RDFS

in a nutshell

fabien, gandon, inria

RDF is a triple model *i.e.* every piece of knowledge is broken down into

(subject , predicate , object)

RDFS stands for RDF Schema

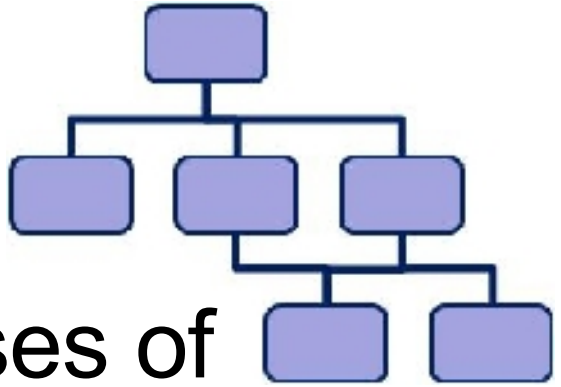
RDFS provides primitives to write lightweight schemas for RDF triples

a little drop of semantics goes a **long way**

RDFS provides primitives to...

... define the vocabulary used in triples

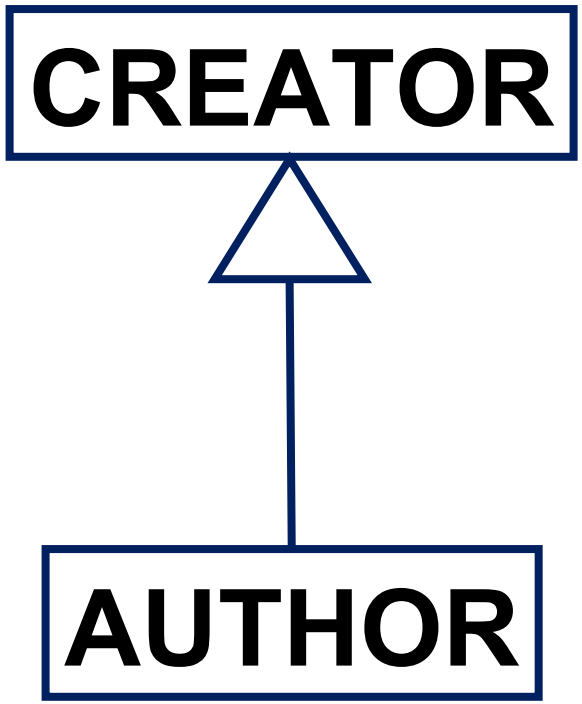
... define elementary inferences



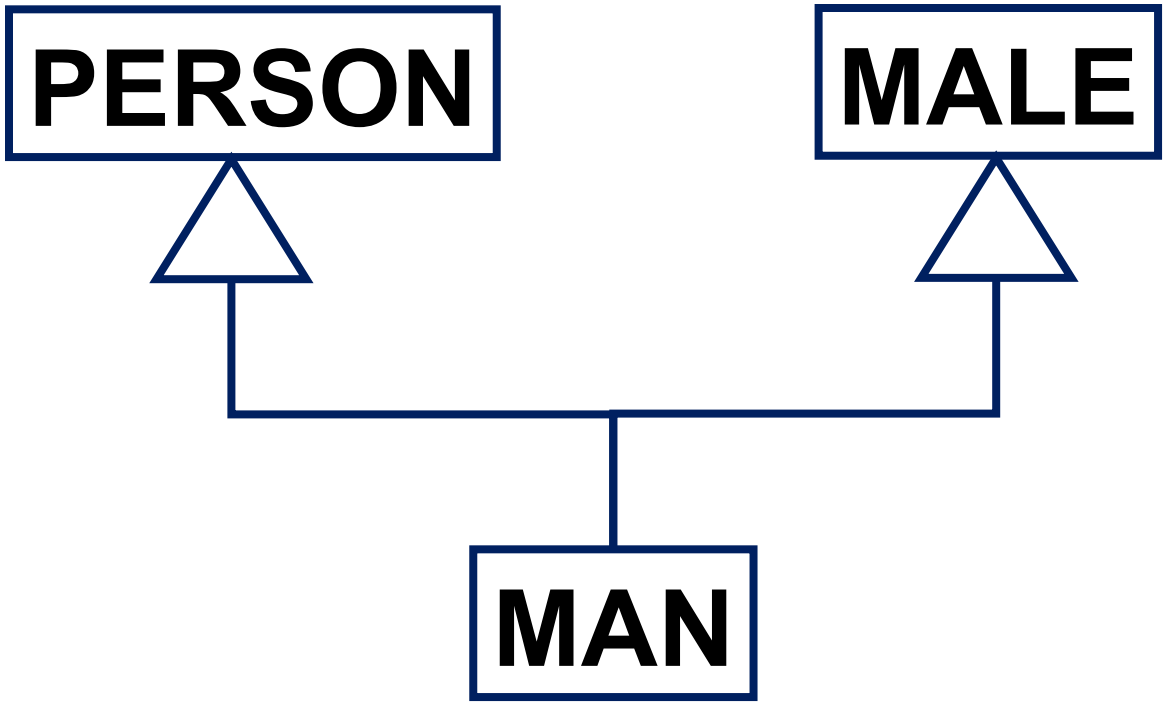
RDFS to define classes of resources and organize their hierarchy


DOCUMENT
REPORT

RDFS to define relations between resources and organize their hierarchy


CREATOR
AUTHOR

RDFS allows for multiple inheritance for classes and properties


PERSON
MALE
MAN

RDFS relations have a signature

RDFS relations have a signature

... the domain is the type of the resource the relation starts from.

... the range is the type of the resource the relation ends to.

RDFS relations with several...

... domains mean all domains apply.

... ranges mean all ranges apply.

Ranges and domains are optional

looks like object programming, **but**

properties

are first class citizens,

they are not defined inside classes,

they have their own hierarchy.

no overwriting

in particular a property can not be refined for sub classes of its range or domain.

multi-instantiation

a resource can have several types,

it can be the instance of several classes

like multiple lights, facets on a resource.

RDFS provides primitives to...

... give labels ...

... give comments ...

... for classes and properties

RDFS using the XML syntax for **RDF...**

```
<rdf:RDF xml:base="http://inria.fr/2005/humans.rdfs"
         xmlns:rdf ="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
         xmlns="http://www.w3.org/2000/01/rdf-schema#>
```

```
<Class rdf:ID="Man">
 <subClassOf rdf:resource="#Person"/>
 <subClassOf rdf:resource="#Male"/>
 <label xml:lang="en">man</label>
 <comment xml:lang="en">a male person</comment>
</Class>
```

```
<rdf:Property rdf:ID="hasMother">
 <subPropertyOf rdf:resource="#hasParent"/>
 <range rdf:resource="#Female"/>
 <domain rdf:resource="#Human"/>
 <label xml:lang="en">has for mother</label>
 <comment xml:lang="en">a female parent</comment>
</rdf:Property>
```

```
</rdf:RDF>
```

RDFS semantics : standard inference rules to derive additional triples from known statements.

IF (c_2, subClassOf, c_1)
AND (x, type, c_2)
THEN (x, type, c_1)

example of type propagation

IF (Man, subClassOf, Person)
AND (Tom, type, Man)
THEN (Tom, type, Person)

IF (p_2, subPropertyOf, p_1)
AND (x, p_2 , y)
THEN (x, p_1 , y)

example of property propagation

IF (author, subPropertyOf, creator)
AND (Tom, author, Report12)
THEN (Tom, creator, Report12)

IF (c_2, subClassOf, c_1)
AND (c_3, subClassOf, c_2)
THEN (c_3, subClassOf, c_1)

example of subClass transitivity

IF (Person, subClassOf, Animal)
AND (Man, subClassOf, Person)
THEN (Man, subClassOf, Animal)

IF (p_2, subPropertyOf, p_1)
AND (p_3, subPropertyOf, p_2)
THEN (p_3, subPropertyOf, p_1)

example of subProp transitivity

IF (parent, subPropertyOf, ancestor)
AND (father, subPropertyOf, parent)
THEN (father, subPropertyOf, ancestor)

IF (p_1, domain, c_1)
AND (x, p_1, y)
THEN (x, type, c_1)

example of domain inference

IF (author, domain, Human)
AND (Tom, author, Report12)
THEN (Tom, type, Human)

IF (p_1, range, c_1)
AND (x, p_1, y)
THEN (y, type, c_1)

example of range inference

IF (author, range, Document)
AND (Tom, author, Report12)
THEN (Report, type, Document)

summary

take-home message on RDF**S**

RDF Schema to...

... define classes and relations of resources and organize their hierarchy

... define signatures of relations (domain, range)

... document them with labels and comments

... define associated inference rules


fabien, gandon