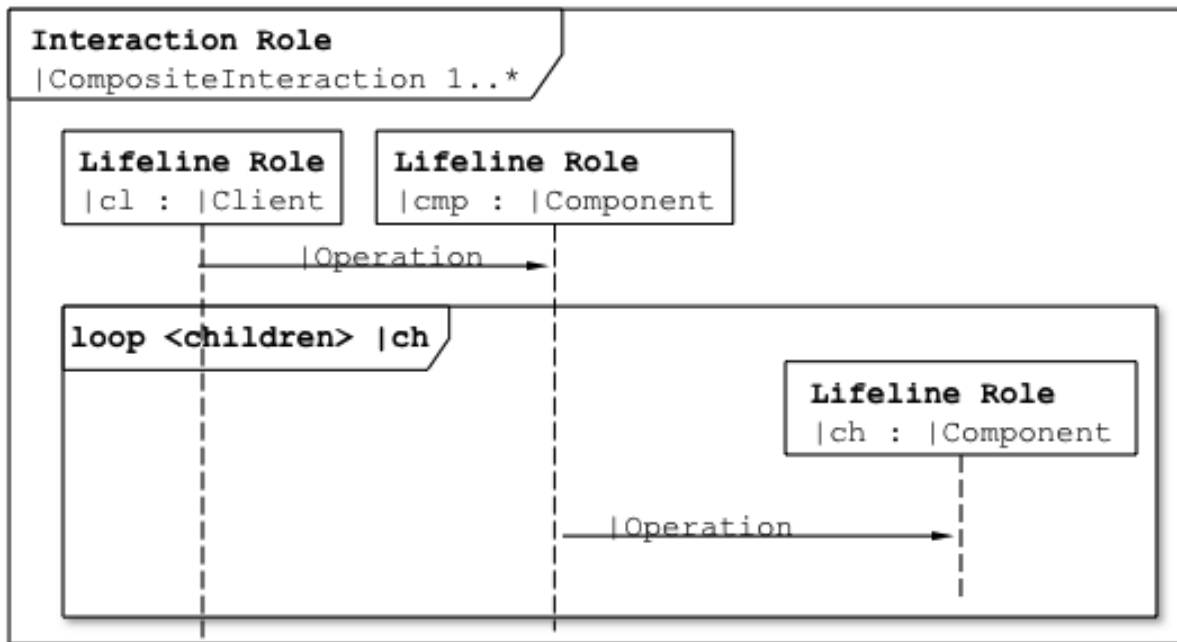
Read and understand [1]. This paper presents RBML, a UML-like notation for representing design patterns. The formal backing of this notation (representing the design patterns' role models as extension of the meta-model) allows for formal treatment of design patterns in actual models.

1a) Task:

Use RBML to represent the COMPOSITE design pattern.

Solution:





1b) Task:

Go back to the task sheet on extensibility patterns and look at your solution for task 1a). Use the RBML techniques presented in [1] to show that this is indeed a realisation of COMPOSITE.

Solution: *Unfortunately, solution hint is not available.*

Bibliography

1. Robert France, Dae-Kyoo Kim, Sudipto Ghosh, Eungee Song, *A UML-Based Pattern Specification Technique*. IEEE Transactions on Software Engineering, Vol 30, number 3, pp 193-206, March 2004. *This paper is available online at the IEEE digital library by visiting <http://ieeexplore.ieee.org/Xplore/DynWel.jsp> and searching for it by title. You should have access to the digital library from any computer in the domain of the Computer Science Department.*

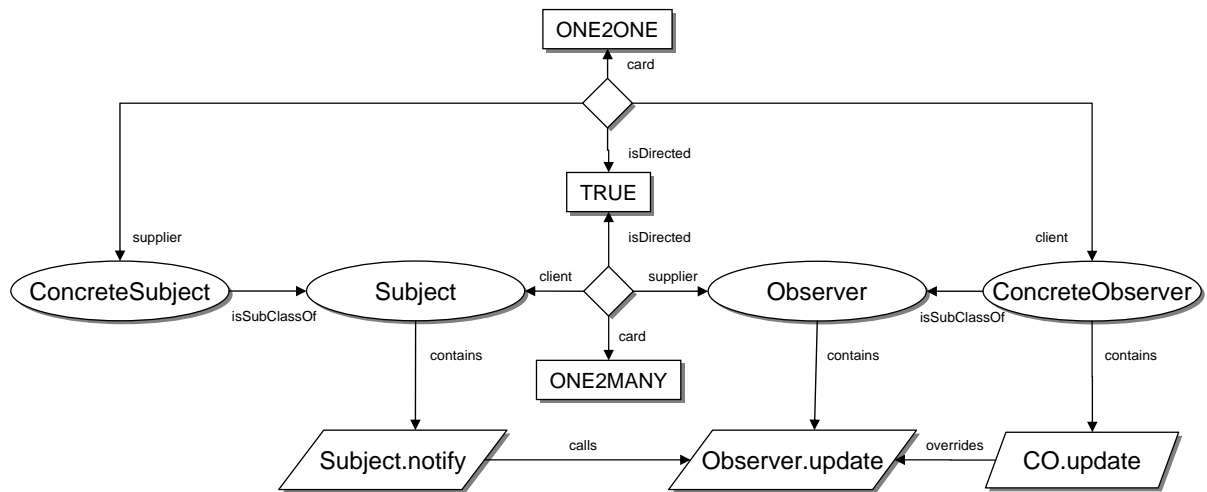
Task 9.2: OWL Observant

Read and understand [1]. This presents an approach that uses Semantic Web technology (in particular ontologies) to model design patterns. An ontology can be viewed (grossly simplifying) as a special kind of class diagram modelling concepts and their relations. An ontology, thus, provides vocabulary allowing to talk about a specific domain.

2a) Task:

Use the technology from [1] to model the OBSERVER design pattern.

Solution: The following is a graphical representation of the resulting ontology. To simplify matters we have left out all properties relating to concepts from ODOL (as specified in *wop.rdf*). Instead, we have used ellipses to denote class templates, diamonds to denote association templates, parallelograms for method templates and rectangles for data-type values.



Bibliography

1. Jens Dietrich and Chris Elgar. *A Formal Description of Design Patterns Using OWL*. In Proc. 2005 Australian Software Engineering Conference (ASWEC'05), IEEE Press, 2005.

Task 9.3: Discussion: Formal Representation of Design Patterns

From your experience with the pattern formalisations looked at so far, what are the benefits and drawbacks of attempts at formalising design patterns?

Solution: The main points to be discussed here are: Ambiguity, Relations between Patterns, Automation and Tool Support, Difficulty, Lack of Variation in formally specified patterns, ...