GRAPHWISE
AI THRIVES ON WHOLE DATA

June 4st, 2025

Graphwise
Vienna University of Economics and Business

OWL strict

A Constrained OWL Fragment for Knowledge Graph Practicioners

Robert David, Albin Ahmeti, Shqiponja Ahmetaj, Axel Polleres

Motivation

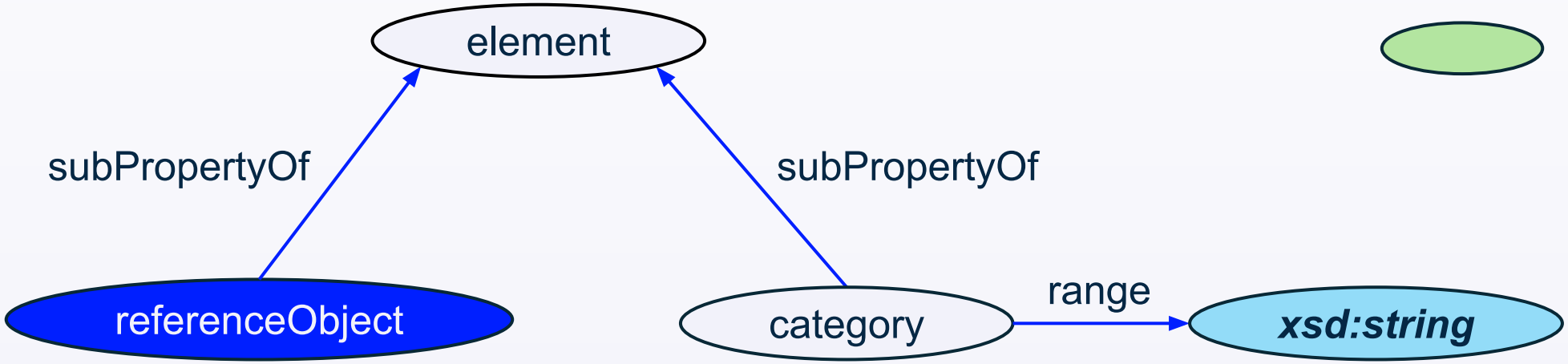
GRAPHWISE
AI THRIVES ON WHOLE DATA

- RDF is a very flexible, does not restrict use
- OWL ontologies are represented using RDF
- **BUT:** Flexibility causes ambigUIties!
- Our Scenario: Enterprise Knowledge Graphs - what matters?
  - Quality & stability of schemas
  - Editors to manage ontologies
  - Applications using (instance) data

Ambiguity Example

GRAPHWISE
AI THRIVES ON WHOLE DATA

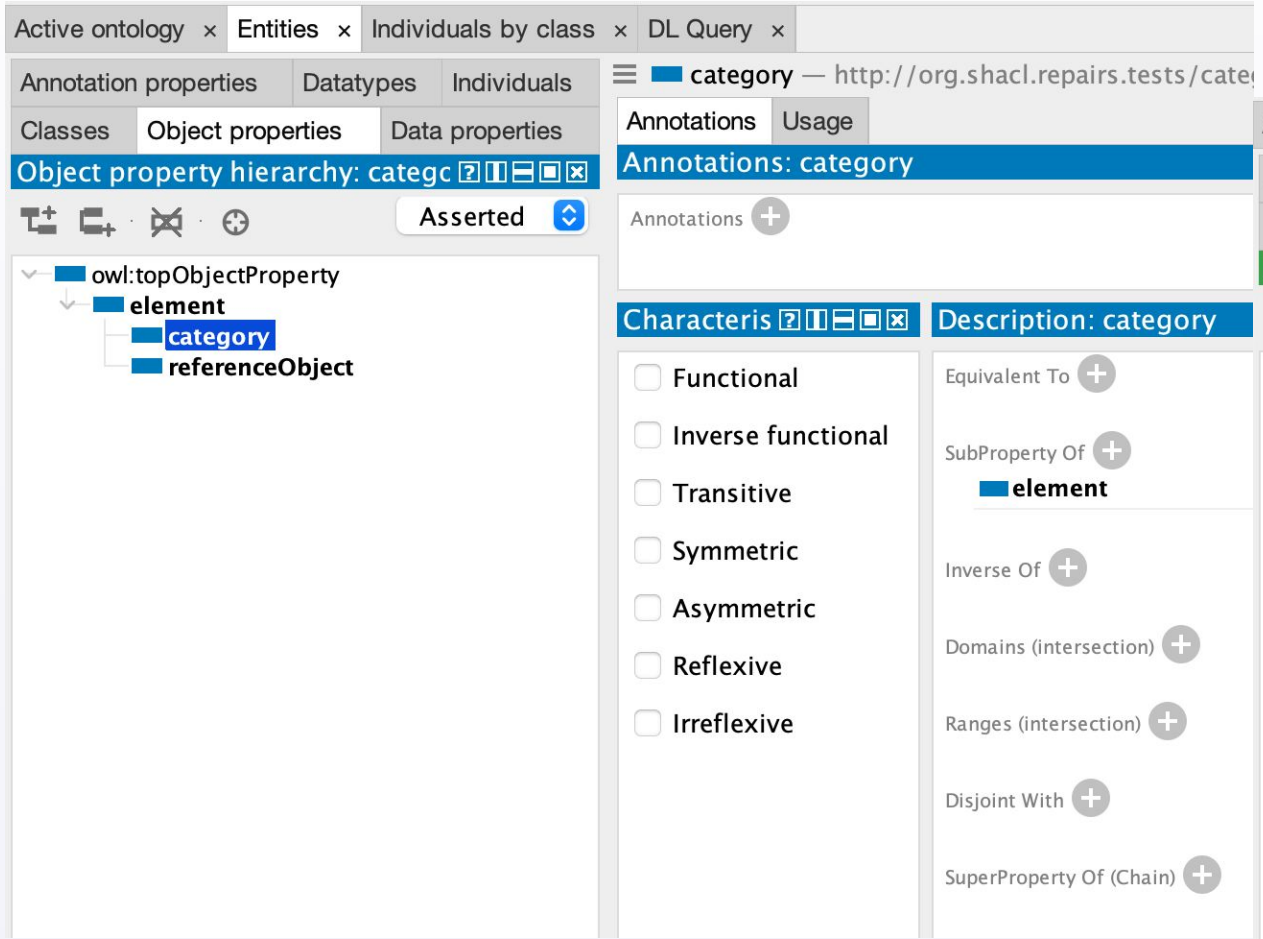
Part of the DITA ontology

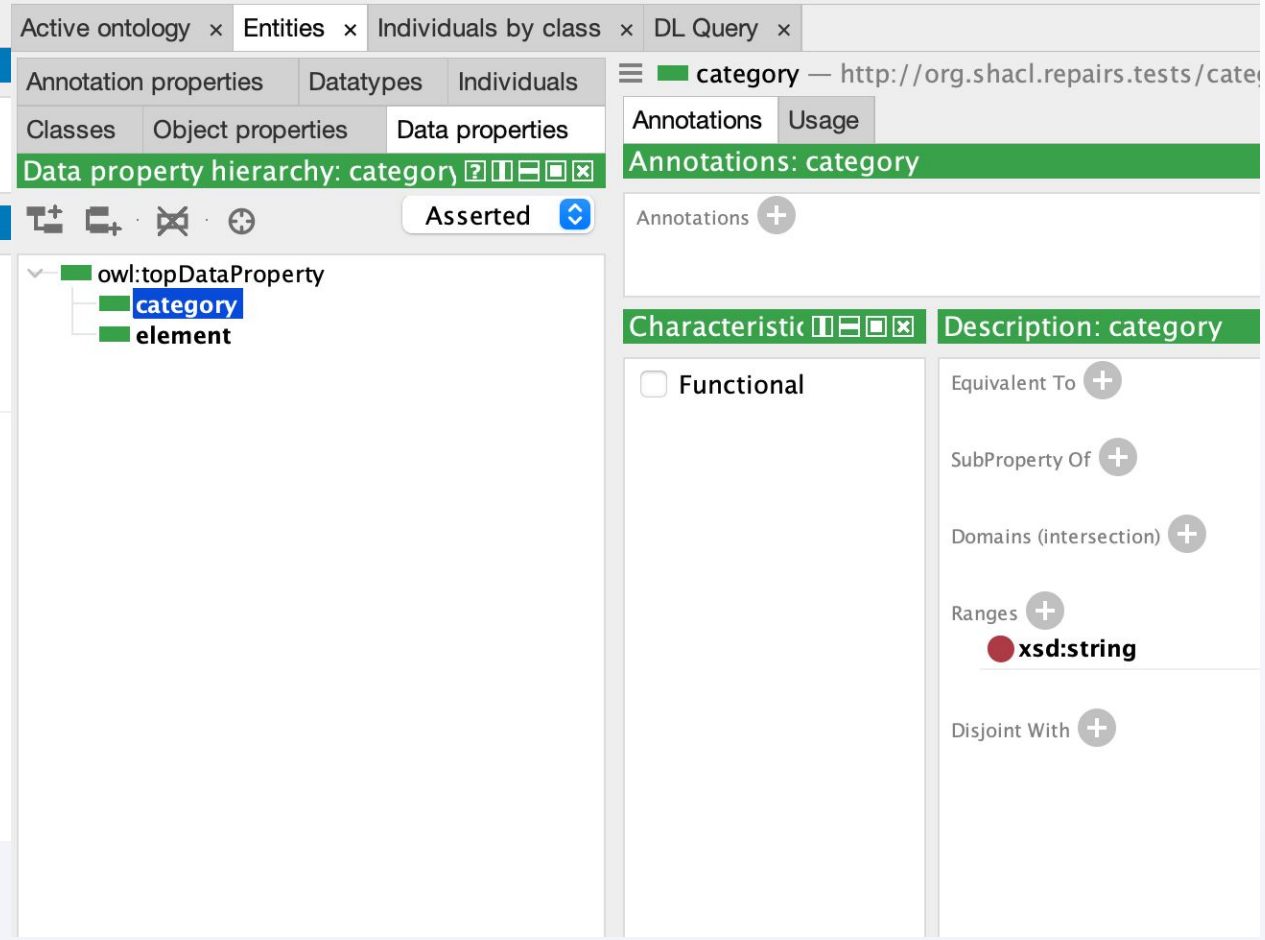


DatatypeProperty
ObjectProperty
rdf:Property
Datatype
Class

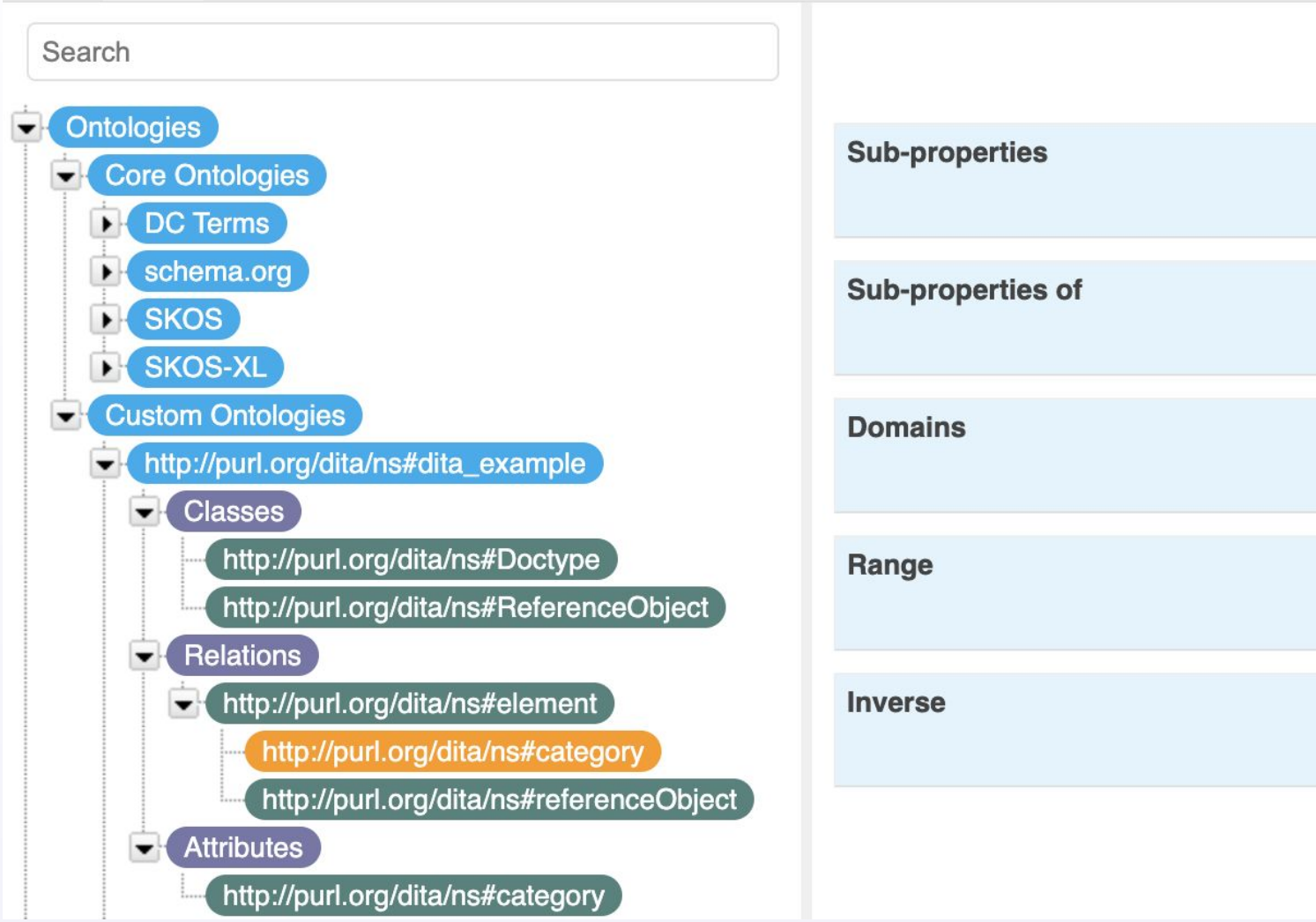
AmbigUIty Example – Protégé

GRAPHWISE
AI THRIVES ON WHOLE DATA



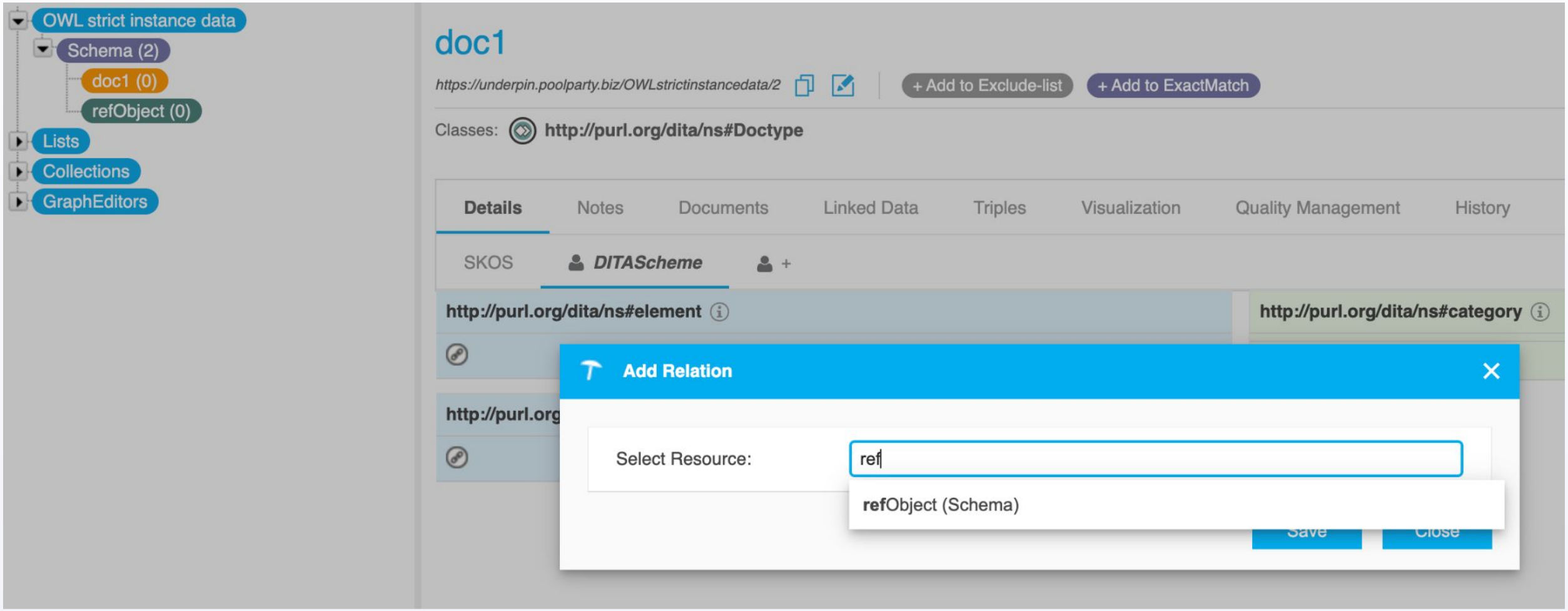


AmbigUIty Example – PoolParty



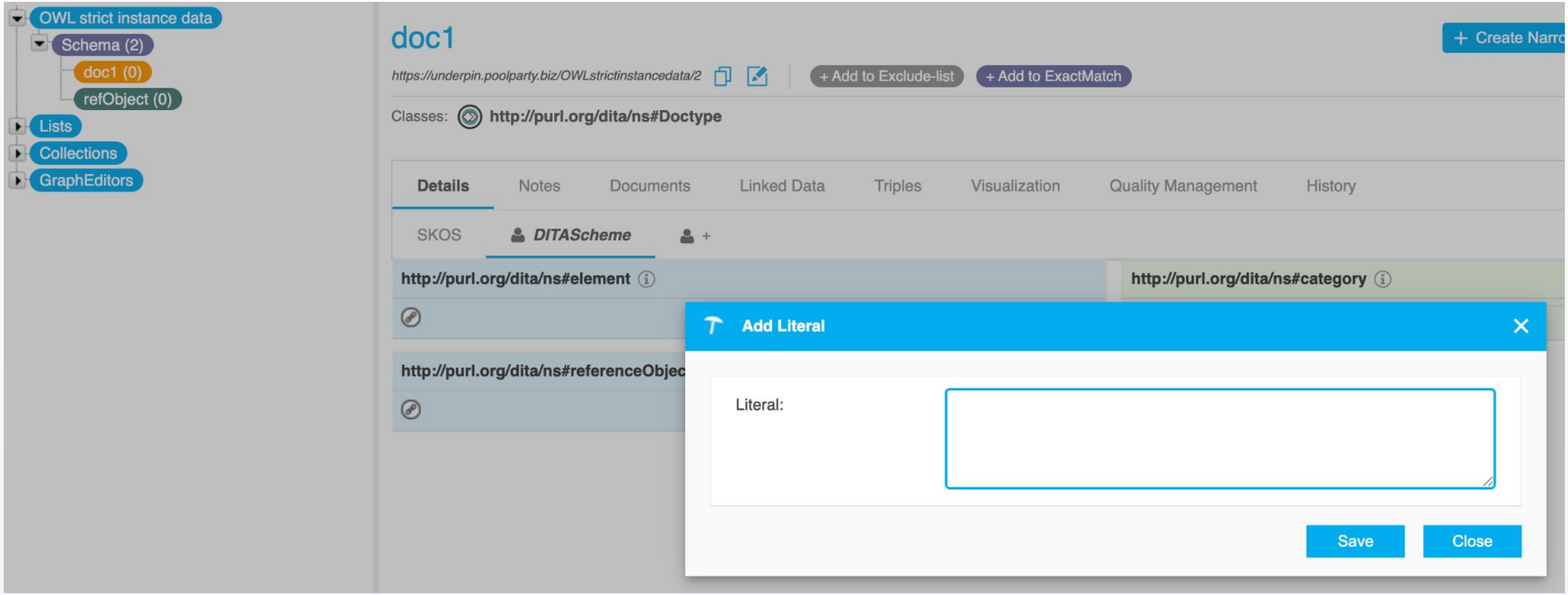
Concrete Range Type – Relation

GRAPHWISE
AI THRIVES ON WHOLE DATA



Concrete Range Type – Datatype

GRAPHWISE
AI THRIVES ON WHOLE DATA



OWL strict

GRAPHWISE
AI THRIVES ON WHOLE DATA

- Goal: to resolve ambiguities
  - by rules and constraints…
  - … enforcing ***syntactic restrictions*** on the RDF graph

  → “well-defined/behaving fragment”

- i.e., syntax constraints only (RDF graph); no OWL reasoner needed

How about existing OWL fragments?

GRAPHWISE
AI THRIVES ON WHOLE DATA

Name	W3C Rec.	Purpose
RDF 1999	yes	first RDF recommendation version
RDF 1.0	yes	February 2004
RDF 1.1	yes	June 2014
RDFS 1.0	yes	schema description
RDFS 1.1	yes	schema description
Minimal RDF(S) [26]	no	small and practical
OWL 1 Lite	yes	minimum complexity
OWL 1 DL	yes	Description logics
OWL 1 Full	yes	maximum expressivity
OWL 2 EL	yes	many classes & properties
OWL 2 QL	yes	query rewriting
OWL 2 RL	yes	rules & reasoning
RDFS++	no	AllegroGraph's minimal OWL profile
OWL Horst [30]	no	scalable reasoning
OWLSIF	no	Oracle's OWL with if-semantics
OWLPrime	no	Oracle's application-oriented OWL profile
OWL DBP [3]	no	DBPedia schema
OWL Wikidata	no	Wikidata schema
OWL LD [15]	no	Linked data practical use
OWL IC [29]	no	integrity constraints
OWL Flight [11]	no	logic programming & constraints

- None of those "does the job"... Reasoning ***does not fix*** ambigUIties

OWL strict has an orthogonal goal:

GRAPHWISE
AI THRIVES ON WHOLE DATA

Restricting the ***vocabulary use*** *(tailored for editors)*:

C = { *RDFSClass, OWLClass, Datatype, Property,*
ObjectProperty, DatatypeProperty, AnnotationProperty,
OntologyProperty, SymmetricProperty,
FunctionalProperty }

P = { *subClassOf, subPropertyOf, domain, range,*
inverseOf, disjointWith }

Requirements to avoid Ambiguities

GRAPHWISE
AI THRIVES ON WHOLE DATA

- **R-0 “standard use” of the RDF(S) and OWL vocab, i.e.**

  ~~`rdf:type rdfs:subClassOf my:Class.`~~

- **R-1** ontology terms in exactly one of the OWL built-in classes *OWLClass, Datatype, ObjectProperty, DatatypeProperty*
- **R-2** “consistent” *subpropertyOf* hierarchy, i.e., subproperties preserve the same class membership
- **R-3** *domain* and *range* restricted to *one* (union of) class, i.e. choices to satisfy the domain/range
- **R-4** “consistent” *domain* and *range* in *subpropertyOf* hierarchies, i.e. same (sub)classes on super-properties

OWL strict

GRAPHWISE
AI THRIVES ON WHOLE DATA

Restricting the syntax:

- 26 syntactic restrictions on the ontology RDF graph
- Implement the requirements **R-0** to **R-4**
- Examples
  - ***SR-4*** *If subPropertyOf (x, y) ∈ G, then x and y are both members of exactly one class of ObjectProperty or DatatypeProperty.*
  - ***SR-13*** *If range(x, y), DatatypeProperty(x) ∈ G, then Datatype(y) ∈ G.*
  - ***SR-18*** *If subPropertyOf (x, y), ObjectProperty(x) ∈ G, then ObjectProperty(y) ∈ G.*

Expressing OWL strict in SHACL

GRAPHWISE
AI THRIVES ON WHOLE DATA

Set of 10 SHACL shapes to implement syntactic restrictions.

Disjoint class membership constraints (for all ontology terms):

ClassShape XOR DatatypeShape XOR PropertyShape

ProertyShape -> ObjectPropertyShape XOR DatatypePropertyShape

Additional property restrictions:

FunctionalPropertyShape, SymmetricPropertyShape, InversePropertyShape

Consistency in the subproperty hierarchy:

DomainConsistencyShape, RangeConsistencyShape

https://github.com/robert-david/OWL-strict/blob/main/ontology_repair_shapes.ttl

Expressing OWL strict in SHACL

GRAPHWISE
AI THRIVES ON WHOLE DATA

SHACL implementation example:

:ObjectPropertyShape a sh:NodeShape;

sh:class owl:ObjectProperty;	*SR-10*
sh:property [sh:path rdfs:range; sh:node :ClassShape; **sh:maxCount 1**];	*SR-12*
sh:property [sh:path (rdfs:range owl:unionOf [sh:zeroOrMorePath rdf:rest] rdf:first); sh:node :ClassShape;];	*SR-12*
sh:property [sh:path rdfs:subPropertyOf; sh:class owl:ObjectProperty;];	*SR-17*
sh:not [sh:class owl:DatatypeProperty;].	*SR-0*

https://github.com/robert-david/OWL-strict/blob/main/ontology_repair_shapes.ttl

Transforming ontologies to OWL strict

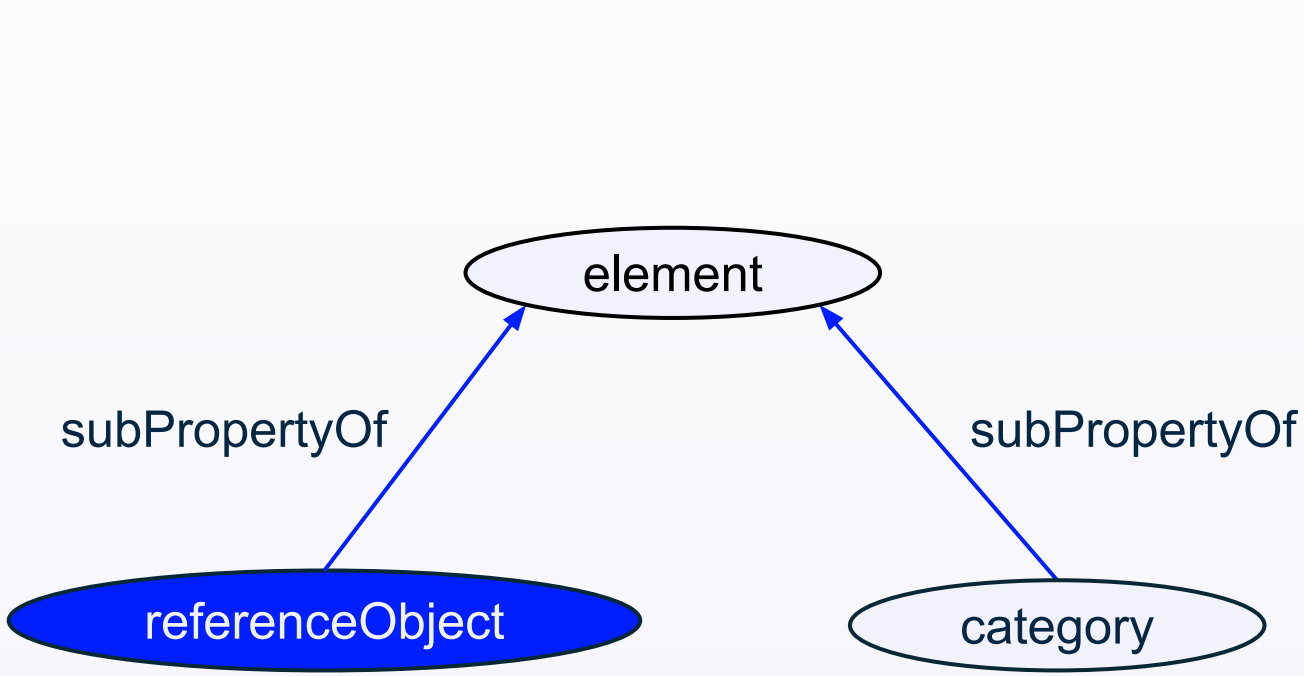
GRAPHWISE
AI THRIVES ON WHOLE DATA

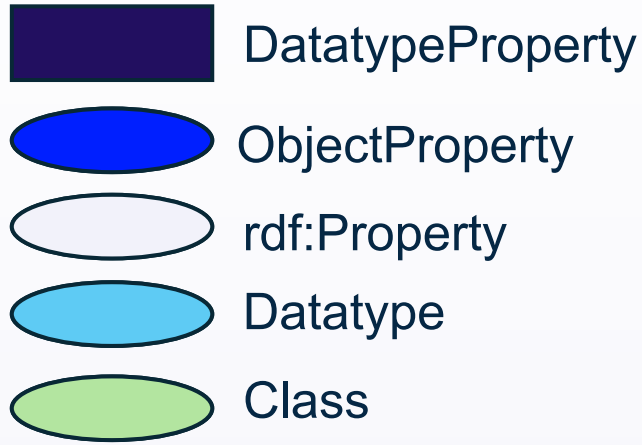
Using ***SHACL repairs*** (see our ISWC 2022 paper):

- Algorithm to modify SHACL data graph ontology RDF to conform to the SHACL constraints:
  - produces ***minimal*** sets of additions and deletions of triples…
  - … guaranteed to conform to OWL strict

Repairing Ambiguities Example 1/3

GRAPHWISE
AI THRIVES ON WHOLE DATA





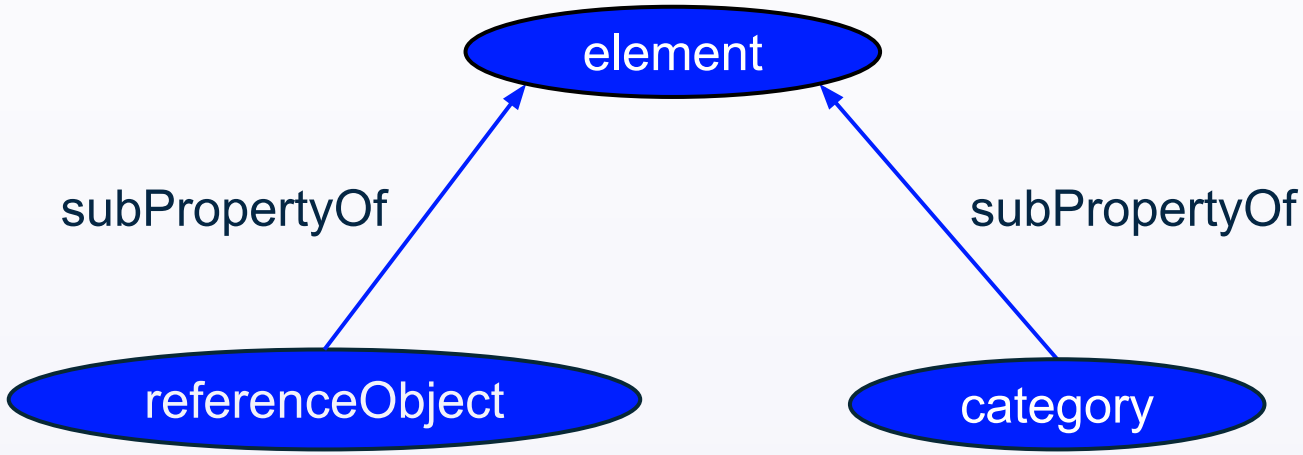
SR-4 If $subPropertyOf(x, y) \in G$, then x and y are both members of exactly one class of ObjectProperty or DatatypeProperty.

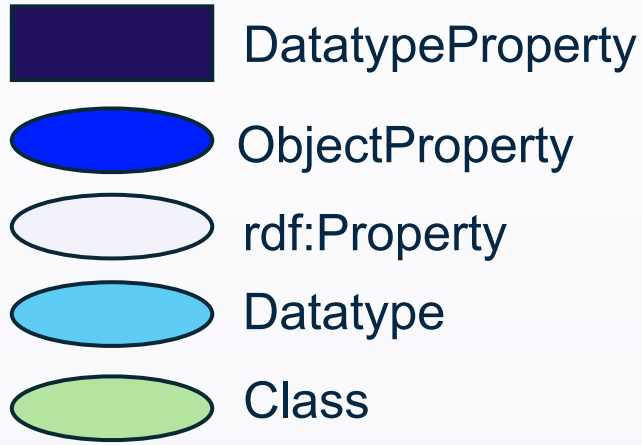
SR-18 If $subPropertyOf(x, y)$, ObjectProperty$(x) \in G$, then ObjectProperty$(y) \in G$.

SR-19 If $subPropertyOf(x, y)$, DatatypeProperty$(x) \in G$, then DatatypeProperty$(y) \in G$.

Repairing Ambiguities Example 1/3

GRAPHWISE
AI THRIVES ON WHOLE DATA





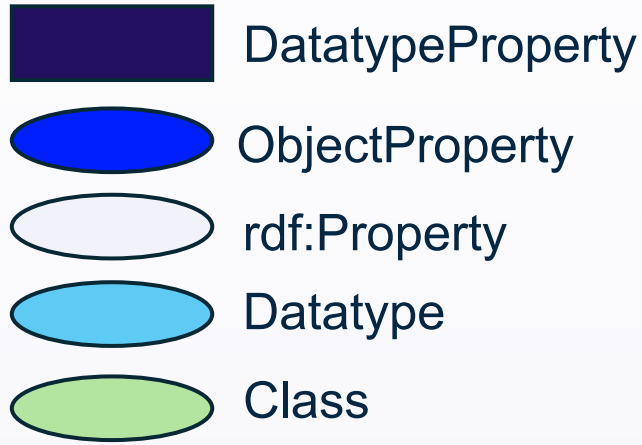
SR-4 If $subPropertyOf(x, y) \in G$, then x and y are both members of exactly one class of ObjectProperty or DatatypeProperty.

SR-18 If $subPropertyOf(x, y)$, ObjectProperty$(x) \in G$, then ObjectProperty$(y) \in G$.

SR-19 If $subPropertyOf(x, y)$, DatatypeProperty$(x) \in G$, then DatatypeProperty$(y) \in G$.

Repairing Ambiguities Example 2/3

GRAPHWISE
AI THRIVES ON WHOLE DATA



SR-6 If $range(x, y) \in G$, then x is a member of exactly one class of ObjectProperty or DatatypeProperty.

SR-12 If $range(x, y)$, $\mathsf{ObjectProperty}(x) \in G$, then $\mathsf{OWLClass}(y) \in G$.

SR-13 If $range(x, y)$, $\mathsf{DatatypeProperty}(x) \in G$, then $\mathsf{Datatype}(y) \in G$.

Repairing Ambiguities Example 2/3

GRAPHWISE
AI THRIVES ON WHOLE DATA

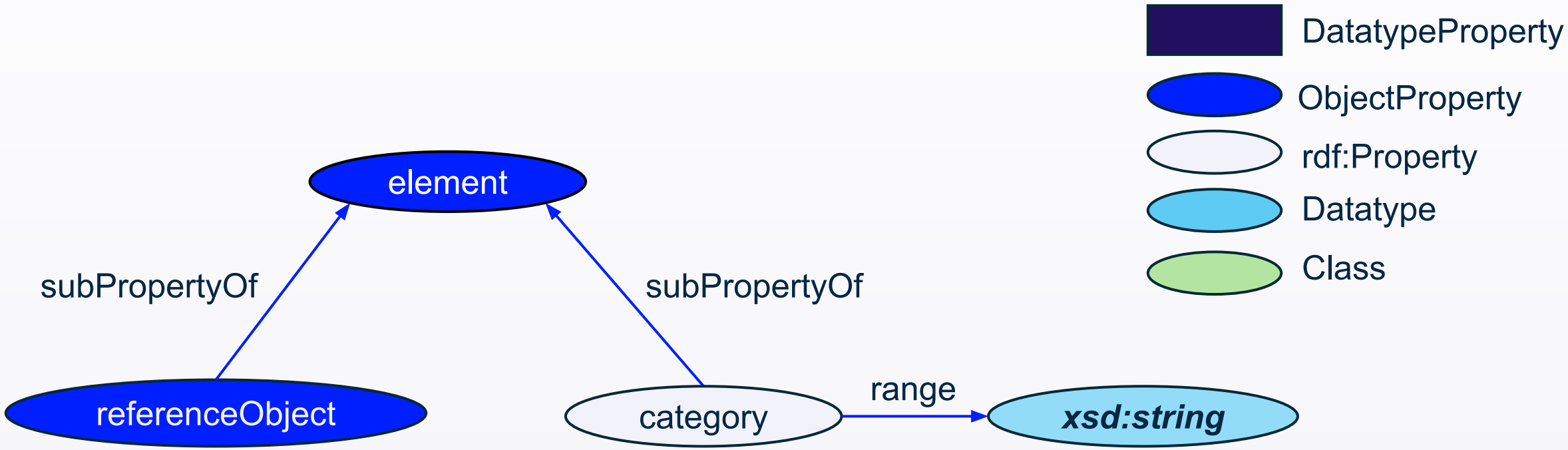
SR-6 If $range(x, y) \in G$, then x is a member of exactly one class of ObjectProperty or DatatypeProperty.

SR-12 If $range(x, y)$, $\mathsf{ObjectProperty}(x) \in G$, then $\mathsf{OWLClass}(y) \in G$.

SR-13 If $range(x, y)$, $\mathsf{DatatypeProperty}(x) \in G$, then $\mathsf{Datatype}(y) \in G$.

Repairing Ambiguities Example 3/3

GRAPHWISE
AI THRIVES ON WHOLE DATA



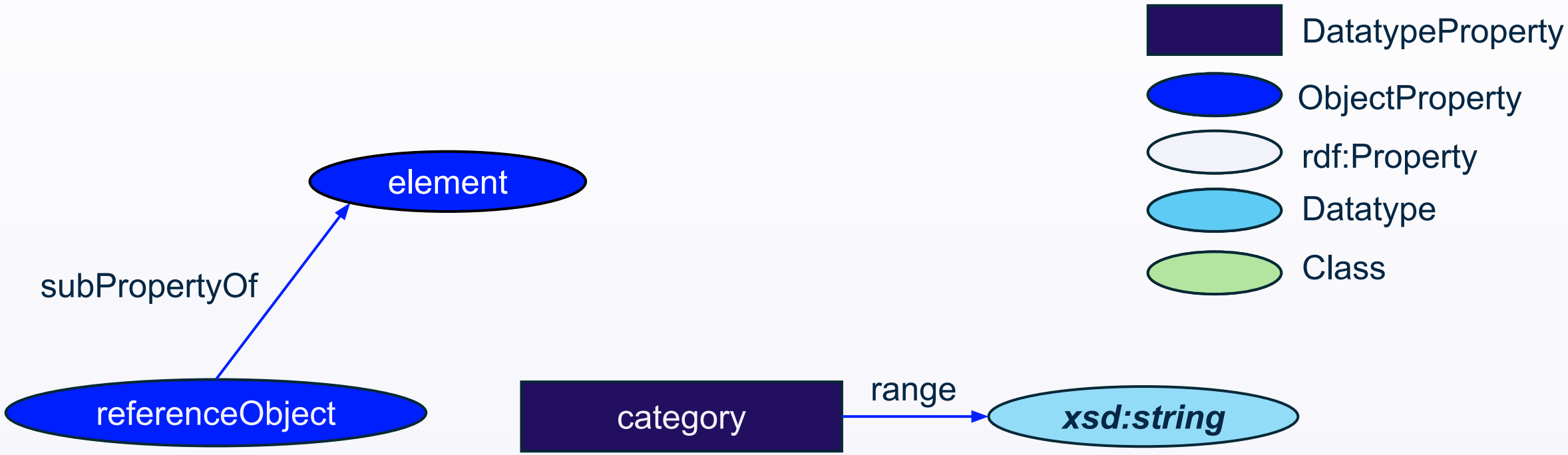
SR-0 There is no node in $V(G)$ that belongs to more than 1 of the classes OWLClass, Datatype, ObjectProperty and DatatypeProperty.

SR-18 If $subPropertyOf(x, y)$, ObjectProperty$(x) \in G$, then ObjectProperty$(y) \in G$.

SR-13 If $range(x, y)$, DatatypeProperty$(x) \in G$, then Datatype$(y) \in G$.

Repairing Ambiguities Example 3/3

GRAPHWISE
AI THRIVES ON WHOLE DATA



SR-0 There is no node in $V(G)$ that belongs to more than 1 of the classes OWLClass, Datatype, ObjectProperty and DatatypeProperty.

SR-18 If $subPropertyOf(x, y)$, ObjectProperty$(x) \in G$, then ObjectProperty$(y) \in G$.

SR-13 If $range(x, y)$, DatatypeProperty$(x) \in G$, then Datatype$(y) \in G$.

Evaluation with published ontologies

GRAPHWISE
AI THRIVES ON WHOLE DATA

We “repaired” the following ontologies:

- Darwin Information Typing Architecture (DITA)
- gist
- European Union Agency for Railways (ERA)
- SemOpenAlex
- EBUCorePlus
- Data Product Model (DPROD)

Evaluation with published ontologies

GRAPHWISE
AI THRIVES ON WHOLE DATA

Ontology	Initial size	Additions	Deletions	$\text{OWL}_{\text{strict}}$ size	Conforms
DITA	490	104	9	585	yes
gist core	3060	50	9	3101	yes
ERA	7163	69	54	7178	yes
SemOpenAlex	848	0	0	848	yes
EBUCorePlus	11930	2044	27	13947	yes
DPROD	147	11	7	151	yes

Table 2. Statistics for $\text{OWL}_{\text{strict}}$ ontologies.

Open questions/Future works:

GRAPHWISE
AI THRIVES ON WHOLE DATA

Conformance for ontologies was successfully achieved, BUT:

- Preferences - define repair choices, e.g. keep rdfs:subPropertyOf
  - which ones? how to specify?
- Additional quality requirements, e.g. labels
- Performance optimizations for repairs (large ontologies)