elements that define relationships. Figure 2-7 on page 2-15 shows the model elements that define dependencies. Figure 2-8 on page 2-16 shows the various kinds of classifiers. Figure 2-9 on page 2-17 shows auxiliary elements for template parameters, presentation elements, and comments.

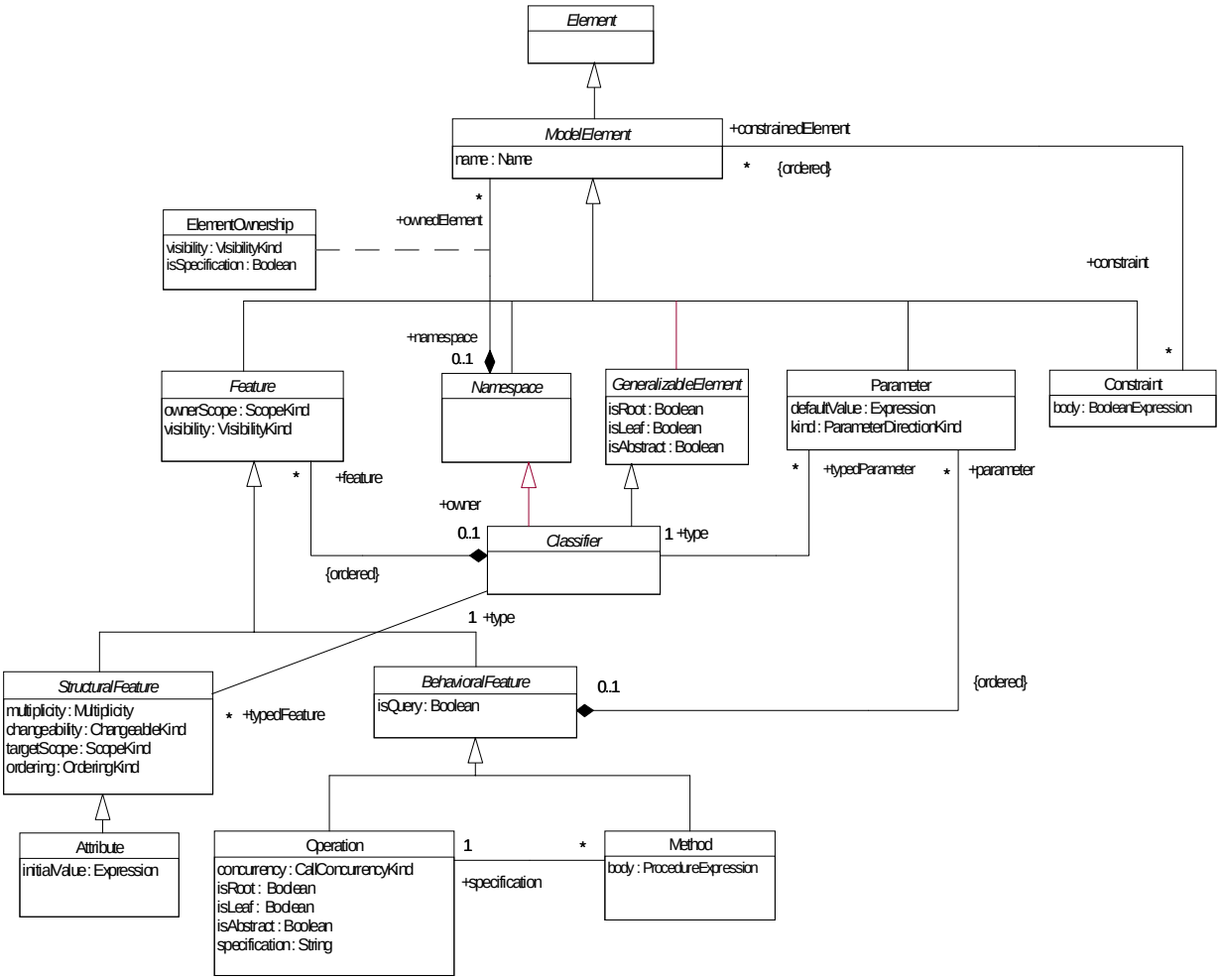

Figure 2-5 Core Package - Backbone

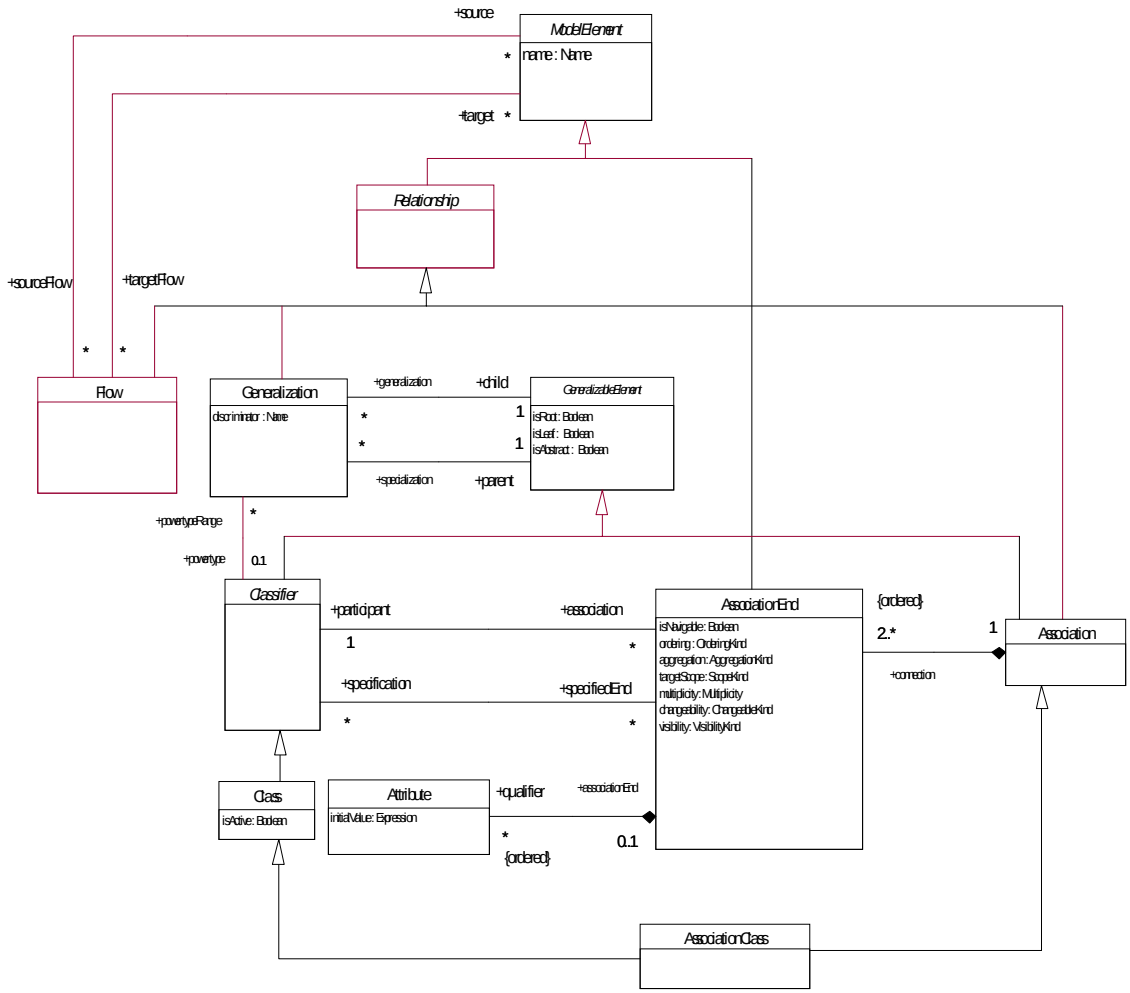

Figure 2-6 Core Package - Relationships

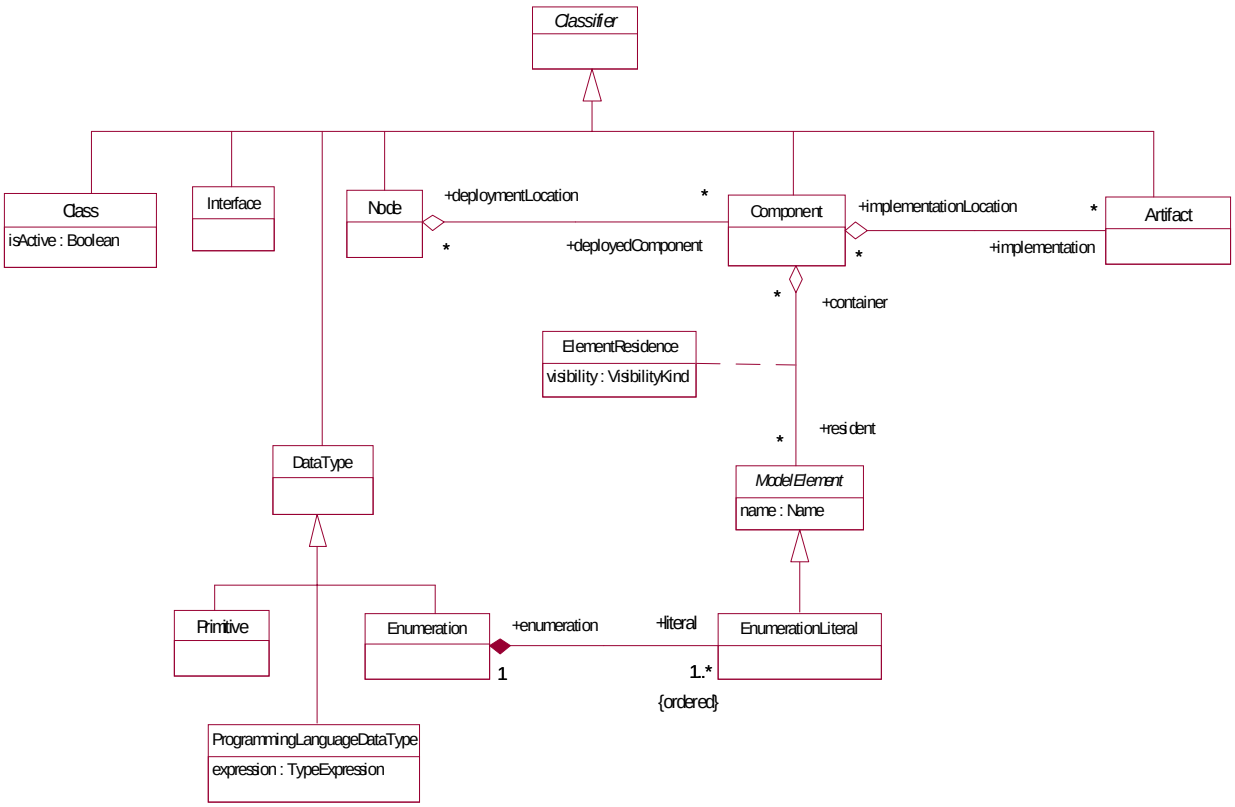

Figure 2-8 Core Package - Classifiers

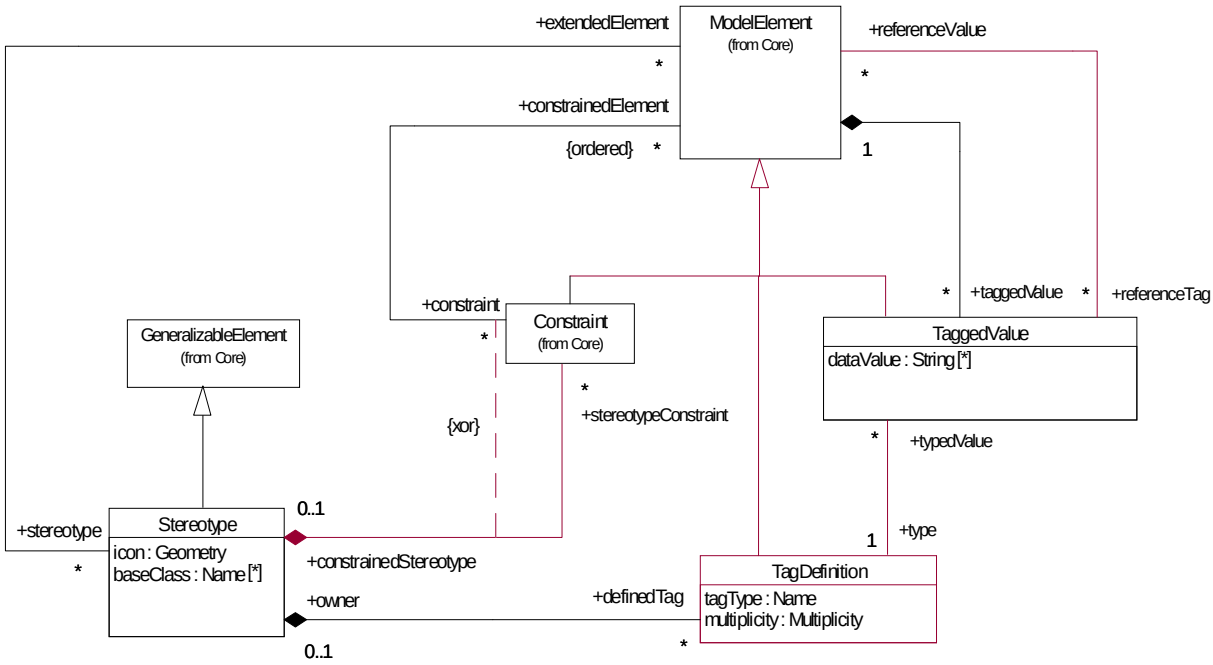

Figure 2-10 Extension Mechanisms

2.6.2.1 Constraint (as extended)

The constraint concept allows new semantics to be specified linguistically for a model element. The specification is written as an expression in a designated constraint language. The language can be specially designed for writing constraints (such as OCL), a programming language, mathematical notation, or natural language. If constraints are to be enforced by a model editor tool, then the tool must understand the syntax and semantics of the constraint language. Because the choice of language is arbitrary, constraints are an extension mechanism.

In the metamodel, a constraint directly attached to a model element describes semantic restrictions that this model element must obey. Constraints attached to a Stereotype apply to each model element that bears that stereotype. Note that, for the case of constraints attached to stereotype definitions, the scope of the constraint is the UML metamodel and not the model in which it is defined. This allows the definition of well-formedness rules for stereotypes in the same manner as the well-formedness rules of other metamodel elements.