Resource Description Languages

Dr. V.S. Felix Enigo, DCSE, SSNCE

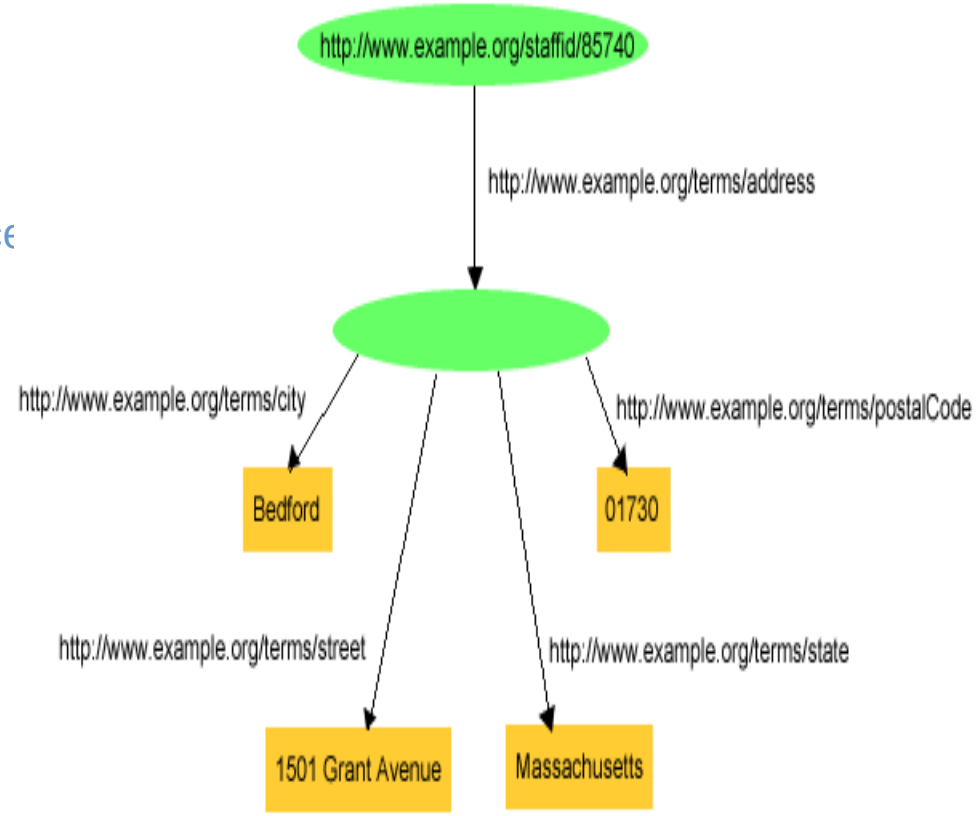
Concepts

- Resource Description Framework (RDF) - A data modelling language
- Triples are expressions, making statements about web resources
- Triples follow a *subject–predicate–object* structure
- *subject* denotes the resource
- *predicate* (also called a property) denotes a relationship between the *subject* and the *object*
- Ex. "The sky has the color blue" (subject - "the sky", predicate - "has the color", object denoting "blue")

Contd…

- An ***RDF triple*** contains three components:
- the ***subject***, is a URI reference or a blank node
- the ***predicate*** is an RDF URI reference
- the ***object*** is an RDF URI reference, unicode string literal or a blank node
- Blank node are anonymous resource resource (URI or literal is not given)
- Can be used as subject or object

- Blank Node Example



URI Reference

- URI that names a resource does not have to be dereferenceable at all

- **Examples of URI references**

- https://example.org/absolute/URI/with/absolute/path/to/resource.txt
- //example.org/scheme-relative/URI/with/absolute/path/to/resource.txt
- /relative/URI/with/absolute/path/to/resource.txt
- relative/path/to/resource.txt
- ../../../resource.txt
- ./resource.txt#frag01
- resource.txt
- #frag01

Contd…

- producers and consumers of RDF statements must agree on the semantics of resource identifier
- Example:
- http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#Merlot
- is intended by its owners to refer to the class of all Merlot red wines by vintner (i.e., instances of the above URI each represent the class of all wine produced by a single vintner)

Contd…

- RDF used widely in Semantic Web, Knowledge Management Applications
- Reason – simple data model, ability to model disparate, abstract concepts
- Collection of RDF statements represents a labeled, directed multi-graph
- Graphical structure suits for knowledge representation than other relational or ontological models
- RDFS, OWL – build upon RDF

RDF Example

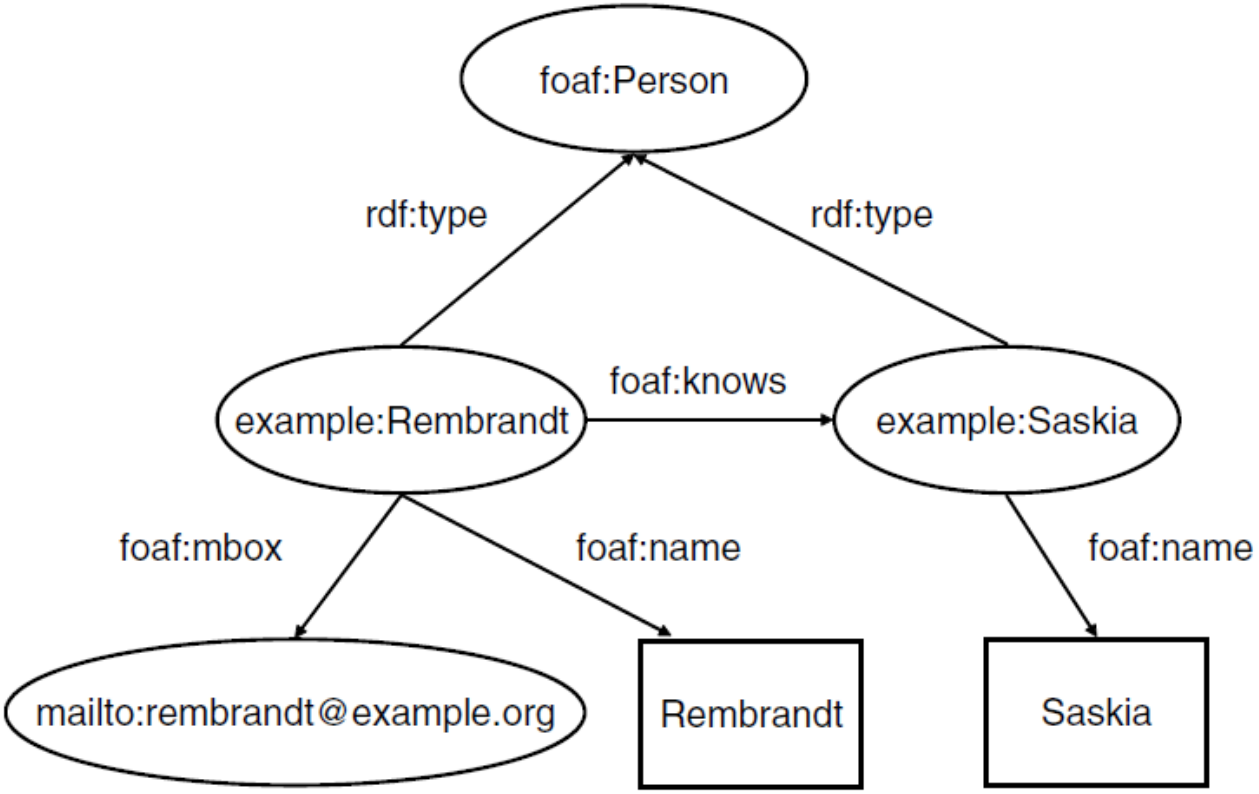
- RDF document describes a person named Rembrandt in turtle language

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#label> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix example: <http://www.example.org/> .

example:Rembrandt rdf:type foaf:Person .
example:Saskia rdf:type foaf:Person .
example:Rembrandt foaf:name "Rembrandt" .
example:Rembrandt foaf:mbox <mailto:rembrandt@example.org> .
example:Rembrandt foaf:knows example:Saskia .
example:Saskia foaf:name "Saskia"
```

Visualization

- six statements shown as a directed, labelled graph



RDFSchema

- Descriptive power of RDF is minimal
- RDF used with RDFS
- RDFS – extension of RDF – uses classes and subclasses
- Classes and properties connected by specifying the domain and range of properties

RDF Vocabulary

Basic constructs

- *rdf :type*
- *rdf :Property*
- *rdf :XMLLiteral*

Collections

- *rdf :List*
- *rdf :Seq*
- *rdf :Bag*
- *rdf :Alt*
- *rdf :first*
- *rdf :value*
- *rdf :rest*
- *rdf :nil*
- *rdf : n*

Reification

- *rdf :Statement*
- *rdf :subject*
- *rdf :predicate*
- *rdf :object*

RDFS Vocabulary

Basic constructs

- *rdfs:domain*
- *rdfs:range*
- *rdfs:Resource*
- *rdfs:Literal*
- *rdfs:Datatype*
- *rdfs:Class*
- *rdfs:subClassOf*
- *rdfs:subPropertyOf*

Collections

- *rdfs:member*
- *rdfs:Container*
- *rdfs:ContainerMembershipProperty*

Documentation & reference

- *rdfs:comment*
- *rdfs:seeAlso*
- *rdfs:isDefinedBy*
- *rdfs:label*

Example

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
foaf:Person rdf:type owl:Class .
foaf:Person rdfs:label "Person" .
foaf:Person rdfs:subClassOf foaf:Agent
foaf:Person owl:disjointWith foaf:Document .

foaf:knows rdf:type owl:ObjectProperty .
foaf:knows rdfs:label "knows" .
foaf:knows rdfs:domain foaf:Person .
foaf:knows rdfs:range foaf:Person .

foaf:name rdf:type owl:DatatypeProperty .
foaf:name rdfs:label "name" .
foaf:name rdfs:subPropertyOf rdfs:label .
foaf:name rdfs:domain owl:Thing .
foaf:name rdfs:range rdfs:Literal
```

Features of RDF

- same model of subject/predicate/object to describe instances and classes
- Not necessary to separate description of instances from description of classes not needed
- Ex. to describe a class using typical instances (weekday is a class of 5 days)
- RDF Flexibility - *rdfs:Class are not treated any different from user defined* classes
- *Ex. foaf :name property as a subproperty of the rdfs:label property.*
- domain specific labelling property is a subproperty of the more general *rdfs:label property*

Contd…

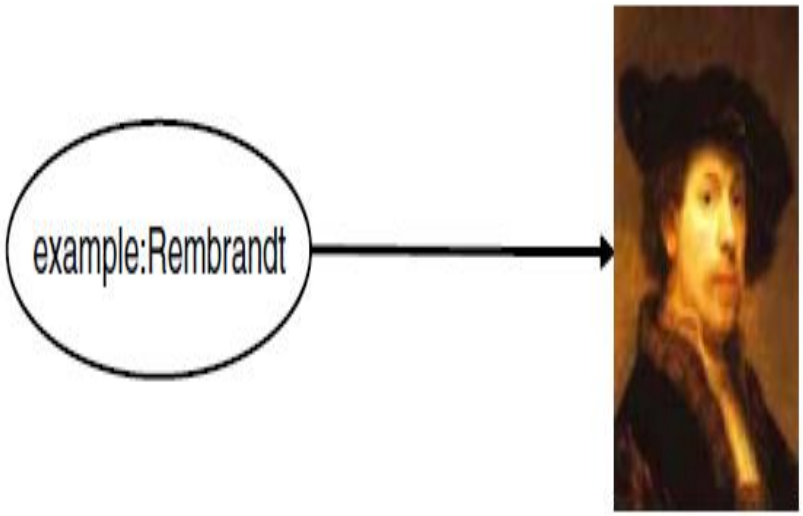
- RDF makes no clear separation between classes, instances and properties.
- Metamodeling - One can create classes of instances
- *Ex. rdf :type is used on both an instance (Rembrandt,* whose class is Person), and a class (Person, whose class is the class of all classes)
- *rdfs:label property is used on both instances, classes and properties*

Contd…

- RDF language constructs are not strictly interpreted
- Ex. even though the range of the *foaf :knows property is defined to be foaf :Person*
- *It can be added the statement* that *example:Rembrandt foaf :knows mailto:saskia@example.org*
- interpreted as a hint that the resource *mailto:saskia@example.org is both an email address and a Person.*

RDF and the notion of semantics

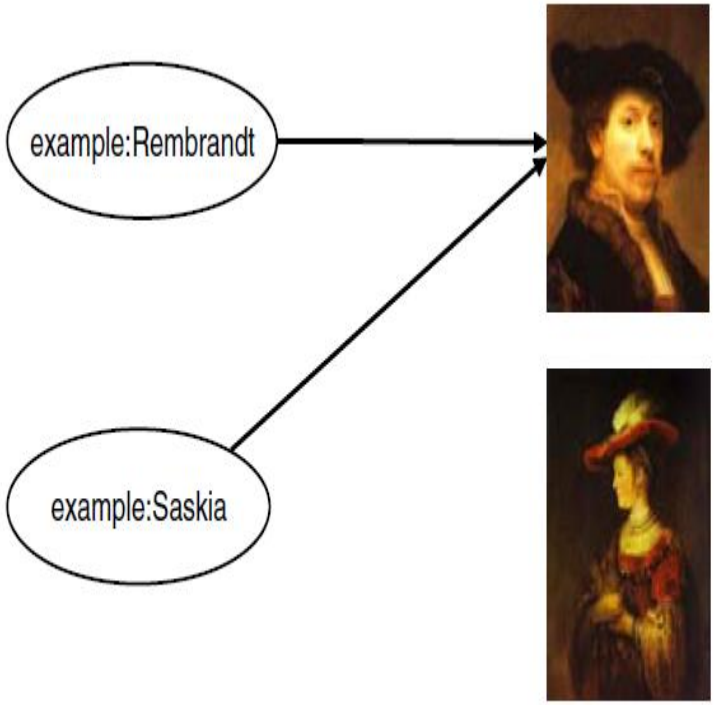
- Semantic added to RDF(S) constructs in a model-theoretic semantics way
- model-theoretic semantics - a mapping from one model to a metamodel where the truth of propositions is already uniquely determined (also called an *interpretation)*
- Instead of formal can be an ontology of independently existing reality
- Ex. mapping between symbols and the objects or relations they intended to describe



Example

- RDF are used to put constraints and exclude thereby some unintended interpretations
- To exclude interpretations where the two instances are mapped to the same object
- Ex. instance Rembrandt is *owl:differentFrom* *the second instance Saskia*

- An unintended model where symbols for different persons are mapped to the same object. Excluded using (*owl:differentFrom)*



RDFS – Axiomatic Triples

- RDF(S) provides axiomatic triples that are true in every RDF(S) interpretation

- Ex. the semantics of *rdfs:subPropertyOf is given by the following rules:*

$$(aaa, rdfs{:}subPropertyOf, bbb) \wedge (bbb, rdfs{:}subPropertyOf, ccc) \rightarrow (aaa, rdfs{:}subPropertyOf, ccc)$$

$$(aaa, rdfs{:}subPropertyOf, bbb) \rightarrow (aaa, rdf{:}type, rdf{:}Property)$$

$$(aaa, rdfs{:}subPropertyOf, bbb) \rightarrow (bbb, rdf{:}type, rdf{:}Property)$$

$$(xxx, aaa, yyy) \wedge (aaa, rdfs{:}subPropertyOf, bbb) \rightarrow (xxx, bbb, yyy)$$

Contd…

- RDF semantics is that interpretation of language is based on an open world assumption and is kept monotonic
- Monotonicity means additional knowledge added to an RDF knowledge base cannot make previous inferences invalid
- Example, if we specify that the range of the foaf:knows property is Person and then state that Rembrandt knows an instance another class such as Pluto the dog, do not cause a (logical) contradiction
- it is assumed that there could exist a statement defining that some other class (e.g. Dog) is also in the range of *foaf :knows.*