
RDF e oltre

[Oreste Signore](mailto:os@orestesignore.eu), <os@orestesignore.eu>

Summer School LDA

Libraries in the digital age: linked data technologies for a global knowledge sharing
Pula (Cagliari), 29 agosto - 1° settembre 2016

Presentazione: <http://www.orestesignore.eu/education/lda/slides/lda2.html>

Documento pdf: <http://www.orestesignore.eu/education/lda/slides/lda2.pdf>

Formato XHTML realizzato usando il tool Slidy di Dave Raggett.

Slidy dovrebbe funzionare in tutti i browser moderni con Javascript abilitato. Usare freccia destra/sinistra per muoversi da una slide all' altra.

Vedi [la pagina di aiuto di Slidy](#) per ulteriori informazioni.



Contenuto

- RDFS
- Ontologie
- OWL

RDFSchema

Perché RDF Schema?

- RDF è un linguaggio universale per descrivere le risorse usando il proprio vocabolario
- È possibile scrivere statement (triple s-p-o) RDF sintatticamente corretti, che possono essere sensati:

Leonardo	autoreDi	Gioconda
Cimabue	maestroDi	Giotto

o privi di senso

Michelangelo	autoreDi	Leonardo
--------------	----------	----------

Perché RDF Schema?

- Alcune cose essenziali per descrivere la nostra " *conoscenza addizionale*":
 - *definire le "cose" che vogliamo descrivere (istanze e classi)*
 - *definire le eventuali restrizioni (es. dominio e codominio)*
 - *relazioni addizionali (e sottoproprietà)*
- Questo è il ruolo di *RDF Schema*
 - *ufficialmente: "RDF Vocabulary Description Language"*
 - *il termine "Schema" sopravvive per ragioni storiche...*

Classi, Risorse, ...

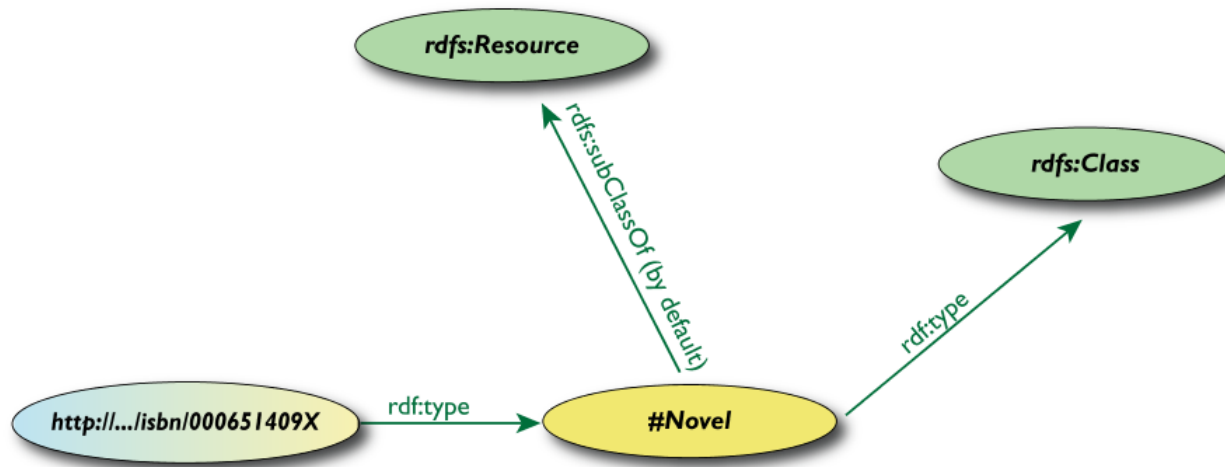
- Consideriamo un esempio classico delle ontologie:
 - prendiamo il termine "*cane*"
 - "*ogni cane è un mammifero*"
 - "*«Attila» è un cane*"
 - etc.
- RDFS definisce *risorse* e *classi*:
 - qualunque cosa in RDF è una "risorsa"
 - le "classi" sono risorse, ma...
 - ...sono anche collezione di possibili risorse (quindi "*individuals*")
 - "mammifero", "cane", ...



Classi, Risorse, ... (cont.)

- Le relazioni sono definite tra classi/risorse:
 - *"typing"*: un individuo appartiene a una specifica classe ("«Attila» è un cane")
 - per essere più precisi: "`<anag:96RCI> is-an-instance-of cane`"
 - *"subclassing"*: le istanze di una classe sono anche istanze dell' altra ("ogni cane è un mammifero")
- *RDFS formalizza queste relazioni in RDF*

Classi, Risorse in RDF(S)



- RDFS definisce `rdfs:Resource`, `rdfs:Class` come nodi; `rdf:type`, `rdfs:subClassOf` come proprietà
 - (sono tutti URI speciali, usiamo il namespace per semplicità)
- `rdfs:Class` è la “*classe di tutte le classi*” cioè qualunque classe è un' istanza di `rdfs:Class`
- una risorsa *può appartenere a più classi*

Un esempio di RDFS in RDF/XML

- La parte dello schema:

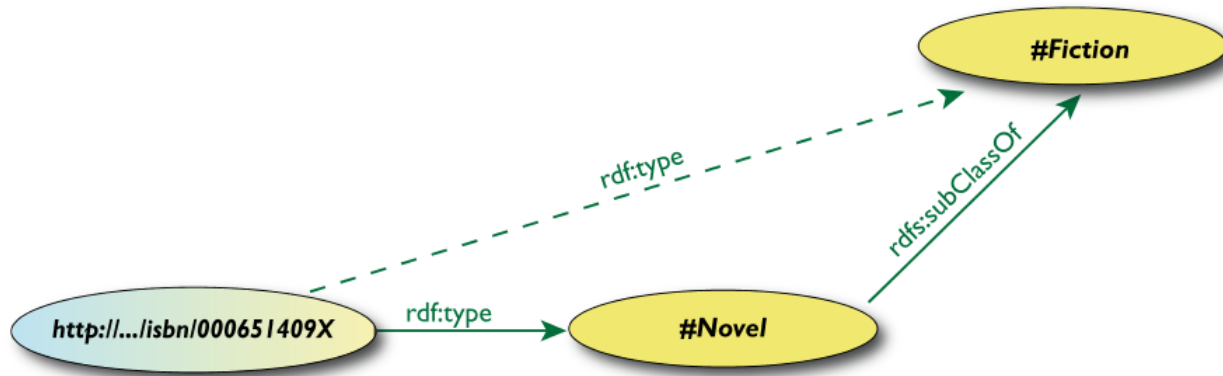
```
<rdf:Description rdf:ID="Novel">  
  <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>  
</rdf:Description>
```

- I dati RDF per una specifica istanza:

```
<rdf:Description rdf:about="http://.../isbn/000651409X">  
  <rdf:type rdf:resource="http://.../bookSchema.rdf#Novel"/>  
</rdf:Description>
```

- Nel KR tradizionale questi due componenti sono spesso identificati come: "*Terminological axioms*" and "*Assertions*" (**T-box** e **A-box**)

Inferenza delle proprietà



- `<http://.../isbn/000651409X rdf:type #Fiction>`
- *non* è nei dati RDF originali...
- ...ma può essere *dedotta* dalle regole RDFS
- I migliori ambienti RDF restituiscono anche questa tripla

Proprietà

- Property è una classe speciale: **rdf:Property**, che è la “*classe di tutte le proprietà*”
 - *le proprietà sono anch' esse risorse, identificate da URI*
- La proprietà hanno dei *vincoli* di *dominio* (*domain*) e *codominio* (*range*)
 - *quali elementi (individual) possono comparire come object o subject*
- È anche possibile definire delle “**sub-property**”
 - *tutte le risorse connesse da una sub-property sono anche connesse dalla property*

Proprietà (cont.)

- *Le proprietà sono anch' esse risorse, identificate da URI...*
 - Quindi le *proprietà di proprietà* possono essere espresse come... *RDF properties*
 - *è una cosa un po' sofisticata, ma viene usata spesso*
 - Per esempio, (**P** **rdfs:range** **C**) significa che:
 1. **P** è una proprietà, cioè un' istanza della classe **rdf:Property**
 2. **C** è una classe, cioè un' istanza della classe **rdfs:Class**
 3. le risorse denotate dagli oggetti delle triple che hanno come predicato **P** sono istanze della classe **C**
- questo è uno statement RDF con subject **P**, object **C**, e property **rdfs:range**

Anche un po' di RDFS può dare enormi vantaggi...

- Ritornando all' esempio iniziale del merge
- potremmo aver usato:
 - **f:auteur** *subproperty-of* **a:author** e viceversa
- (ma vedremo che c'è un altro modo per farlo...)
- In alcuni casi, è necessario gestire conoscenza più sofisticata (vedi in seguito...)

Alcune Classi e proprietà predefinite

- RDF(S) ha alcune classi e proprietà predefinite
- Non sono "concetti" nuovi nel **RDF Model**, ma solo risorse con una semantica definita
- Esempi:
 - *collections (o liste)*
 - *containers: sequence, bag, alternatives*
 - *reification*
 - **rdfs:comment, rdf:seeAlso, rdf:value**

Vocabolari

- Grazie a RDFS è possibile definire *vocabolari*
 - *collezioni di proprietà e classi*
 - *relazioni tra di loro e con [termini di altri vocabolari](#)*
- Qualche esempio:
 - *termini di Dublin Core: creator, date, ...*
 - *termini di FOAF: caratterizzazione delle persone*
 - *Creative Commons: copyright classes, license relations, ...*
 - *termini di schema.org: events, organizations, places, reviews, ...*

Ontologie (OWL)

Ontologie

- RDFS è utile, ma non è in grado di soddisfare tutti i requisiti
- Applicazioni complesse hanno ulteriori necessità:
 - *un programma può **ragionare** su alcuni termini? Per es.:*
 - "if «Person» resources «A» e «B» hanno la stessa «foaf:email» property, then «A» e «B» sono identici"
 - *se qualcun altro definisce un insieme di termini, possiamo sapere se sono gli stessi già noti?*
 - ***costruire** classi, non limitarsi a dar loro un nome*
 - *restringere il codominio (range) di una proprietà **quando usata per una specifica classe***
 - *classi disgiunte o equivalenti*
 - *etc.*

Ontologie (cont.)

- Occorre il supporto di *ontologie* nel Semantic Web:

"definisce i concetti e le relazioni usate per descrivere e rappresentare un dominio di conoscenza"

- Occorre un *Web Ontology Languages*
 - *RDFS* può essere considerato un *Web Ontology Language* semplice
 - *OWL* offre un insieme più ampio e complesso di possibilità
- I linguaggi devono essere un compromesso tra:
 - *semantica ricca per applicazioni complesse*
 - *fattibilità, implementabilità*

Cosa è un'ontologia? (1)

Neches et al. (1991)

*An ontology defines the basic terms and relations comprising the **vocabulary** of a topic area as well as the **rules** for combining terms and relations to define **extensions** to the vocabulary.*

Gruber (1993)

An ontology is an explicit representation of a conceptualization

Borst (1997)

Ontologies are defined as a formal specification of a shared conceptualization

Studer et al. (1998) (Merging and explaining Gruber and Borst)

*An ontology is a formal, explicit specification of a shared conceptualisation. A '**conceptualisation**' refers to an abstract model of some phenomenon in the world by having identified the relevant concepts of that phenomenon. '**Explicit**' means that the type of concepts used, and the constraints on their use are explicitly defined. For example, in medical domains, the concepts are diseases and symptoms, the relations between them are causal and a constraint is that a disease cannot cause itself. '**Formal**' refers to the fact that the ontology should be machine readable, which excludes natural language. '**Shared**' reflects the notion that an ontology captures consensual knowledge, that is, it is not private to some individual, but accepted by a group.*

Cosa è un'ontologia? (2)

Guarino

A logical theory which gives an explicit, partial account of a conceptualization

*A set of **logical axioms** designed to account for the **intended meaning** of a vocabulary.*

*A **specific artifact** designed with the purpose of **expressing the intended meaning** of a vocabulary*

Jim Hendler

*A set of **knowledge terms**, including the vocabulary, the **semantic interconnections** and some **simple rules** of inference and logic for some particular topic*

Disaccordo o accordo?

- Definizioni *diverse*, ma *accordo* sul concetto
- Un' ontologia include non solo i termini che sono *esplicitamente* definiti in essa, ma anche la *conoscenza* che ne può essere *derivata* mediante un processo di *inferenza*
- Un' ontologia cattura *conoscenza consensuale*
- "*Little semantics goes a long way*" (Jim Hendler)

Classificazione delle ontologie

Lightweight

- essenzialmente *tassonomie*
- includono *concetti*, tassonomie dei concetti, *relazioni* tra concetti, *proprietà* che descrivono concetti

Heavyweight

- forniscono *restrizioni* addizionali sulla semantica del dominio
- aggiungono *assiomi* e *vincoli* alle ontologie lightweight
- assiomi e vincoli chiariscono il *significato inteso (intended meaning)* dei termini inclusi nell'ontologia

Ontologie: tecniche di modellazione della conoscenza

Molto informali

espresse in *linguaggio naturale* (quindi non machine readable/understandable)

Semi-informali

forma ristretta e *strutturata* del linguaggio naturale

Semi-formali

linguaggio artificiale e *definito formalmente* (es. OWL)

Rigorosamente formali

termini definiti meticolosamente con *semantica formale*, *teoremi* e *prove di proprietà* quali solidità e completezza

Livelli di rappresentazione della conoscenza

- Il *grado di formalizzazione* dei concetti e delle loro relazioni varia in modo considerevole tra i vari domini di conoscenza
- *Lower end*
 - lessici e tassonomie semplici (sistemi ordinati di classificazione in cui i termini sono in relazione gerarchica)
 - esempio: [Iconclass](#)
- *Middle level*
 - thesauri (vocabolari controllati strutturati in modo da esplicitare le relazioni tra termini e concetti, spesso utilizzati per reperire le informazioni da un database o da una base dati documentaria)
 - esempio: Art & Architecture Thesaurus ([AAT](#))
- *High end*
 - teorie logiche assiomatizzate, che includono regole per assicurare la correttezza e la validità logica di proposizioni (statement) espresse nella lingua della specifica disciplina
 - esempio: CIDOC object-oriented Conceptual Reference Model ([CIDOC CRM](#))

Cosa è meglio?

- Le ontologie semi-formali si sono dimostrate efficaci per l'*information integration*
- Le ontologie semi-formali richiedono uno sforzo di sviluppo *significantly smaller* e sono più diffuse e più utili delle ontologie formali
- Le ontologie semi-formali possono gestire informazione *parziale* (incompleta) e *potenzialmente incoerente* (inconsistent), in particolare nelle asserzioni dell'ontologia
- The [GoPubMed](#) example
- "*Little semantics goes a long way*" (Jim Hendler)

Il Semantic Web: la visione del W3C

Ipotesi e sfide

- il Web è intrinsecamente *distribuito*
- le macchine possono accedere ad un *insieme strutturato di informazioni* e a un insieme di *regole di inferenza* da utilizzare per il *ragionamento automatico*
- per ragionare sui dati occorrono le *ontologie*
- fornire un *linguaggio* per esprimere dati e regole per ragionare sui dati
- *esportare* sul web delle regole da qualunque sistema di rappresentazione della conoscenza

Metadati

- informazioni, *comprensibili dalla macchina*, relative a una risorsa web o a qualche altra cosa

Ontologie sul Web

- Le ontologie sono *sul Web*. Quindi:
 - *le applicazioni possono utilizzare varie ontologie differenti, oppure*
 - *... le stesse ontologie, ma espresse in lingue diverse*
 - *le equivalenze tra termini, e le relazioni intercorrenti tra di loro, possono diventare un problema non banale*

Il Semantic Web: le tecnologie W3C

Resource Description Framework (RDF)

- strumento base per la *codifica*, lo *scambio* e il *riutilizzo* di metadati strutturati
- consente l'*interoperabilità* tra applicazioni che si scambiano sul web informazioni *machine-understandable*

Web Ontology Language (OWL)

- linguaggio per *descrivere* proprietà, vincoli, cardinalità, etc.
- permette di *esportare ontologie* in modo interoperabile

OWL: tre sottolinguaggi

OWL Lite

per rappresentare *classificazioni gerarchiche* e *vincoli semplici*. Permette una migrazione rapida per thesauri e altre tassonomie.

OWL DL

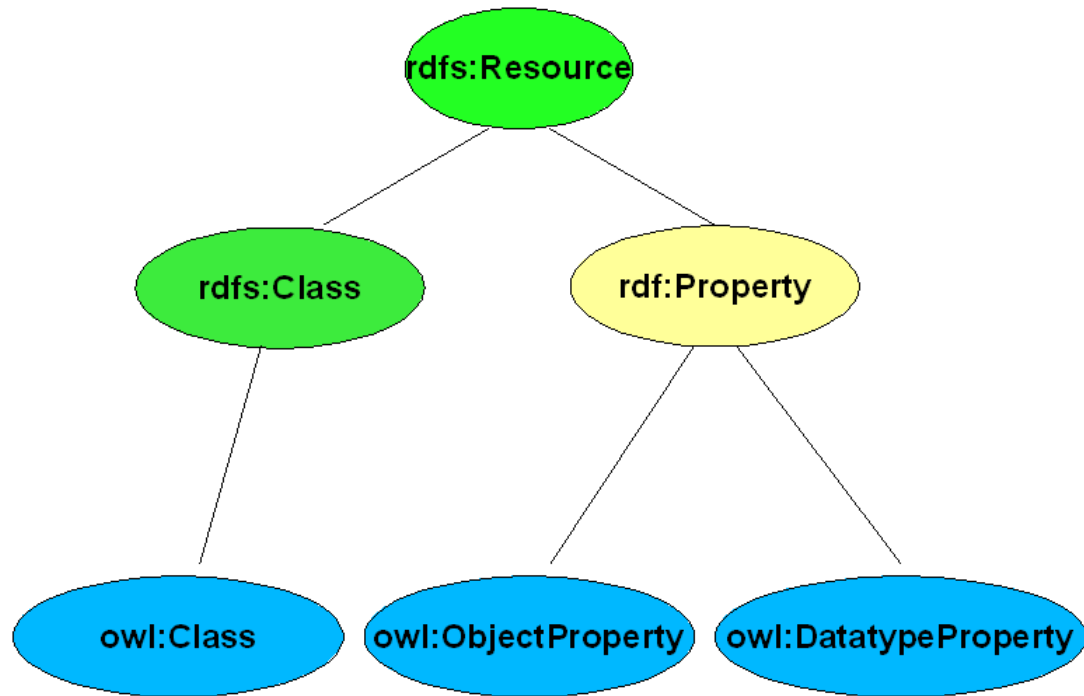
per utenti che desiderano la *massima potenza espressiva* garantendo comunque la *completezza computazionale* (tutte le conclusioni sono computabili) e la *decidibilità* (tutte le computazioni terminano in un tempo finito)

OWL Full

per utenti che desiderano la *massima potenza espressiva* e la libertà sintattica di RDF.

Non fornisce garanzie sui tempi di computazione, e difficilmente sarà supportato nella sua interezza da software che implementano il ragionamento

Relazioni di sottoclasse tra alcune primitive RDF/RDFS e OWL



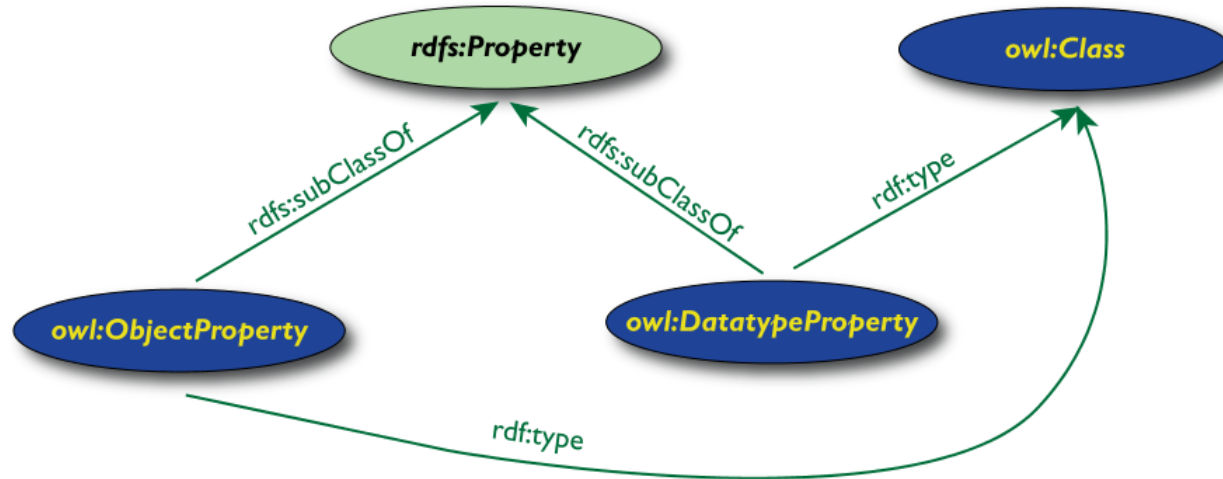
Le classi in OWL

- In RDFS, è possibile unicamente definire sottoclassi di classi esistenti
- In OWL, si possono *costruire* classi a partire da quelle esistenti:
 - *enumerazione*
 - *intersezione, unione, complemento*
 - *mediante restrizioni delle proprietà (valori ammissibili, cardinalità, simmetria, transitività, dipendenza funzionale)*
- Per questo, OWL introduce i suoi elementi **Class** e **Thing** per distinguere le *classi* dalle *istanze*



Caratterizzazione delle proprietà

- In OWL è possibile caratterizzare il *comportamento* delle proprietà (simmetrica, transitiva, dipendenza funzionale, dipendenza funzionale inversa,...)
- OWL mantiene separate le proprietà dei dati
 - "datatype property" significa che il codominio è una costante *tipata*



Object e Datatype Property

owl:ObjectProperty

- consentono di mettere in relazione tra di loro gli oggetti (es.: possiede, insegna, etc.)

owl:DatatypeProperty

- consentono di mettere in relazione gli oggetti con i valori (es. numeroDiTelefono, nome, dataDiNascita, etc.)
- non esistono tipi predefiniti
- è possibile utilizzare i data type di XML Schema, restando così aderenti all' architettura a strati del Semantic Web

Caratterizzazione delle proprietà

`owl:minCardinality`

`owl:maxCardinality`

`owl:SymmetricProperty`

`owl:TransitiveProperty`

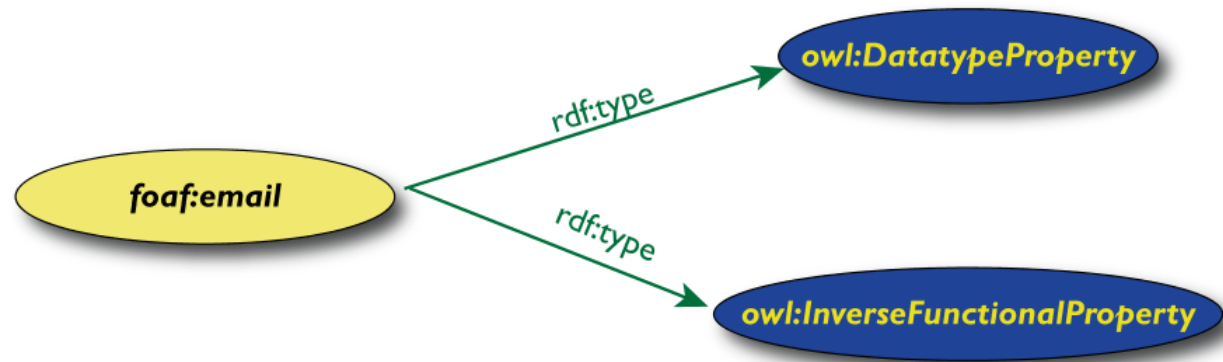
`owl:FunctionalProperty`

- ha al massimo un valore per ogni oggetto (es. dataDiNascita, peso, nome)

`owl:InverseFunctionalProperty`

- proprietà per cui due oggetti diversi non possono avere lo stesso valore (es. èCodiceFiscaleDi, èNumeroDiMatricolaDi, etc.)

Un esempio di caratterizzazione delle proprietà

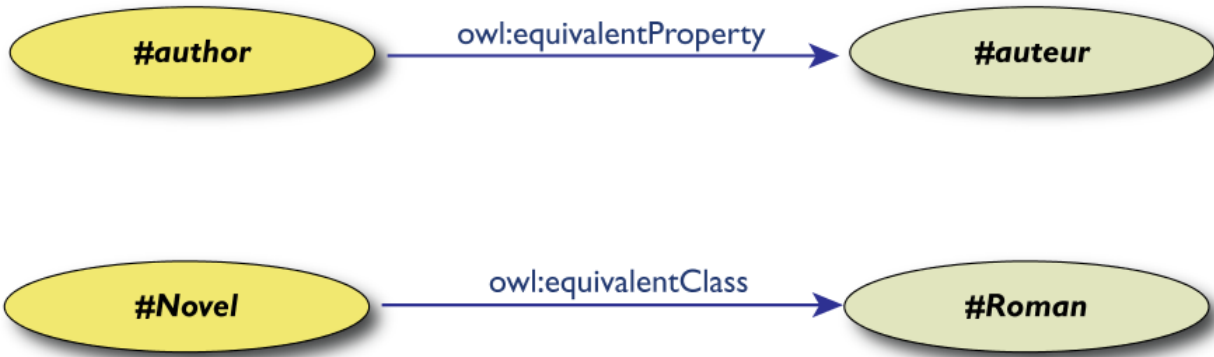


- "foaf:email" è **inverse functional**

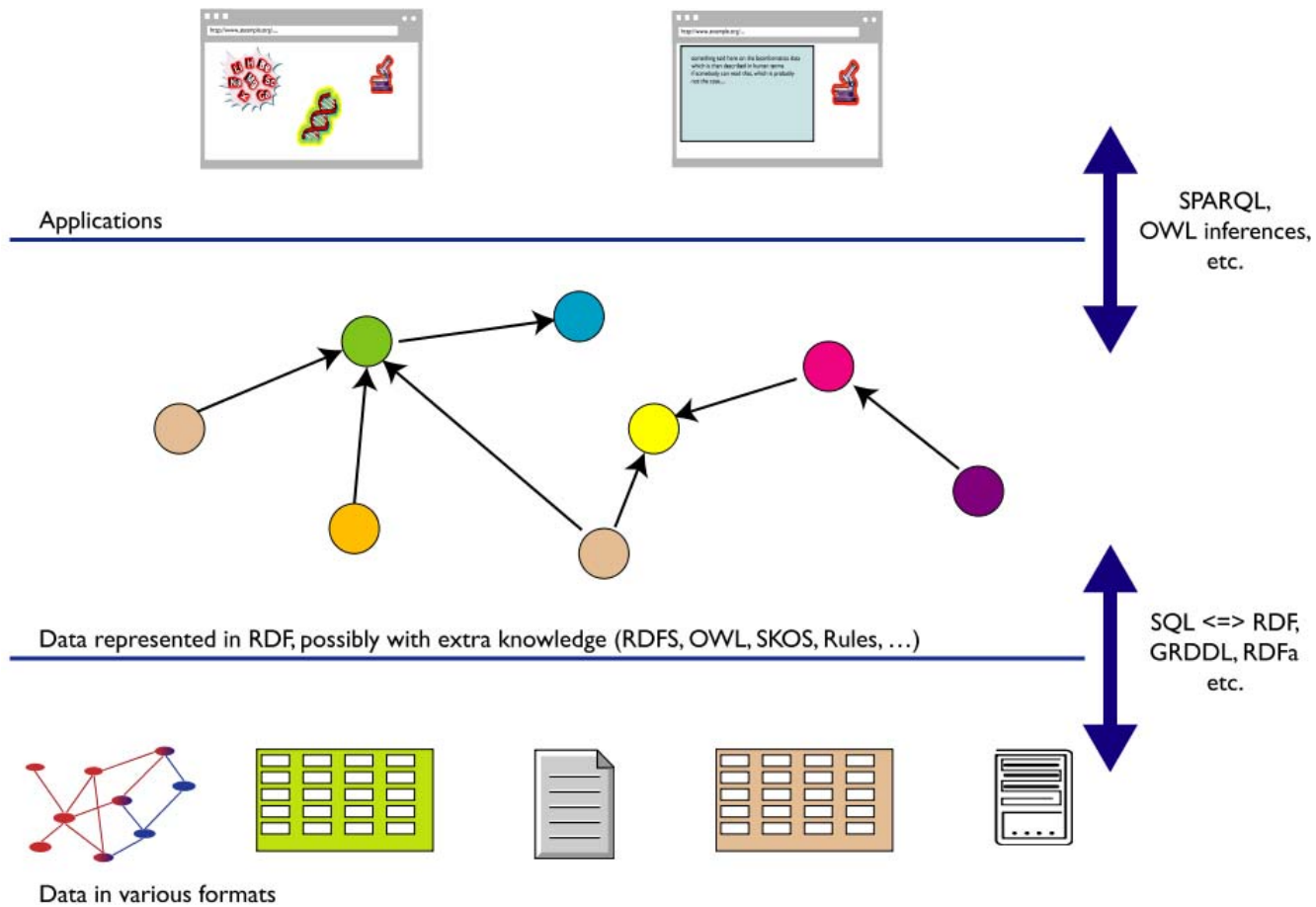
Equivalenza e Relazioni tra termini

- Per le classi:
 - **owl:equivalentClass**: *due classi sono costituite dagli stessi elementi*
 - **owl:disjointWith**: *non esistono istanze in comune*
- Per le proprietà:
 - **owl:equivalentProperty**
 - abbiamo visto il caso di **a:author** e **f:auteur**
 - **owl:inverseOf**: *relazione inversa*
- Per le istanze:
 - **owl:sameAs**: *due URI fanno riferimento alla stessa istanza (per es. un concetto)*
 - **owl:differentFrom**: *negazione di owl:sameAs*

Esempio: equivalenza inglese-francese



L' integrazione dei dati: il ruolo delle tecnologie



Qualche ontologia di riferimento

CIDOC CRM

The *CIDOC Conceptual Reference Model (CRM)* provides definitions and a *formal structure* for describing the implicit and explicit concepts and relationships used in cultural heritage documentation. The CIDOC CRM is intended to promote a *shared understanding* of cultural heritage information by providing a *common and extensible semantic framework* that any cultural heritage information can be mapped to. It is intended to be a common language for domain experts and implementers to formulate requirements for information systems and to serve as a guide for good practice of conceptual modelling. In this way, it can provide the “*semantic glue*” needed to mediate between different sources of cultural heritage information, such as that published by museums, libraries and archives.

<http://www.cidoc-crm.org/>

FRBR

Functional Requirements for Bibliographic Records

A *conceptual entity-relationship model* developed by the International Federation of Library Associations and Institutions (IFLA) that relates user tasks of retrieval and access in online library catalogues and bibliographic databases from a user's perspective. It represents a more holistic approach to retrieval and access as the relationships between the entities provide links to navigate through the hierarchy of relationships. The model is significant because it is *separate from specific cataloguing standards* such as AACR2 or International Standard Bibliographic Description (ISBD).

<http://www.ifla.org/node/2016>

EDM

The *Europeana Data Model* for Cultural Heritage

A vast number of Europe's cultural heritage objects are digitised by a wide range of data providers from the library, museum, archive and audio-visual sectors, and they all use *different metadata standards*. This data needs to appear in a meaningful way in a *cross-cultural, multilingual context* such as Europeana. Numerous cultural heritage resources such as thesauri exist worldwide and have the potential to add valuable content at low cost when re-used. Duplication of effort, however, needs to be avoided. The Linked Open Data environment lacks authoritative data from the cultural heritage community to contribute to the *development of new knowledge*.

The Europeana Data Model (EDM) aims to bridge these gaps in the Europeana context.

<http://pro.europeana.eu/edm-documentation>

Conclusioni

- La rappresentazione della conoscenza è essenziale per una elaborazione automatica delle informazioni esistenti sul web
- Le *tecnologie del Semantic Web* (RDF, RDFS, OWL) consentono di *rappresentare*, *esportare* e *condividere* la conoscenza in maniera interoperabile

Grazie per l' attenzione

Domande?

Se non è sul Web non esiste ...

... troverete sul sito orestesignore.eu (<http://www.orestesignore.eu>), sezione education

le *slide* (<http://www.orestesignore.eu/education/lda/slides/lda2.html>)

Consultate anche il *documento*: <http://www.w3c.it/papers/wsb08.pdf>