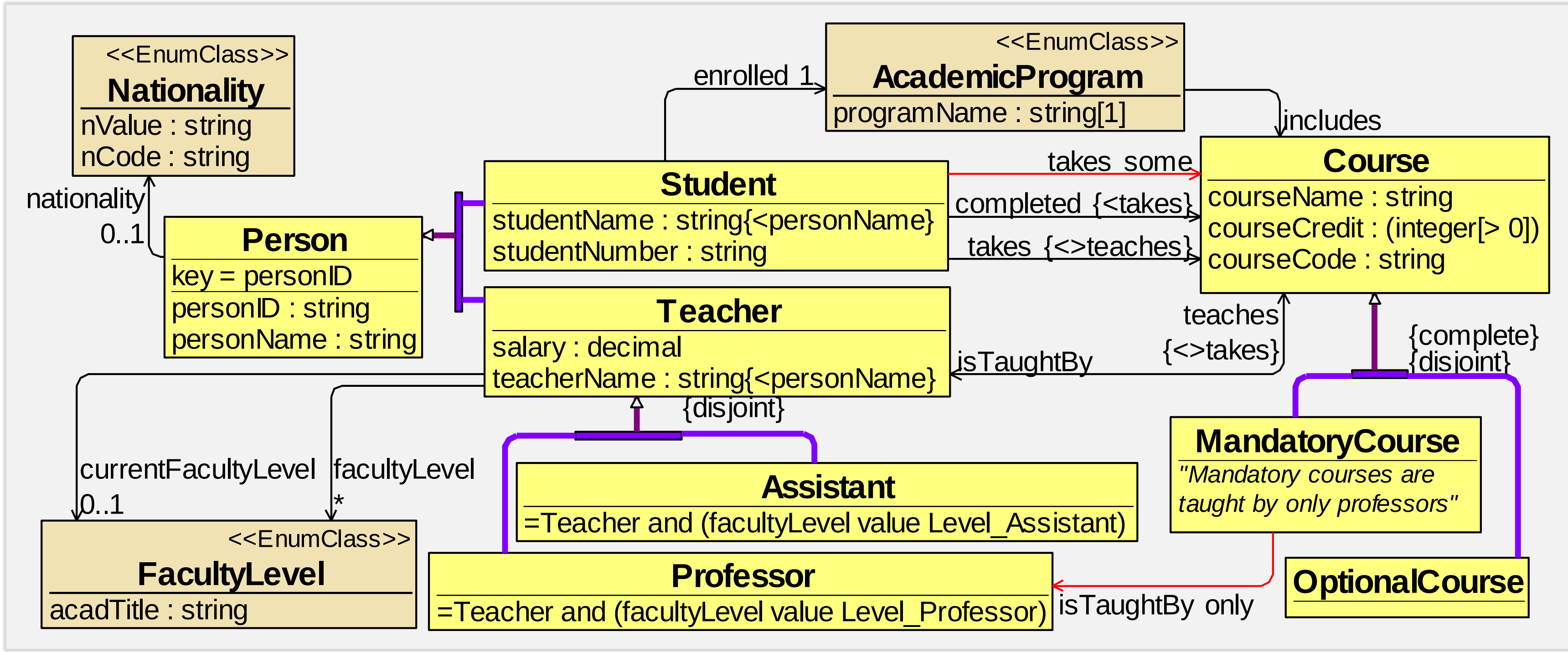


Adding Context in OWL Ontologies: Visual Qualifiers in OWLGrEd

Jūlija Ovčiņņikova, Kārlis Čerāns

Institute of Mathematics and Computer Science, University of Latvia



Ontology editor **OWLGrEd** provides a UML-style graphical notation for OWL 2.

- Classes: represented as UML-style class diagrams.
- Object Properties: shown as associations.
- Data Properties: displayed as attributes within class boxes.
- Individuals: visualized as class instances.
- Logical axioms (Subclass, Equivalence, Disjointness, Restrictions): represented either textually or via visual connectors.
- Visual fields for expressing property characteristics.
- Anonymous Classes: visually represent complex class expressions or restrictions.

Visual Notation Extensions

To introduce property qualifiers into the OWLGrEd visual notation, we:

- provide the option to visually specify object and data properties using dedicated graphical boxes,
- enable describing qualifiers within compartments of these property boxes.

Qualifier annotations

We use the OWL 2 annotation mechanism to introduce property qualifiers such as temporal scopes, provenance information, and confidence levels. These annotations enrich both:

- property declarations (schema-level relationships) and
- property assertions (instance-level facts)

Qualifier property characteristics introduced both locally and globally.

```
AnnotationAssertion(  
  Annotation(rdfs:range xsd:dateTime)  
  Annotation(owl:maxCardinality  
    "1"^^xsd:nonNegativeInteger)  
  ex:qualifier :salary :dateTo)
```

```
AnnotationAssertion(ex:qualifier :salary :dateTo)  
AnnotationPropertyRange(:dateTo xsd:dateTime)  
AnnotationAssertion(owl:maxCardinality :dateTo 1)
```

The prototype extension of the OWLGrEd editor to support visual annotation of OWL axiom is available at:

https://owlgred.lumii.lv/downloads/OWLGrEd_extended.zip

The prototype includes a sample mini-University project using the extra annotation notation.

	<pre>Declaration(Class(:Student)) Declaration(Class(:Course)) Declaration(AnnotationProperty(:datePaid)) Declaration(AnnotationProperty(:tutor)) Declaration(AnnotationProperty(:dateCompleted)) Declaration(AnnotationProperty(:mark)) Declaration(ObjectProperty(:takes)) AnnotationAssertion(Annotation(rdfs:range xsd:dateTime) ex:qualifier :takes :datePaid) AnnotationAssertion(Annotation(rdfs:range :Teacher) ex:qualifier :takes :tutor) Declaration(ObjectProperty(:completed)) AnnotationAssertion(Annotation(rdfs:range xsd:dateTime) ex:qualifier :completed :dateCompleted) AnnotationAssertion(Annotation(rdfs:range xsd:integer) ex:qualifier :completed :mark) ObjectPropertyDomain(:completed :Student) ObjectPropertyRange(:completed :Course) SubObjectPropertyOf(:completed :takes)</pre>
	<pre>Declaration(NamedIndividual(:Alice)) Declaration(NamedIndividual(:ComputerScience)) ClassAssertion(:Student :Alice) ClassAssertion(:MandatoryCourse :ComputerScience) ObjectPropertyAssertion(Annotation(:datePaid "2025-01-01"^^xsd:dateTime) Annotation(:tutor :John) :takes :Alice :ComputerScience)</pre>
	<pre>Declaration(Class(:Teacher)) Declaration(AnnotationProperty(:dateFrom)) Declaration(AnnotationProperty(:dateTo)) Declaration(DataProperty(:salary)) AnnotationAssertion(Annotation(rdfs:range xsd:dateTime) ex:qualifier :salary :dateFrom) AnnotationAssertion(Annotation(rdfs:range xsd:dateTime) Annotation(owl:maxCardinality "1"^^xsd:nonNegativeInteger) ex:qualifier :salary :dateTo) DataPropertyDomain(:salary :Teacher) DataPropertyRange(:salary xsd:decimal)</pre>

OWLGrEd editor:

