Knowledge Graphs
RDF(S)

Knowledge Engineering

SS25

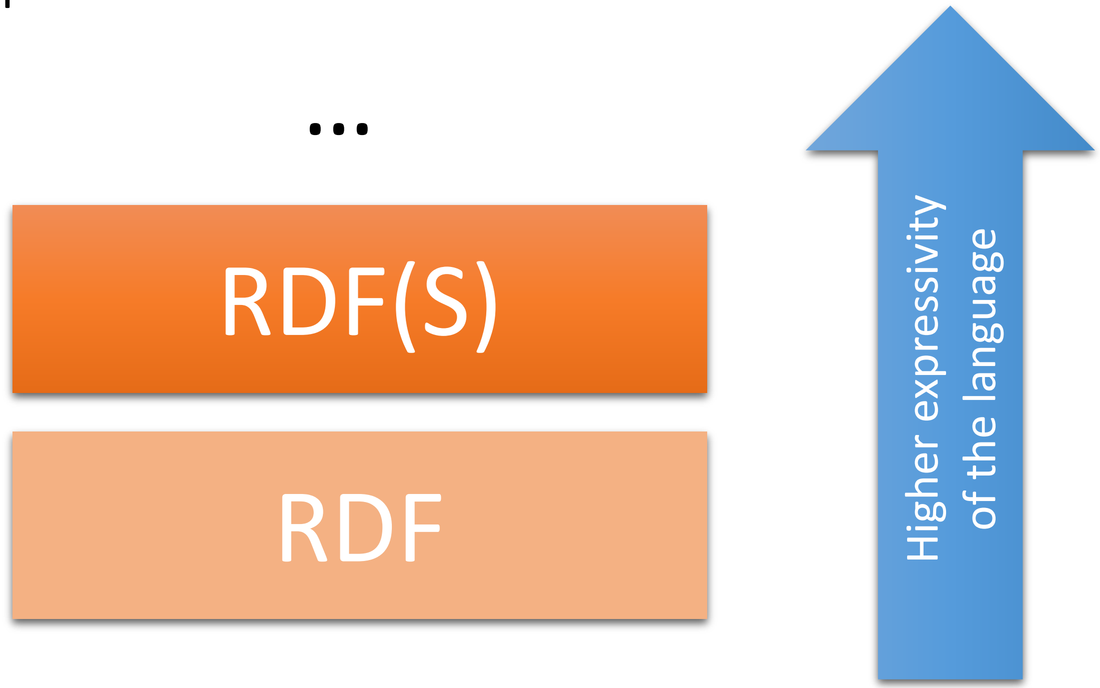
MSc Computer Science

Camerino, 05/05/2025

Prof. Emanuele Laurenzi

RDF Schema

– A schema for RDF graphs.

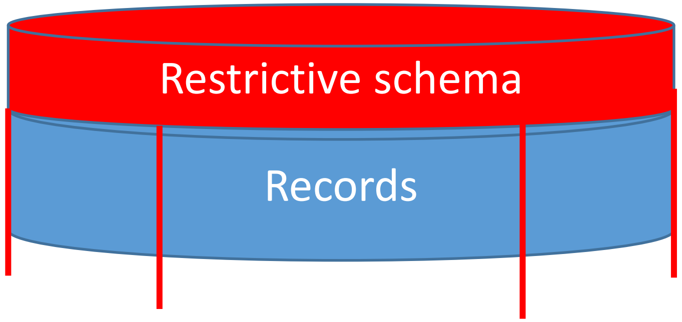


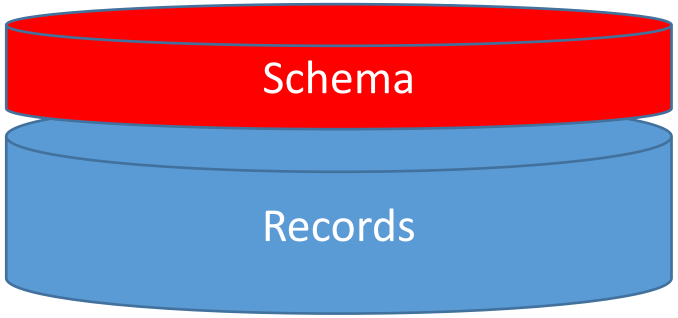
Graph Databases vs. Relational Databases

In SQL databases, you cannot do anything before having a schema (the "DB structure").

In RDF graphs, ***schema is decoupled from "records"***

– Schema can be created after data.
– Schema is optional (data can be queried in the absence of a schema).

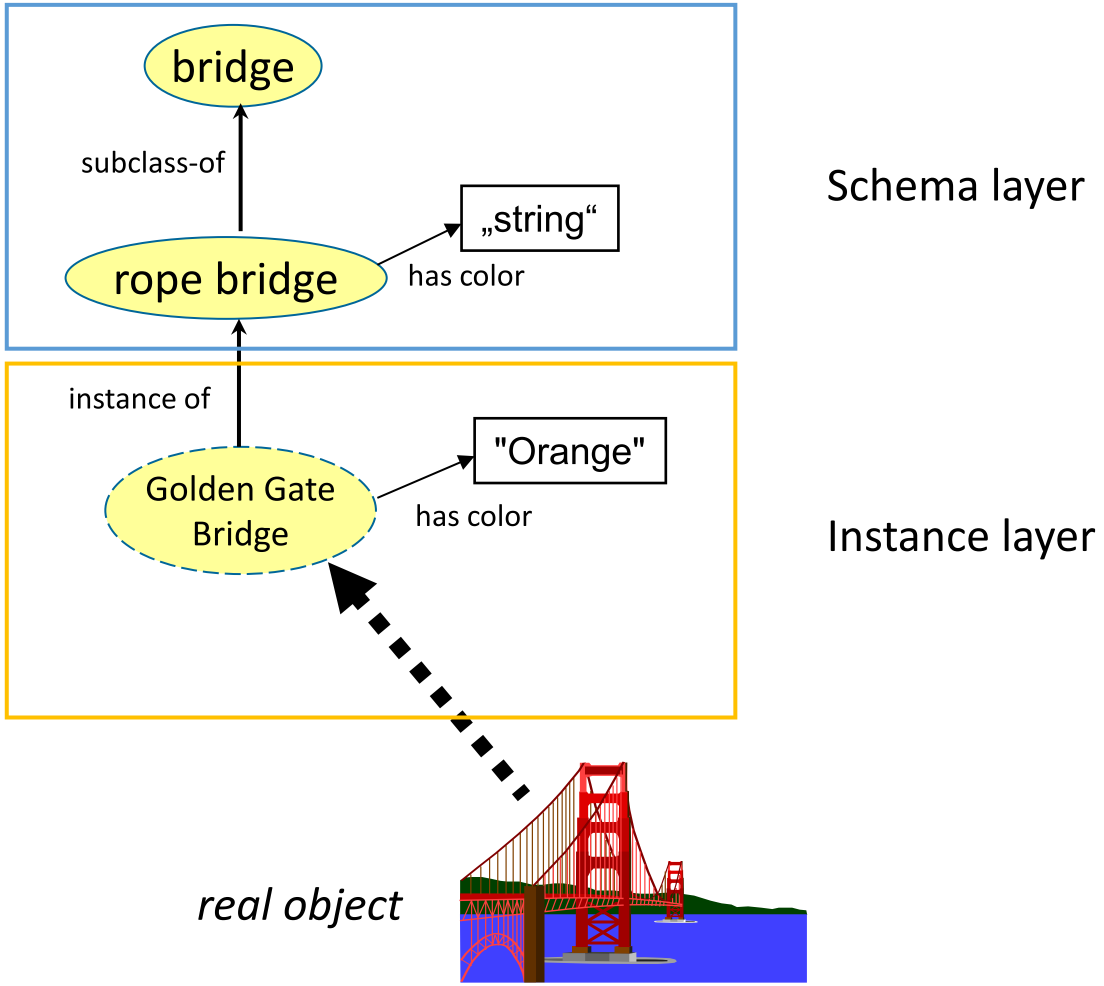




Adapted from lecture of Prof. Dr. Knut Hinkelmann

Key Concepts of RDF Schema

- Classes (Entity Types)
- Sub-class relations
- Property relations
  - Relations
  - Attributes
- Sub-property relations
- Domain and Range restrictions



Adapted from lecture of Prof. Dr. Knut Hinkelmann

Class Hierarchies

- Classes can be organised in hierarchies
  - A is a subclass of B if every instance of A is also an instance of B,
  - Then B is a superclass of A.
- A subclass graph is represented as a tree.
- A class may have multiple superclasses.

- The relationship between sub- and superclass is defined through rdfs:subClassOf.

From lecture of Prof. Dr. Holger Wache

A Graph in the Teaching Domain

with sub-classes

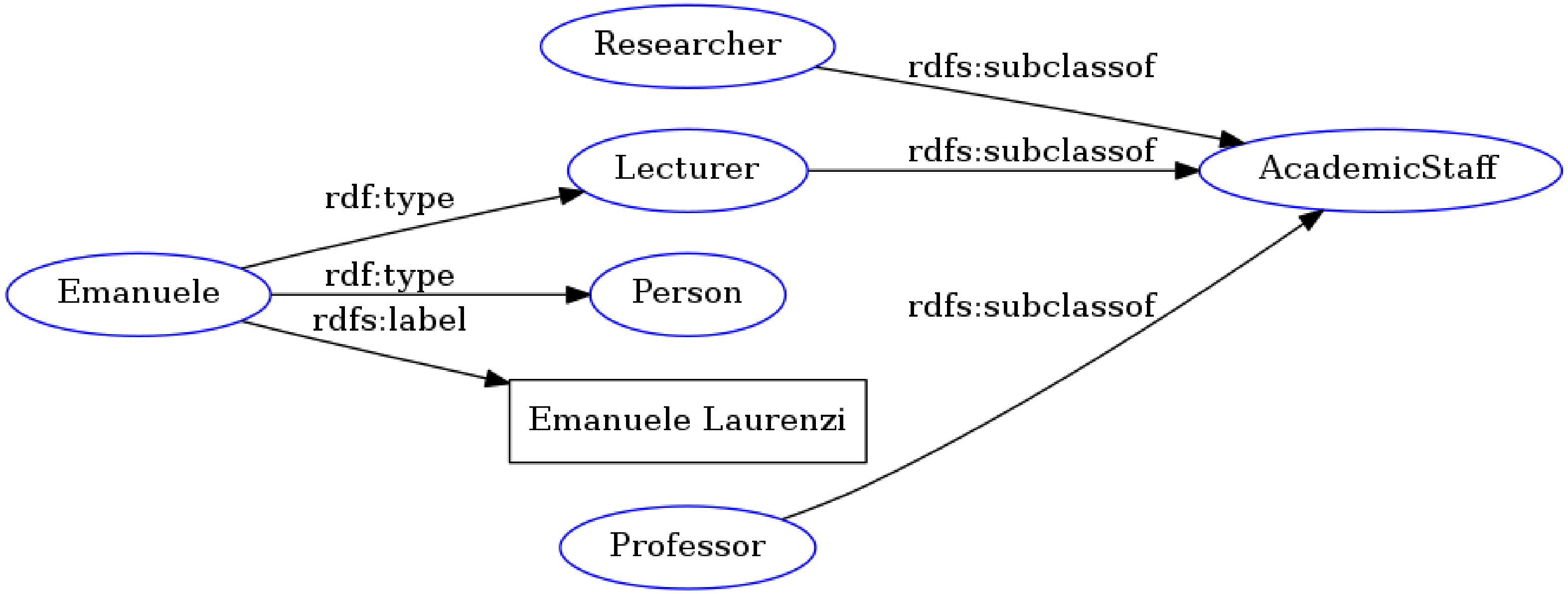
– What are the sub-classes?

```
:Researcher rdfs:subclassof :AcademicStaff.

:Lecturer rdfs:subclassof :AcademicStaff.

:Professor rdfs:subclassof :AcademicStaff.

:Emanuele  rdf:type :Lecturer, :Person ;
           rdfs:label   "Emanuele Laurenzi".
```

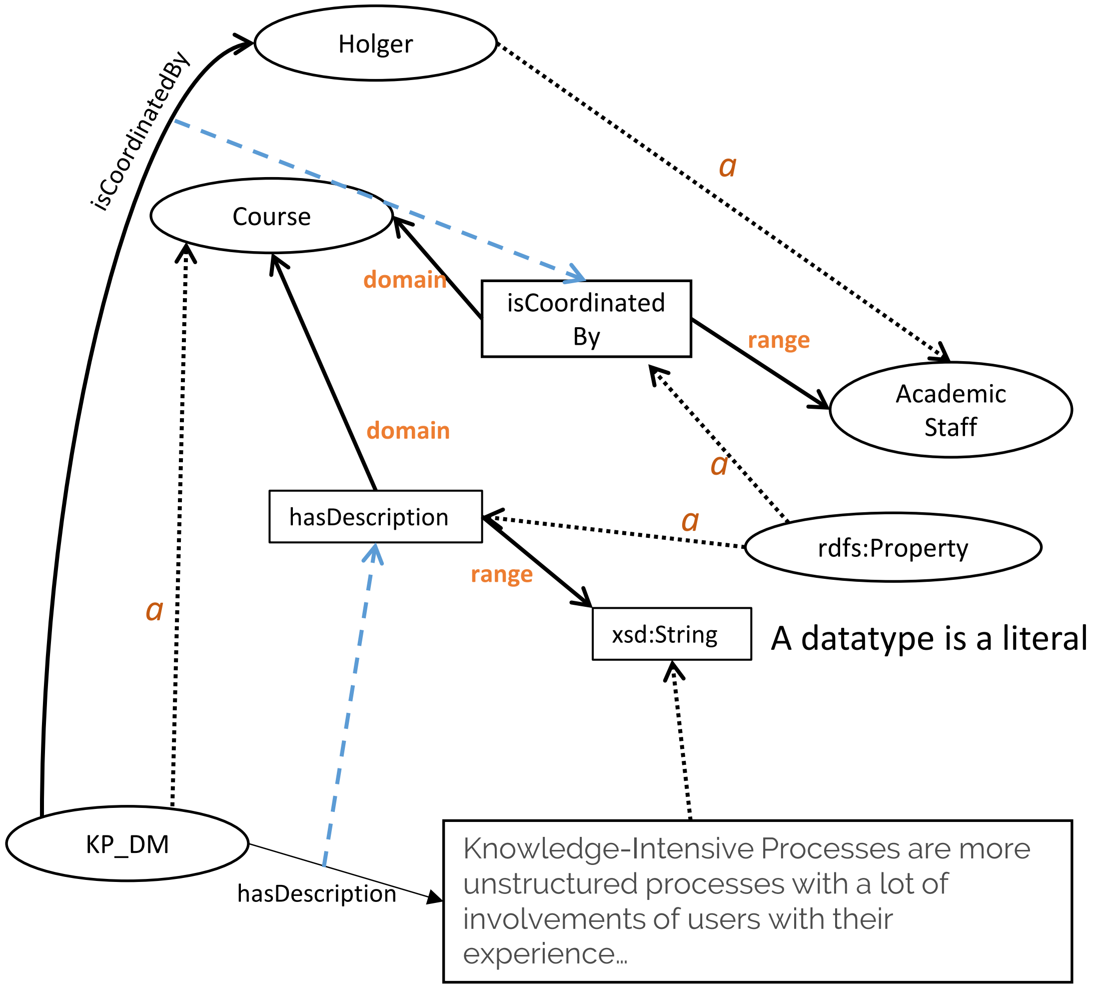


Namespaces:
http://laurenzi.ch#
rdfs: http://www.w3.org/2000/01/rdf-schema#
rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#

Validated with https://www.ldf.fi/service/rdf-grapher

Properties

- In the schema layer, a property is a binary relation that associates instances of one class (Domain) with instances of the target class (Range), e.g.
  - A course is coordinated by an academic staff member,
    - where the property isCoordinatedBy has Domain Course and has Range Academic Staff.
  - A course has a description,
    - where the property hasDescription has Domain Course and has Range xsd:String.
- Domain and Range are defined with the relations rdfs:domain and rdfs:range.



Built-in XML Schema Datatypes

	Datatype	Value space (informative)
Core types	xsd:string	Character strings (but not all Unicode character strings)
	xsd:boolean	true, false
	xsd:decimal	Arbitrary-precision decimal numbers
	xsd:integer	Arbitrary-size integer numbers
IEEE floating-point numbers	xsd:double	64-bit floating point numbers incl. ±Inf, ±0, NaN
	xsd:float	32-bit floating point numbers incl. ±Inf, ±0, NaN
Time and date	xsd:date	Dates (yyyy-mm-dd) with or without timezone
	xsd:time	Times (hh:mm:ss.sss…) with or without timezone
	xsd:dateTime	Date and time with or without timezone
	xsd:dateTimeStamp	Date and time with required timezone
Recurring and partial dates	xsd:gYear	Gregorian calendar year
	xsd:gMonth	Gregorian calendar month
	xsd:gDay	Gregorian calendar day of the month
	xsd:gYearMonth	Gregorian calendar year and month
	xsd:gMonthDay	Gregorian calendar month and day
	xsd:duration	Duration of time
	xsd:yearMonthDuration	Duration of time (months and years only)
	xsd:dayTimeDuration	Duration of time (days, hours, minutes, seconds only)

Limited-range integer numbers	xsd:byte	-128…+127 (8 bit)
	xsd:short	-32768…+32767 (16 bit)
	xsd:int	-2147483648…+2147483647 (32 bit)
	xsd:long	-9223372036854775808…+9223372036854775807 (64 bit)
	xsd:unsignedByte	0…255 (8 bit)
	xsd:unsignedShort	0…65535 (16 bit)
	xsd:unsignedInt	0…4294967295 (32 bit)
	xsd:unsignedLong	0…18446744073709551615 (64 bit)
	xsd:positiveInteger	Integer numbers >0
	xsd:nonNegativeInteger	Integer numbers ≥0
	xsd:negativeInteger	Integer numbers <0
	xsd:nonPositiveInteger	Integer numbers ≤0
Encoded binary data	xsd:hexBinary	Hex-encoded binary data
	xsd:base64Binary	Base64-encoded binary data
Miscellaneous XSD types	xsd:anyURI	Absolute or relative URIs and IRIs
	xsd:language	Language tags per [BCP47]
	xsd:normalizedString	Whitespace-normalized strings
	xsd:token	Tokenized strings
	xsd:NMTOKEN	XML NMTOKENs
	xsd:Name	XML Names
	xsd:NCName	XML NCNames

https://www.w3.org/TR/rdf11-concepts/#section-Datatypes

An example of a RDF(S) graph

containing classes, class hierarchy, instances, and properties

:Researcher rdfs:subclassof :AcademicStaff.

:Lecturer rdfs:subclassof :AcademicStaff.

:Professor rdfs:subclassof :AcademicStaff.

:Emanuele rdf:type :Lecturer, :Person ;

rdfs:label "Emanuele Laurenzi".

:KP_DM rdf:type :Course.

:isTaughtBy rdfs:domain :Course;

rdfs:range :AcademicStaff.

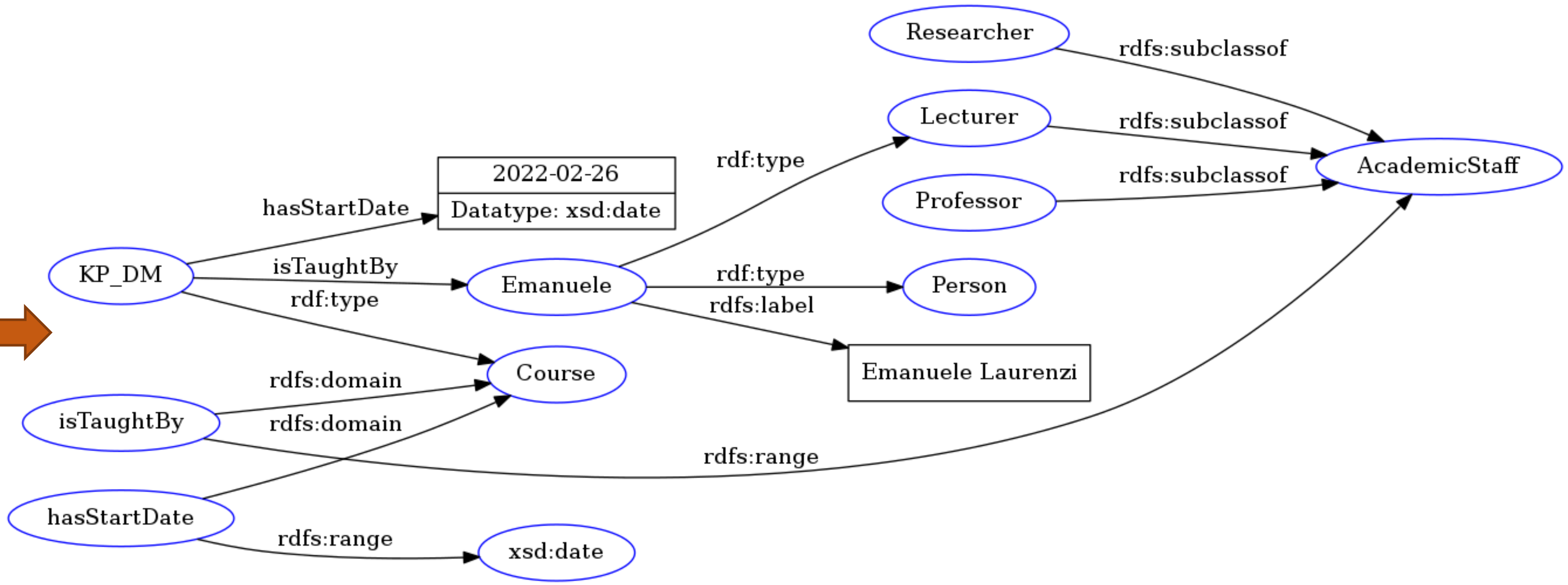
:KP_DM :isTaughtBy :Emanuele.

:hasStartDate rdfs:domain :Course;

rdfs:range xsd:date.

:KP_DM :hasStartDate

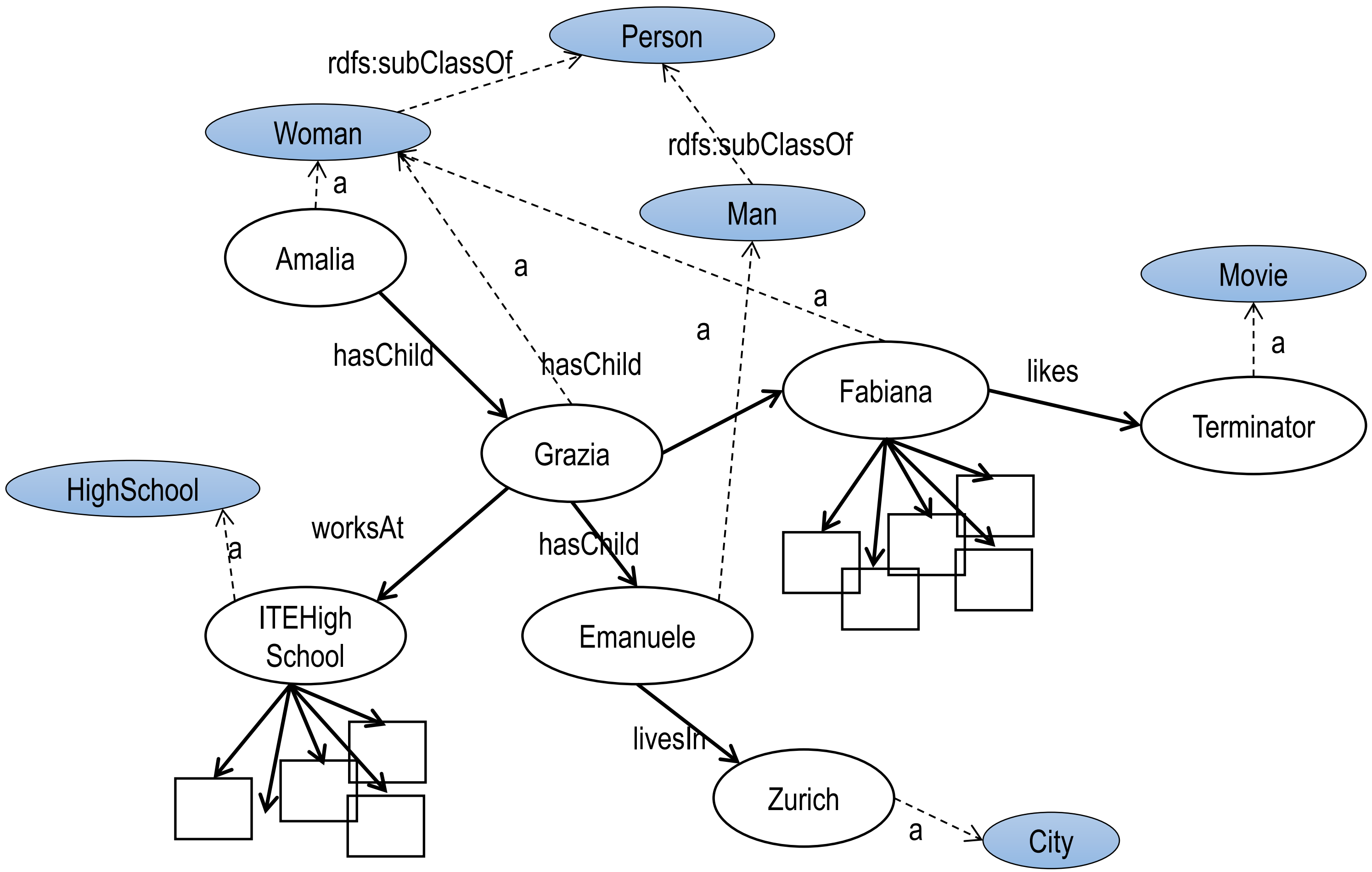
"2022-02-26"^^xsd:date.



Namespaces:
http://laurenzi.ch#
rdfs: http://www.w3.org/2000/01/rdf-schema#
rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#
xsd: http://www.w3.org/2001/XMLSchema#

Exercise

– Given the fragment of this family tree graph, derive the domain and range properties on the schema layer.

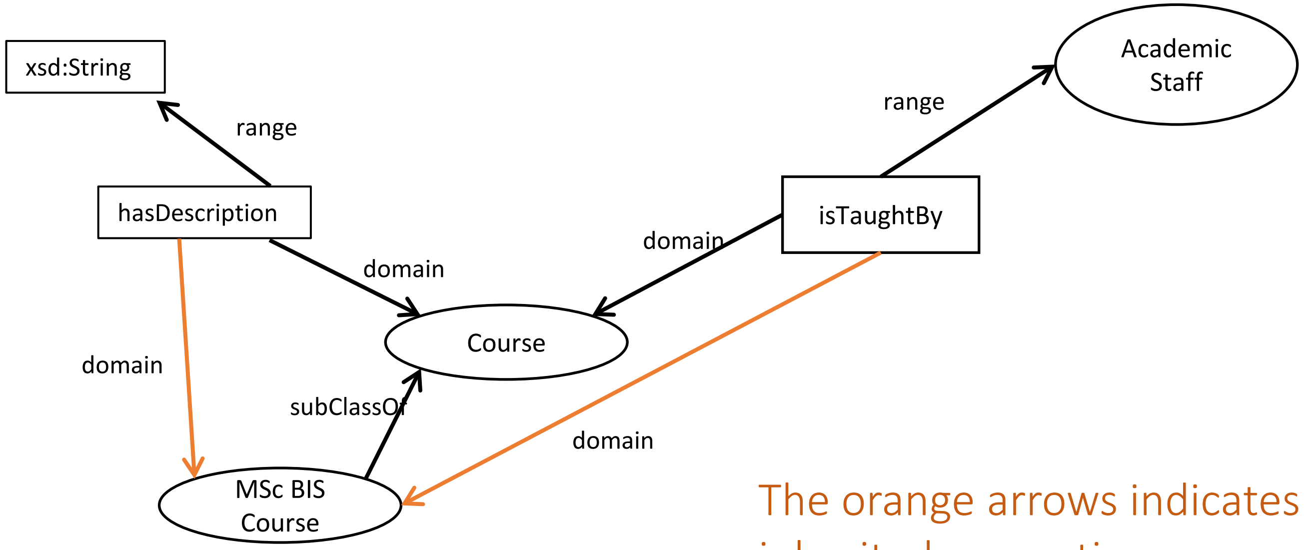


Inheritance in Class Hierarchies

- Properties (i.e. datataypes and relationships) can be inherited by another class.
- The inheriting class is called sub-class.
- The class that inherits is called super-class.
- The inheritance of properties is possible by specifying the rdfs:subClassOf relationships between two classes.

Adapted from Prof. Dr. Holger Wache

An Example for Inheritance



The orange arrows indicates the inherited properties.
Inheritance does not increase the number of statements in the turtle file.

```
:MScBIS_Course
  rdfs:subClassOf :Course .

:hasStartDate
  rdf:type rdf:Property ;
  rdfs:domain :Course ;
  rdfs:range xsd:date .

:isTaughtBy
  rdf:type rdf:Property ;
  rdfs:domain :Course ;
  rdfs:range :AcademicStaff .
```

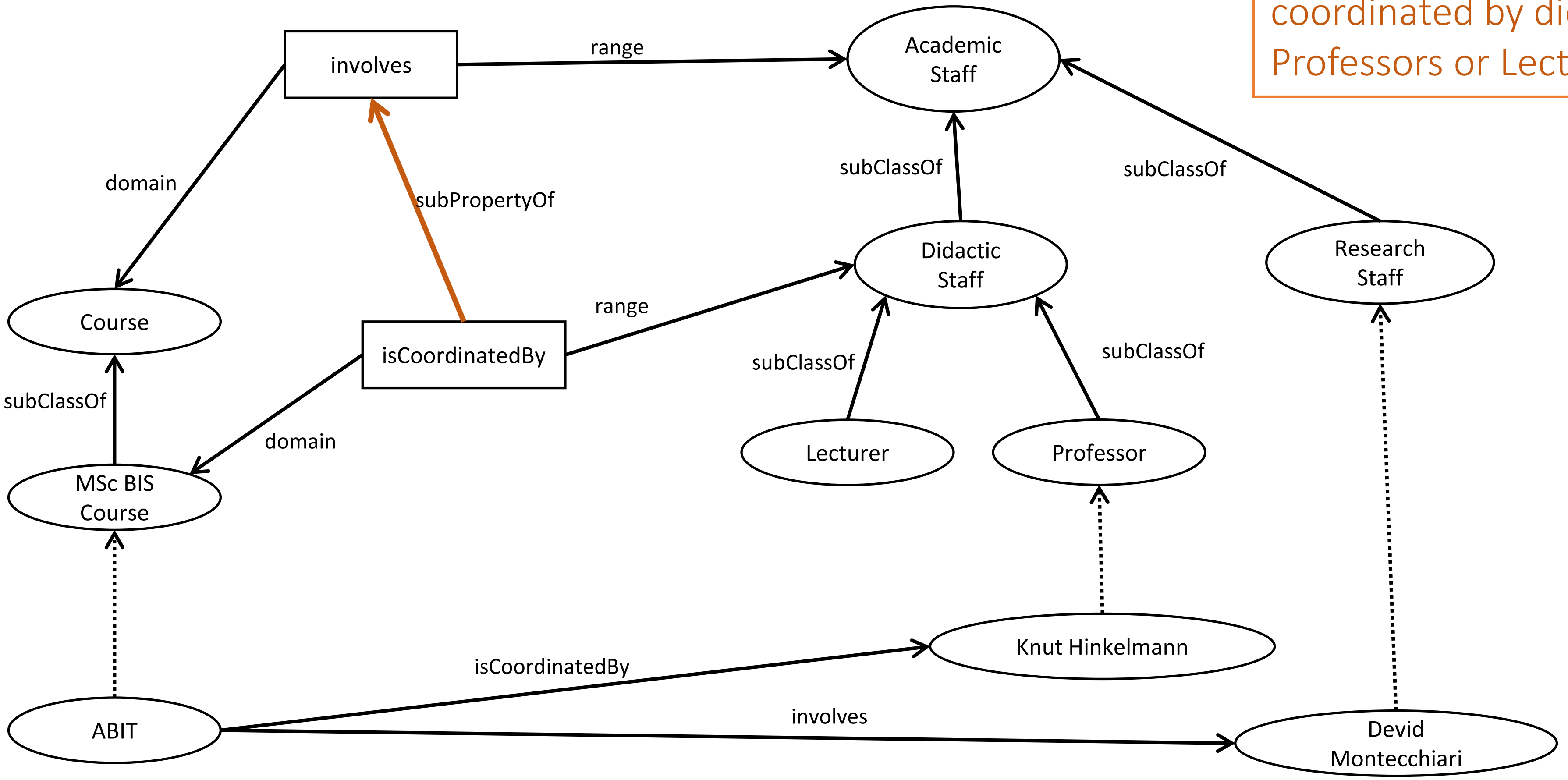
Property Hierarchies

- Hierarchical relationships for properties
  - E.g., "is coordinated by" is a sub-property of "involves"
  - If a course C is coordinated by an academic staff member A, then C also involves A.
- The converse is not necessarily true
  - E.g., A may be the coordinator of course C, or
  - an additional lecturer teaches in course C, or
  - a tutor marks student homework in course C, but they do not not coordinate C.
- P is a sub-property of Q, if Q(x,y) is true whenever P(x,y) is true.
- The relationship between sub- and super-properties is defined through rdfs:subPropertyOf

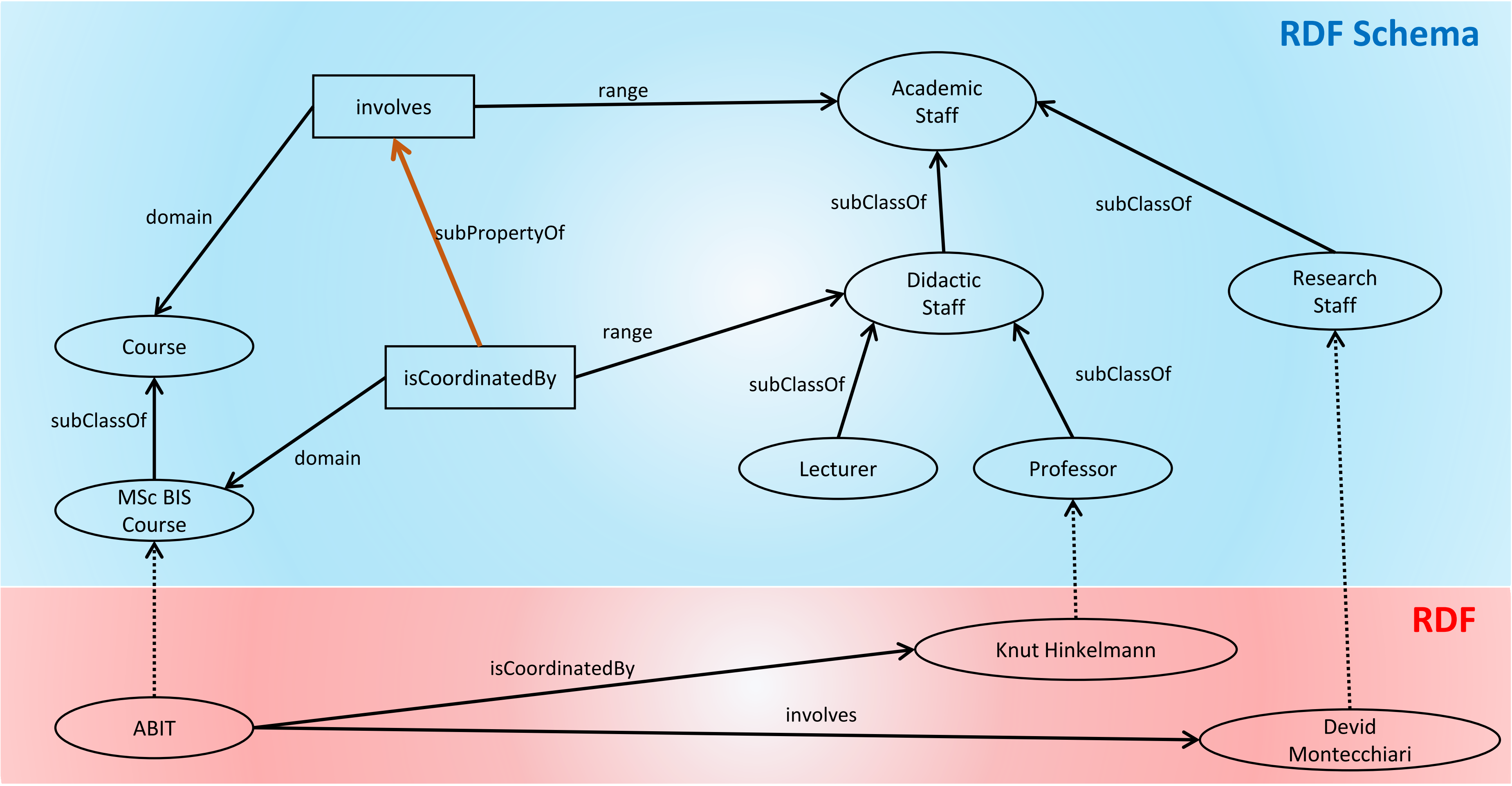
Adapted from lecture of Prof. Dr. Holger Wache

An Example of a Sub-Property

Courses in MSc BIS can only be coordinated by didactic staff: Professors or Lecturers.



RDF Schema Layer and RDF Layer



Adapted from lecture of Prof. Dr. Holger Wache

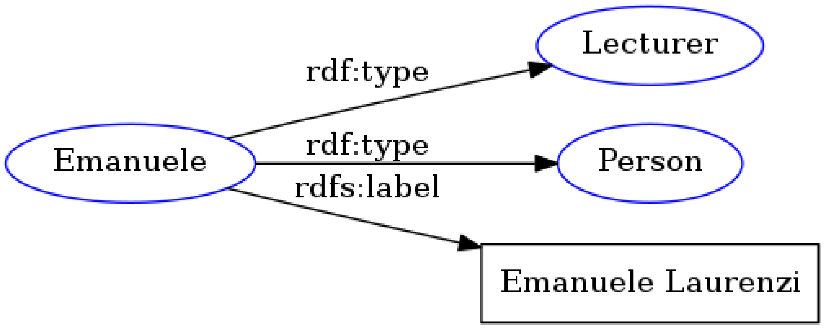
The Property Label

for the good practice.

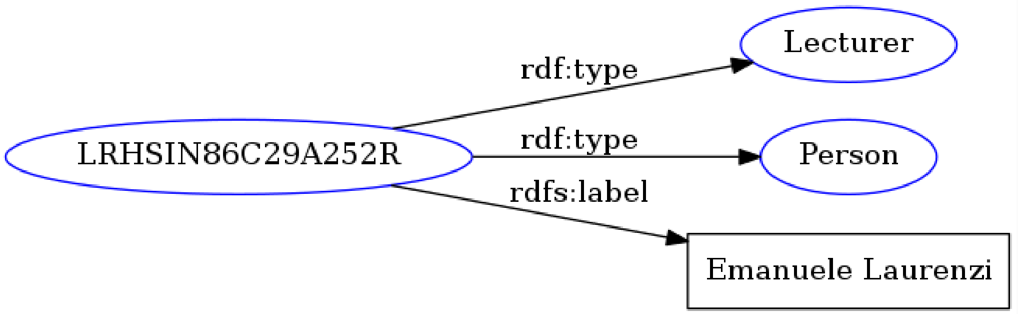
- Every resource (ID/URI) can be associated to a Label -> predefined predicate **rdfs:label**
- A Label provides a human-readable version of a resource's name and is typically displayed in a front-end/web page.
- IDs/URIs act as global identifiers in the entire Web.

No two things in the entire Web should have the same full ID!!!

How many Emanuele do you know?
- Not good as an ID



The good practice:
- Give a unique ID when introducing a new individual/instance.

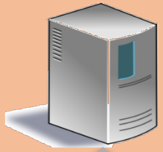


Machine Reasoning with RDFS Semantics

Knowledge-Representation and Reasoning

Repetition

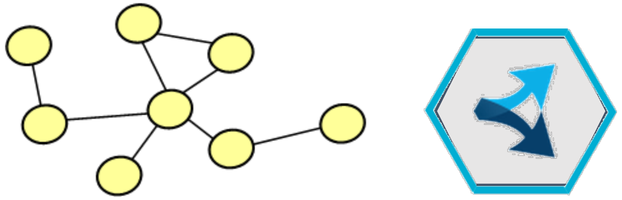
Graph Database/Triplestore and inference engine



consistent

Storing, integration, querying, reasoning.

Knowledge representation formalism



machine-interpretable knowledge

RDF, RDF(S), OWL, SPARQL, SHACL…

Knowledge-based System

consistent

Representation of reality

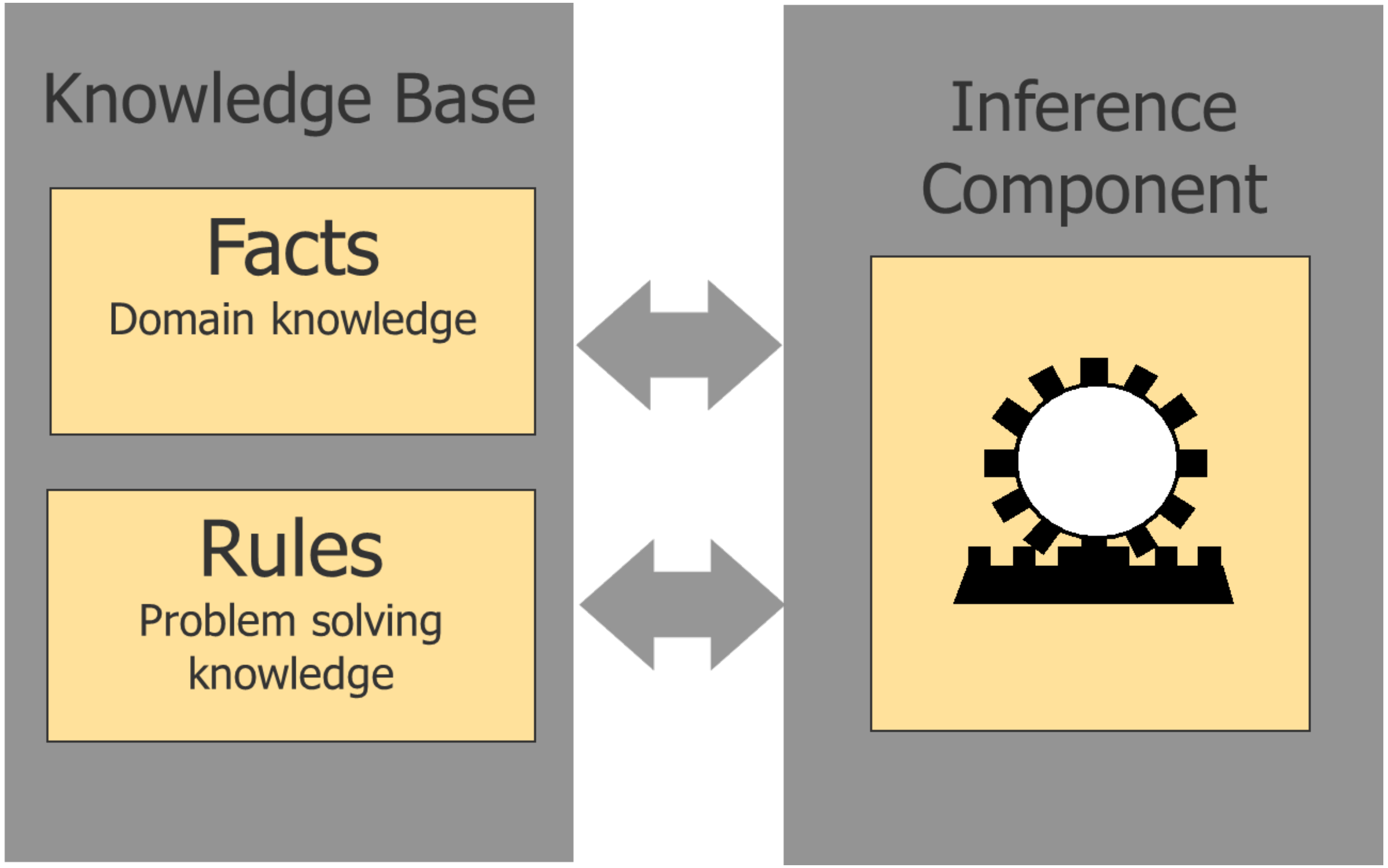
Not formally represented

Adapted from lecture of Prof. Dr. Knut Hinkelmann

Definitions

- The term «knowledge-based systems» is often used synonym for «expert systems». It makes clear that the system has an explicit knowledge base.
- *«An Expert System is an intelligent computer program that uses knowledge and inference procedures to solve problem that are difficult enough to require human expertise for their solutions.»* (Feigenbaum 1982)

Knowledge-Based Systems (Rules & Facts)



Types of Knowledge

Facts: statements about reality

Rules: General proposition about relations or procedure that are valid under specific conditions (e.g. in an „if ... then“-form)

Examples:

Fact:

- Socrates is human

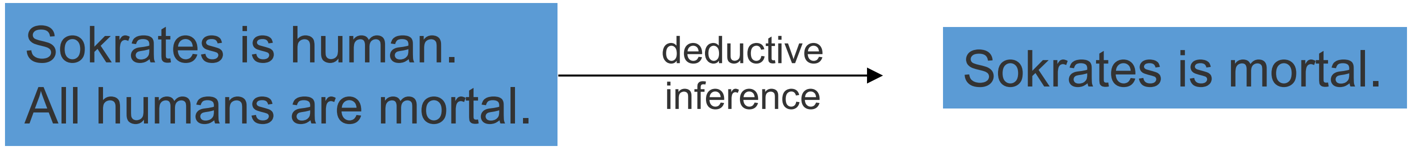
Rule:

- All humans are mortal (IF human THEN mortal)

Prof. Dr. Knut Hinkelmann / Prof. Dr. Holger Wache

Machine Reasoning

- Explicit knowledge:
  - knowledge which is contained in the knowledge base (static knowledge)
- Implicit knowledge:
  - not explicitly stated in the knowledge base
  - is determined from facts by application of rules
- *Derivation = Inference = Reasoning*
  - New knowledge is generated from existing one: Making implicit knowledge explicit

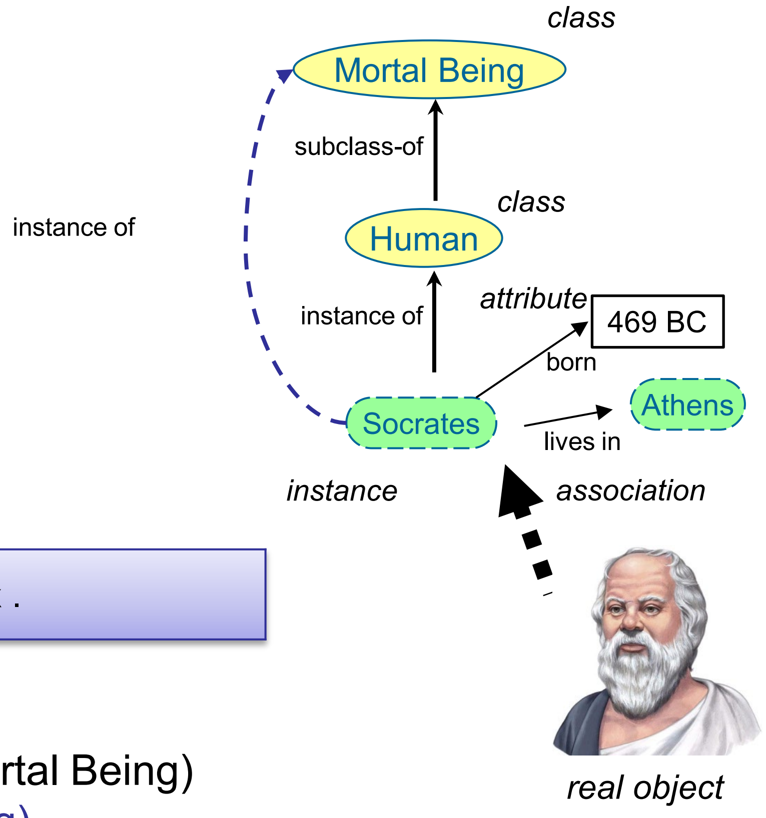


Semantic Rules

- Machine reasoning is enabled by semantic rules
- Semantic (or inference) rules are in the forms:
- **IF E contains certain triples**
  THEN add to E certain additional triples
- where **E** is an arbitrary set of RDF triples.

- Semantic rules are contained in ontology languages, e.g. RDF(S). These are also called entailment rules.
- Additional semantic rules can be created using rule languages like SWRL

An example of an entailment rule



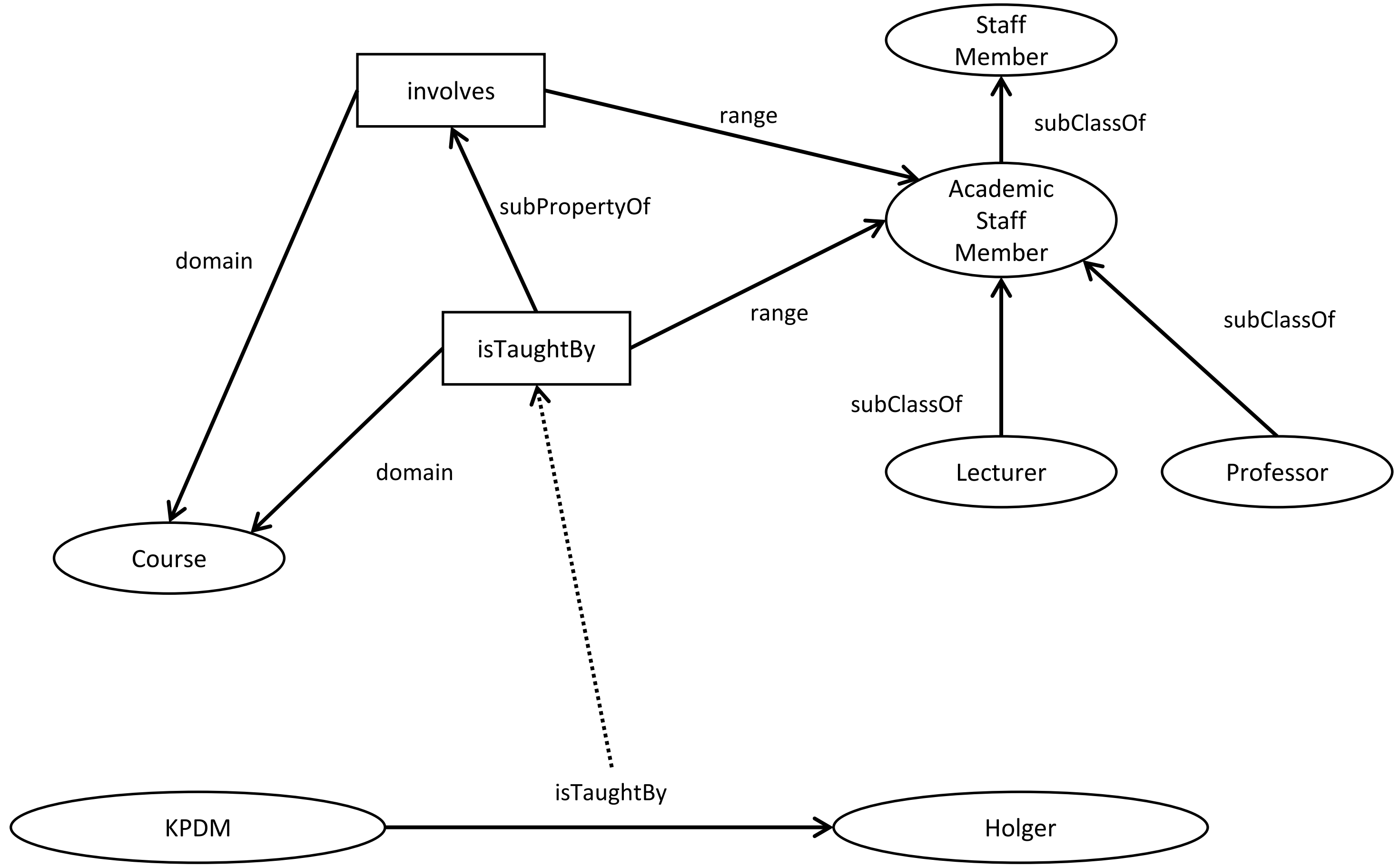
rdfs9	?v rdf:type ?u . ?u rdfs:subClassOf ?x .	?v rdf:type ?x .

IF (Socrate **rdf:type** Human),
(Human **rdfs:subClassOf** Mortal Being)

THEN (Socrate **rdf:type** Mortal Being)

Source: https://www.w3.org/TR/rdf11-mt/

Example for Inferences made by RDF(S)



Entailment Rules of RDF(S)

- RDF(S) comes with 13 built-in inference rules.
- They are typically embedded in the inference engine of Graph databases or Triplestores.
- Reasoning is then performed by triggering the inference engine.

Rule Name	If E contains:	then add:
rdfs1	?u ?p ?n. where ?n is a plain literal (with or without a language tag).	_:nnn rdf:type rdfs:Literal . where _:nnn identifies a blank node allocated to ?n by rule rule lg.
rdfs2	?p rdfs:domain ?x . ?u ?p ?y .	?u rdf:type ?x .
rdfs3	?p rdfs:range ?x . ?u ?p ?v .	?v rdf:type ?x .
rdfs4a	?u ?p ?x .	?u rdf:type rdfs:Resource .
rdfs4b	?u ?p ?v.	?v rdf:type rdfs:Resource .
rdfs5	?u rdfs:subPropertyOf ?v . ?v rdfs:subPropertyOf ?x .	?u rdfs:subPropertyOf ?x .
rdfs6	?u rdf:type rdf:Property .	?u rdfs:subPropertyOf ?u .
rdfs7	?p rdfs:subPropertyOf ?q . ?u ?p ?y .	?u ?q ?y .
rdfs8	?u rdf:type rdfs:Class .	?u rdfs:subClassOf rdfs:Resource .
rdfs9	?u rdfs:subClassOf ?x . ?v rdf:type ?u .	?v rdf:type ?x .
rdfs10	?u rdf:type rdfs:Class .	?u rdfs:subClassOf ?u .
rdfs11	?u rdfs:subClassOf ?v . ?v rdfs:subClassOf ?x .	?u rdfs:subClassOf ?x .
rdfs12	?u rdf:type rdfs:ContainerMembershipProperty .	?u rdfs:subPropertyOf rdfs:member .
rdfs13	?u rdf:type rdfs:Datatype .	?u rdfs:subClassOf rdfs:Literal .

Adapted from lecture of Prof. Dr. Holger Wache

Entailment Rules of RDF(S)

- 13 built-in inference rules divided into
  - "Good" ones and
  - "Bad" ones
- "Good" inference rules add triples to the store which you might want to
- "Bad" inference rules add some strange triples to the store

Rule Name	If E contains:	then add:
rdfs2	?p rdfs:domain ?x . ?u ?p ?y .	?u rdf:type ?x .
rdfs3	?p rdfs:range ?x . ?u ?p ?v .	?v rdf:type ?x .
rdfs5	?u rdfs:subPropertyOf ?v . ?v rdfs:subPropertyOf ?x .	?u rdfs:subPropertyOf ?x .
rdfs7	?p rdfs:subPropertyOf ?q . ?u ?p ?y .	?u ?q ?y .
rdfs9	?u rdfs:subClassOf ?x . ?v rdf:type ?u .	?v rdf:type ?x .
rdfs11	?u rdfs:subClassOf ?v . ?v rdfs:subClassOf ?x .	?u rdfs:subClassOf ?x .

Rule Name	If E contains:	then add:
rdfs6	?u rdf:type rdf:Property .	?u rdfs:subPropertyOf ?u .
rdfs10	?u rdf:type rdfs:Class .	?u rdfs:subClassOf ?u .
rdfs1	?u ?p ?n. where ?n is a plain literal (with or without a language tag).	_:nnn rdf:type rdfs:Literal . where _:nnn identifies a blank node allocated to ?n by rule lg.
rdfs4a	?u ?p ?x .	?u rdf:type rdfs:Resource .
rdfs4b	?u ?p ?v.	?v rdf:type rdfs:Resource .
rdfs8	?u rdf:type rdfs:Class .	?u rdfs:subClassOf rdfs:Resource .
rdfs12	?u rdf:type rdfs:ContainerMembershipProperty .	?u rdfs:subPropertyOf rdfs:member .
rdfs13	?u rdf:type rdfs:Datatype .	?u rdfs:subClassOf rdfs:Literal .

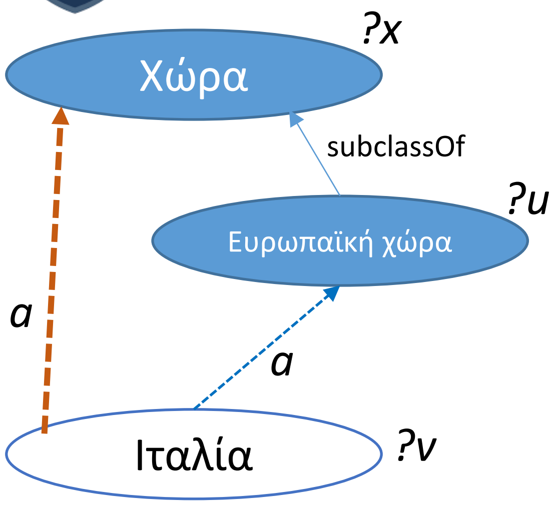
Adapted from lecture of Prof. Dr. Holger Wache

RDF(S) Entailments: Examples

rdfs9	?v rdf:type ?u . ?u rdfs:subClassOf ?x .	?v rdf:type ?x .

IF (Ιταλία **rdf:type** Ευρωπαϊκή χώρα),
(Ευρωπαϊκή χώρα **rdfs:subClassOf** Χώρα)

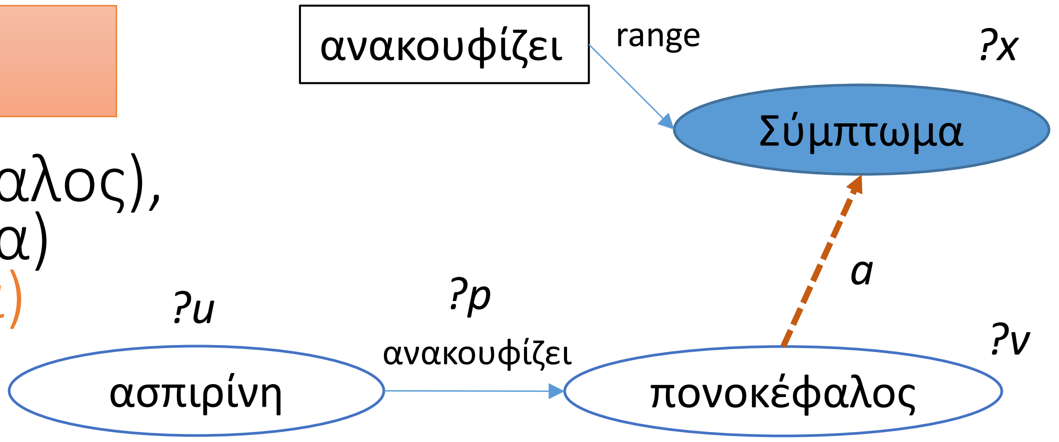
THEN (Ιταλία **rdf:type** Χώρα)



rdfs3	?u ?p ?v . ?p rdfs:range ?x .	?v rdf:type ?x .

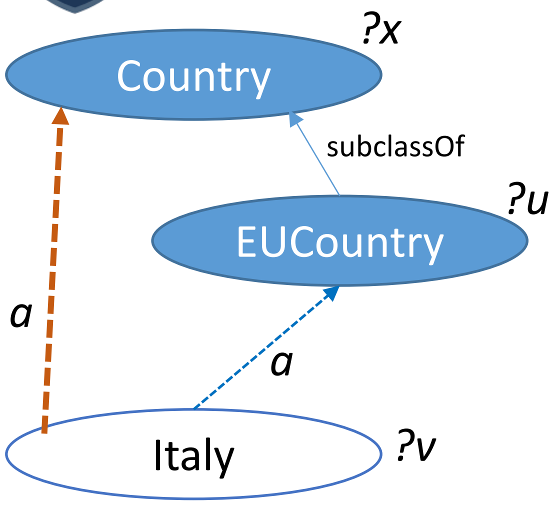
IF (ασπιρίνη, ανακουφίζει, πονοκέφαλος),
(ανακουφίζει **rdfs:range** Σύμπτωμα)

THEN (πονοκέφαλος **rdf:type** Σύμπτωμα)



Adapted from lecture of Prof. Dr. Holger Wache

RDF(S) Entailments: Examples



rdfs9	?v rdf:type ?u . ?u rdfs:subClassOf ?x .	?v rdf:type ?x .

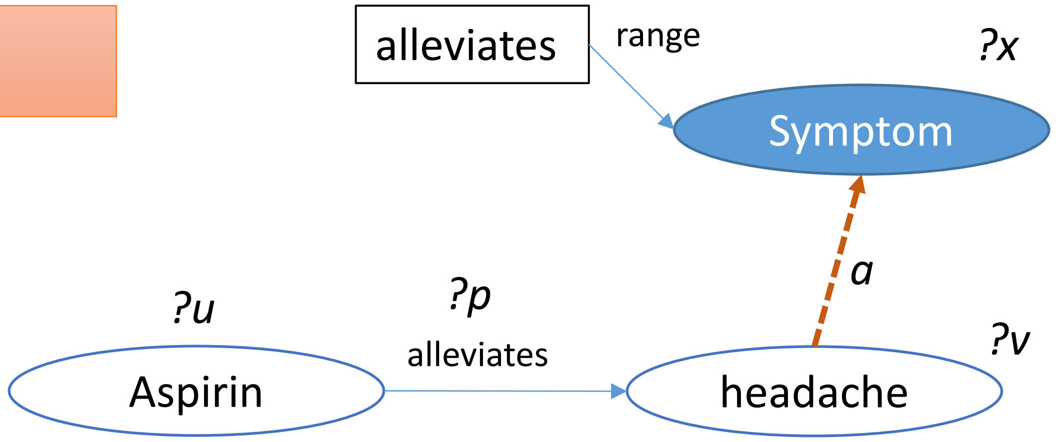
IF (Italy **rdf:type** EuropeanCountry),
(EuropeanCountry **rdfs:subClassOf** Country)

THEN (Italy **rdf:type** Country)

rdfs3	?u ?p ?v . ?p rdfs:range ?x .	?v rdf:type ?x .

IF (aspirin, alleviates, headache),
(alleviates **rdfs:range** Symptom)

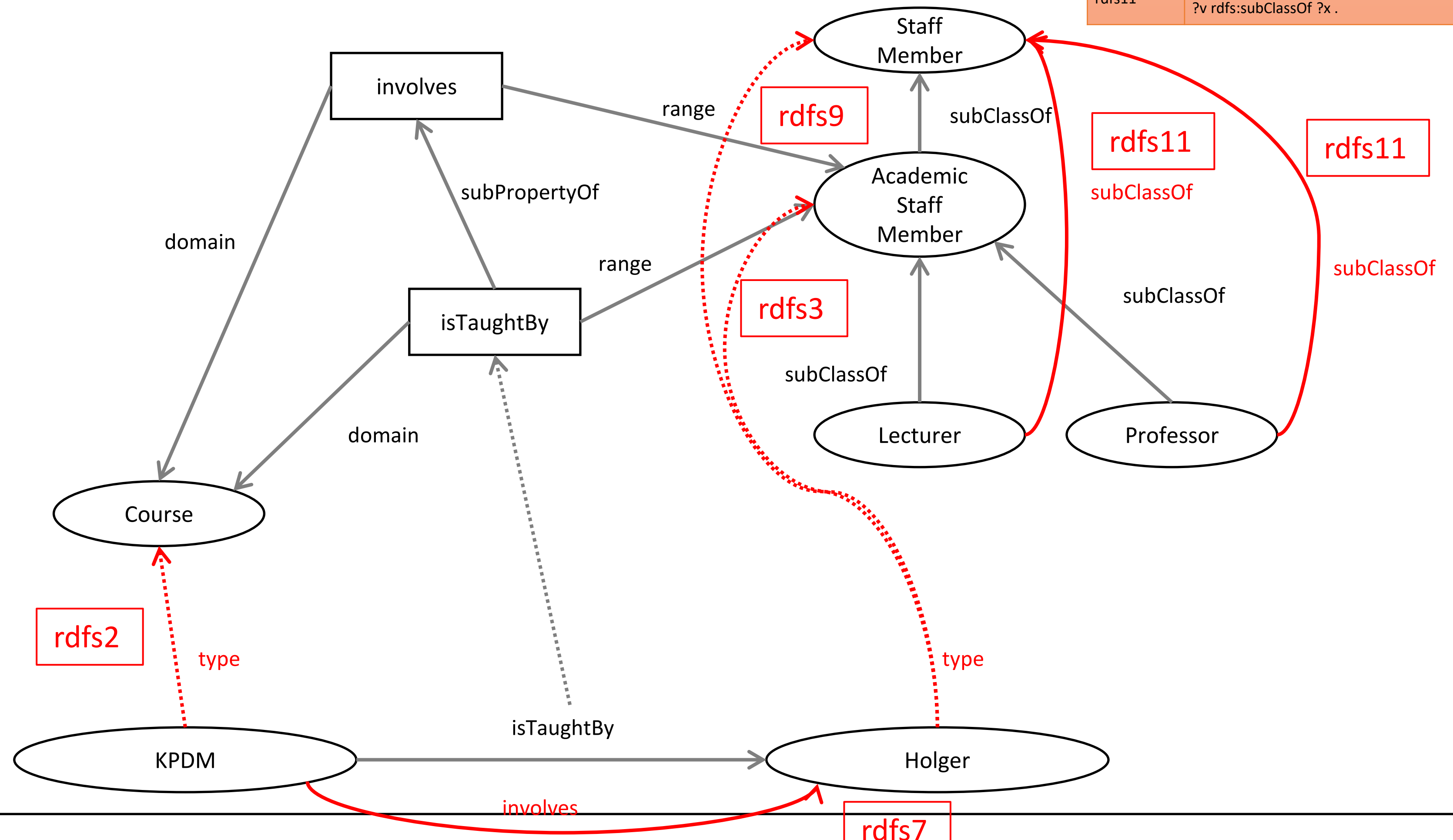
THEN (headache **rdf:type** Symptom)



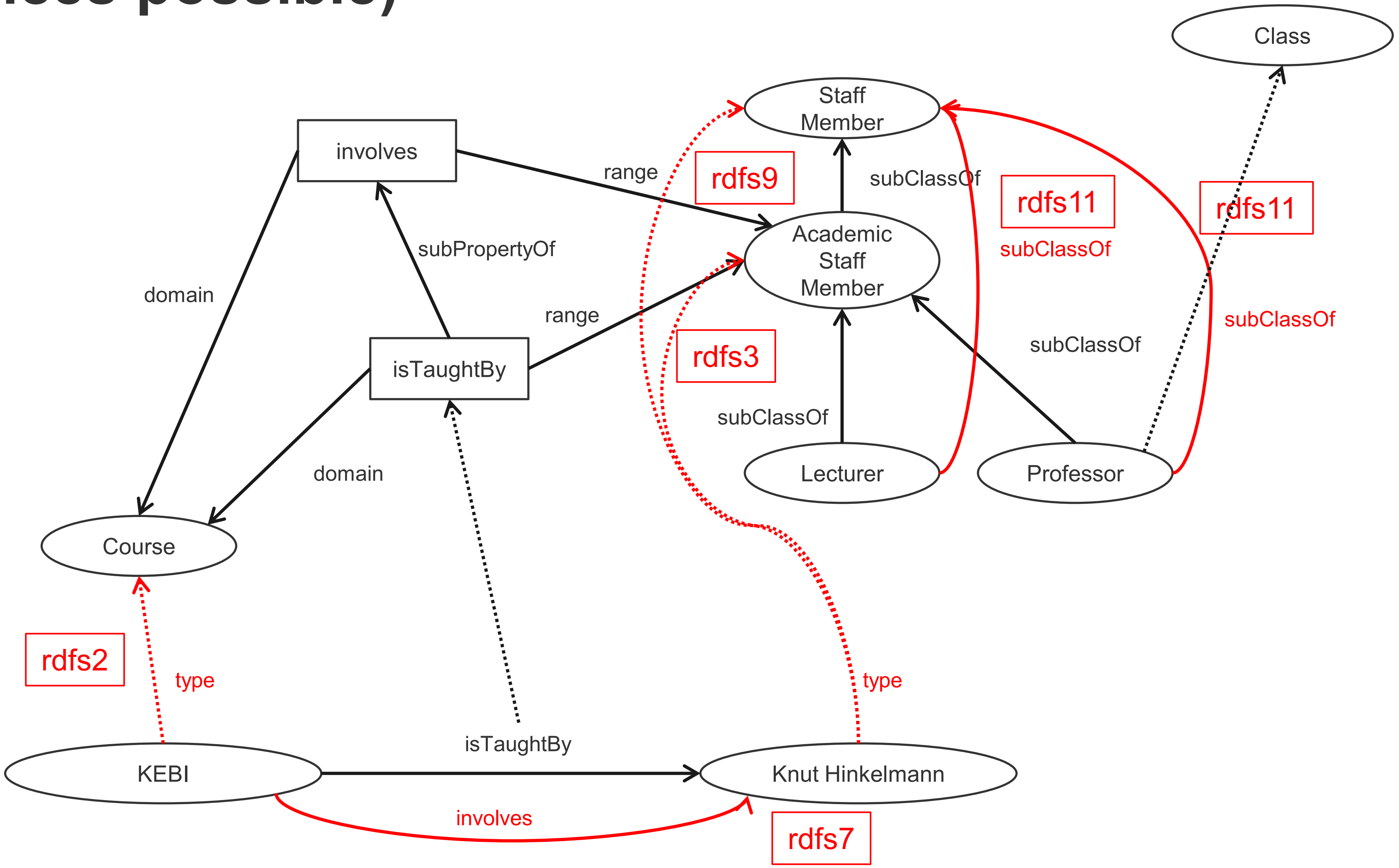
Adapted from lecture of Prof. Dr. Holger Wache

Example for Inferences with the Help of Rules

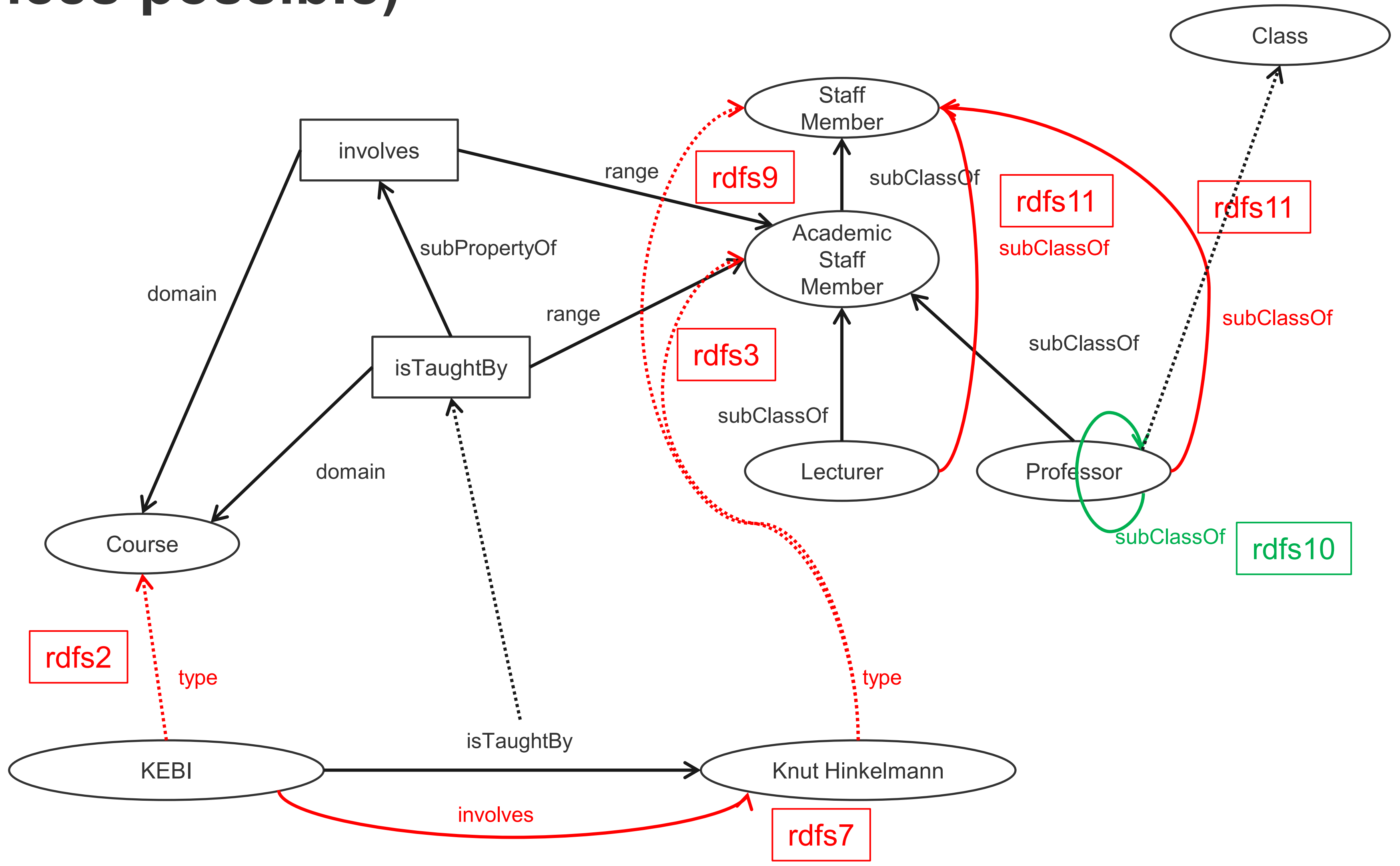
Rule Name	If E contains:	then add:
rdfs2	?p rdfs:domain ?x . ?u ?p ?y .	?u rdf:type ?x .
rdfs3	?p rdfs:range ?x . ?u ?p ?v .	?v rdf:type ?x .
rdfs5	?u rdfs:subPropertyOf ?v . ?v rdfs:subPropertyOf ?x .	?u rdfs:subPropertyOf ?x .
rdfs7	?p rdfs:subPropertyOf ?q . ?u ?p ?y .	?u ?q ?y .
rdfs9	?u rdfs:subClassOf ?x . ?v rdf:type ?u .	?v rdf:type ?x .
rdfs11	?u rdfs:subClassOf ?v . ?v rdfs:subClassOf ?x .	?u rdfs:subClassOf ?x .



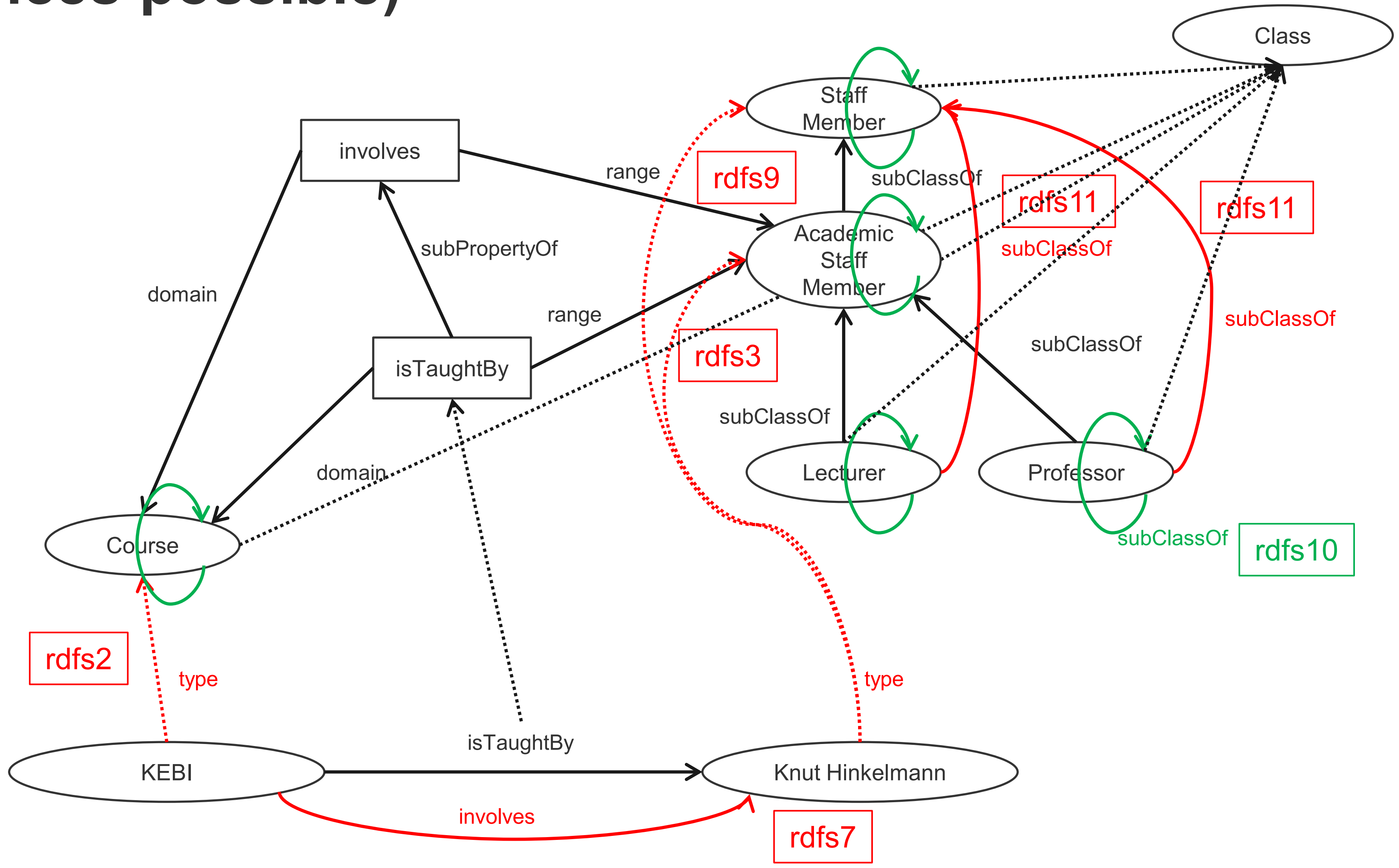
Example for Inferences made by Inference Rules (all inferences possible)



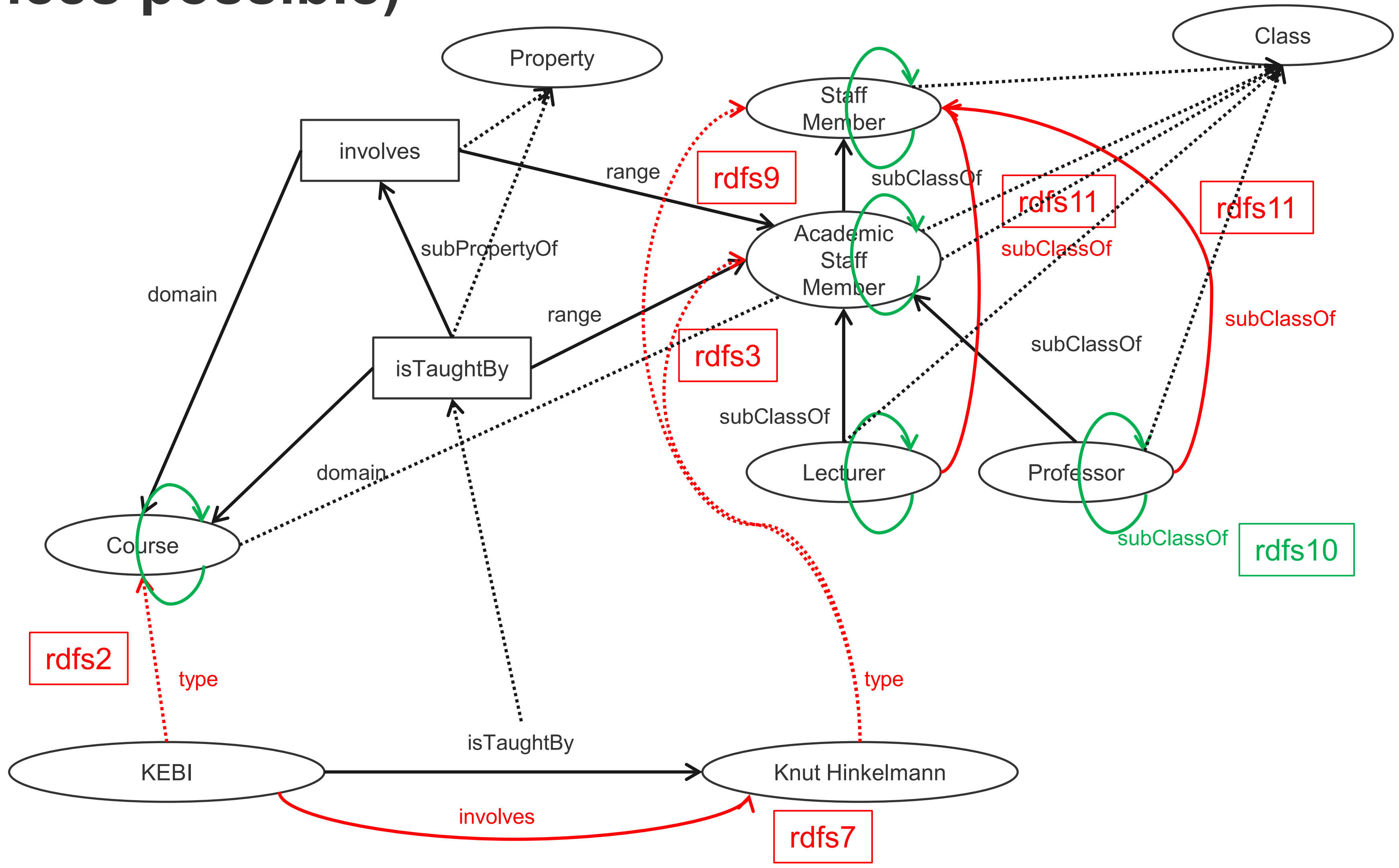
Example for Inferences made by Inference Rules (all inferences possible)



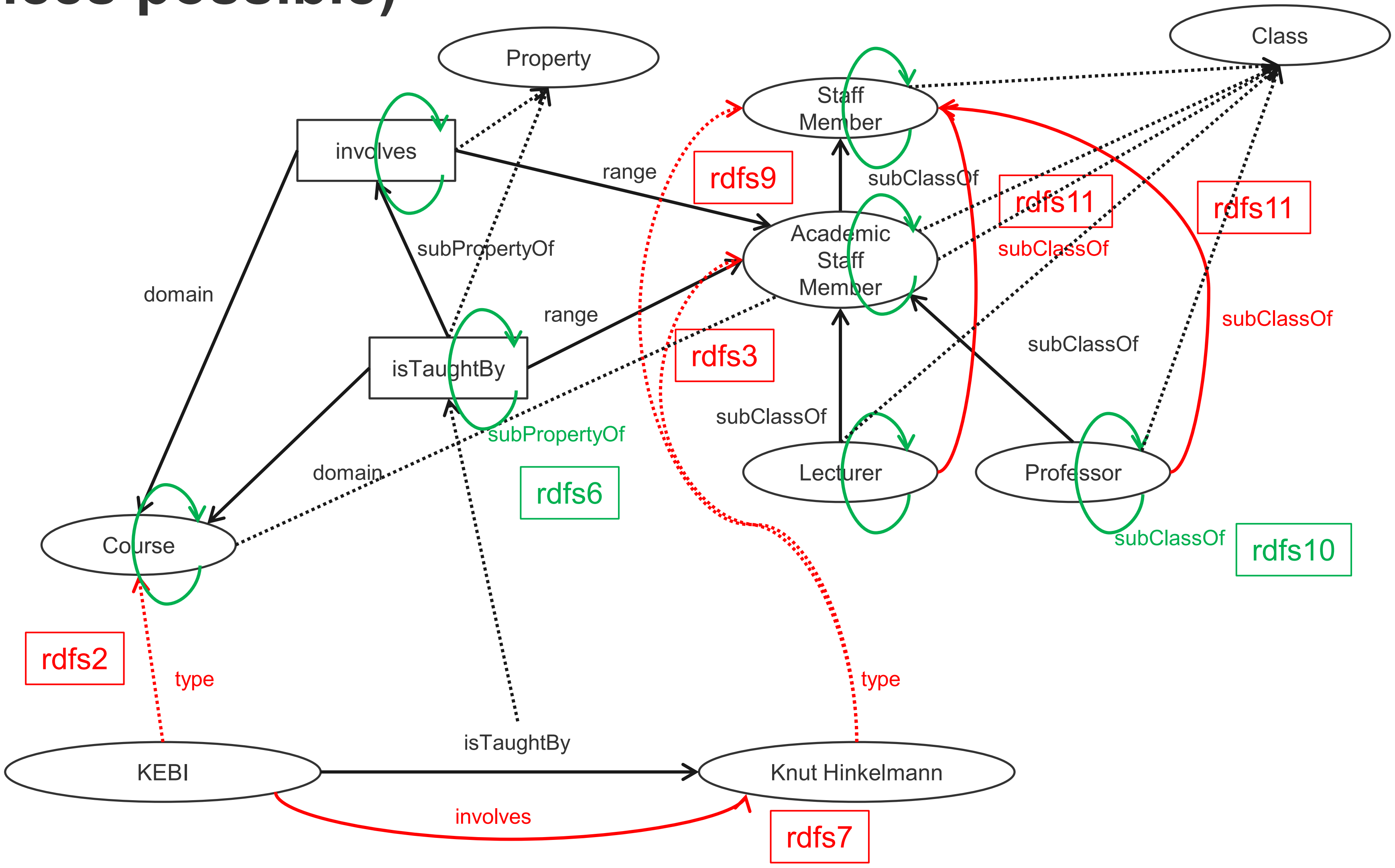
Example for Inferences made by Inference Rules (all inferences possible)



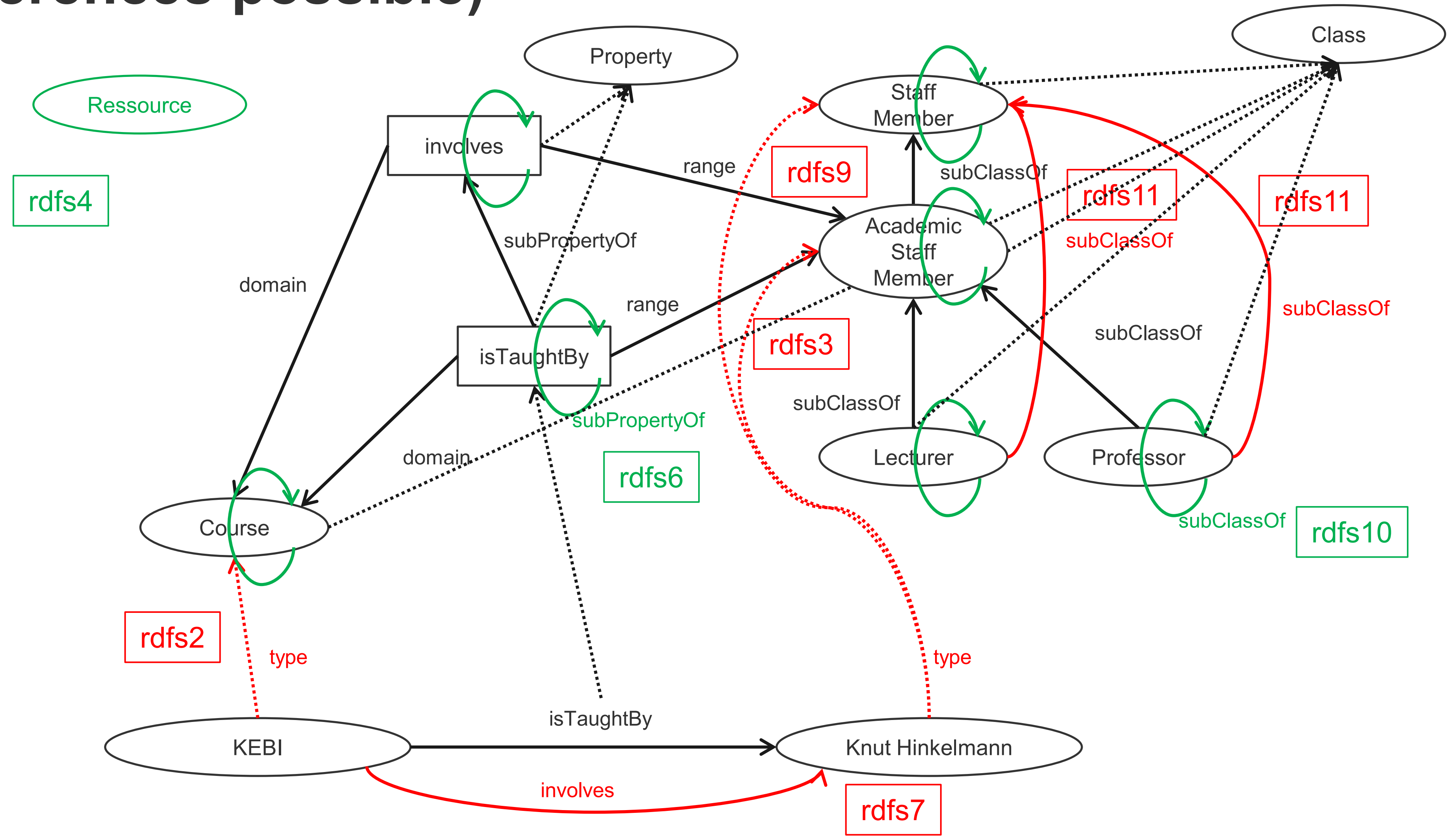
Example for Inferences made by Inference Rules (all inferences possible)



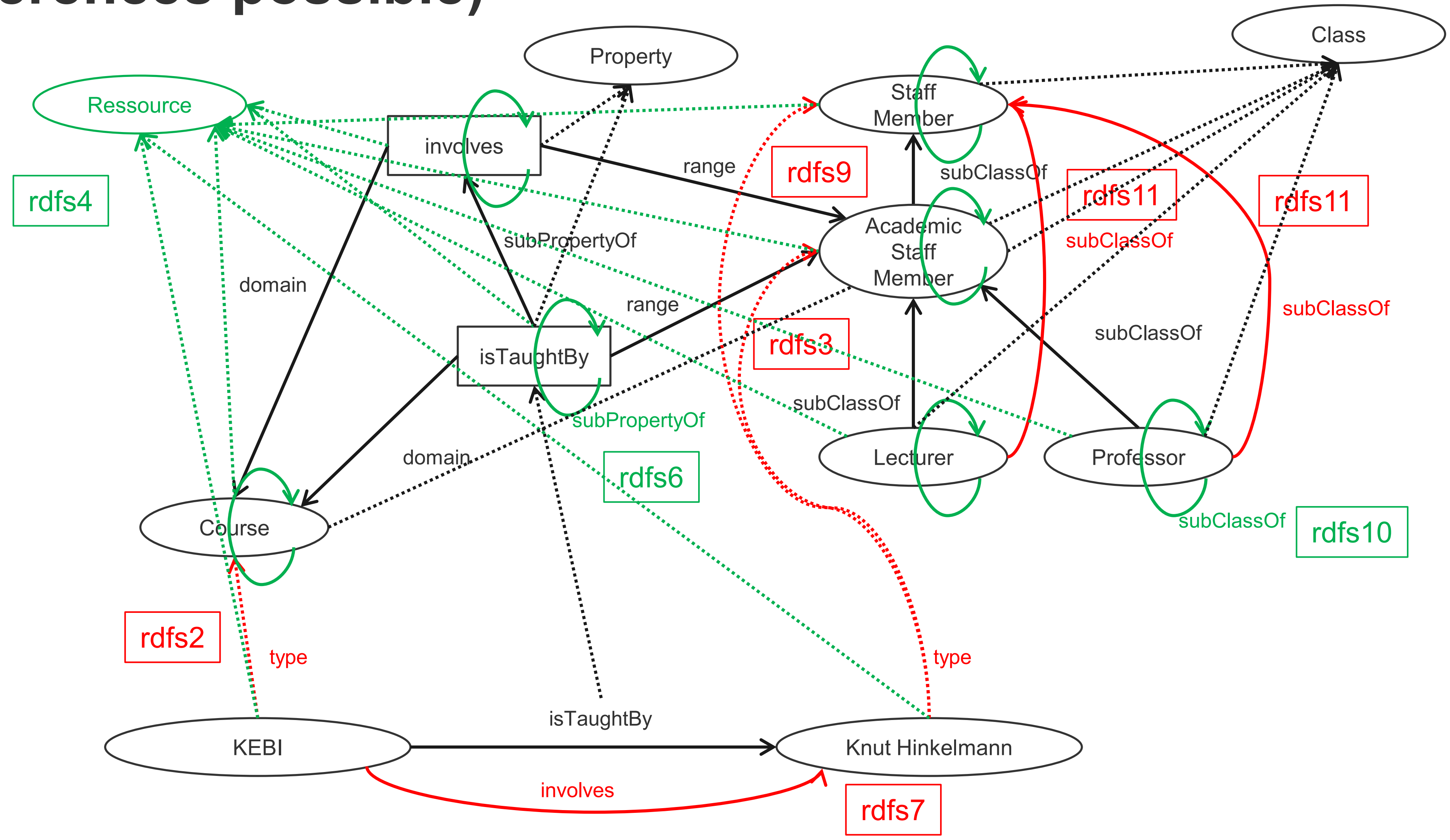
Example for Inferences made by Inference Rules (all inferences possible)



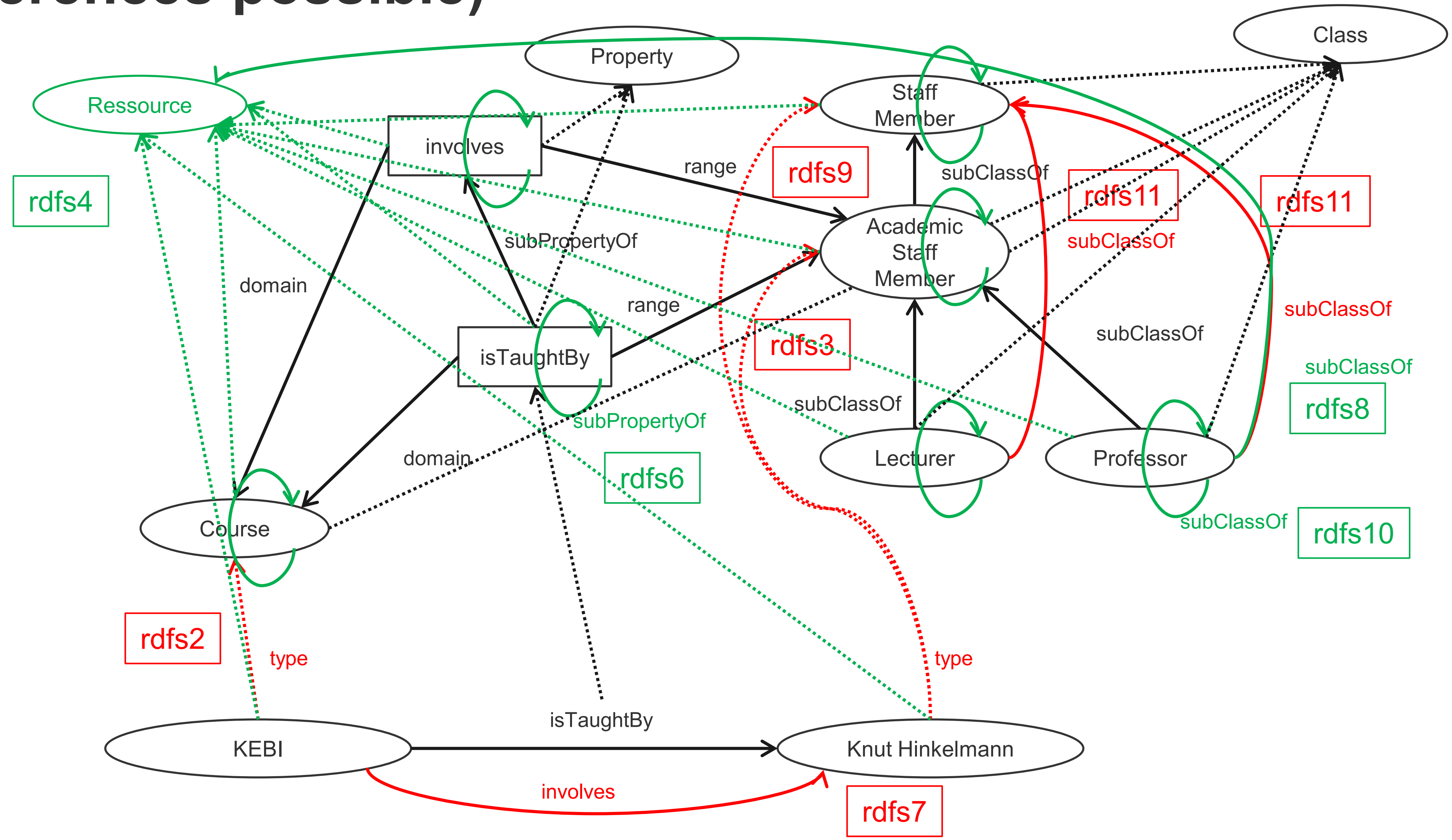
Example for Inferences made by Inference Rules (all inferences possible)



Example for Inferences made by Inference Rules (all inferences possible)



Example for Inferences made by Inference Rules (all inferences possible)



Exercise

- Suppose the following RDF/S graph is given defining the schema for this task: (Please note that some unnecessary details are omitted)
- RDF and RDF/S come with reasoning that can be interpreted as extending the knowledge base. Complete the following RDF graph by introducing all inferable properties. The classes from the schema must not be copied.