

# RDF Schema

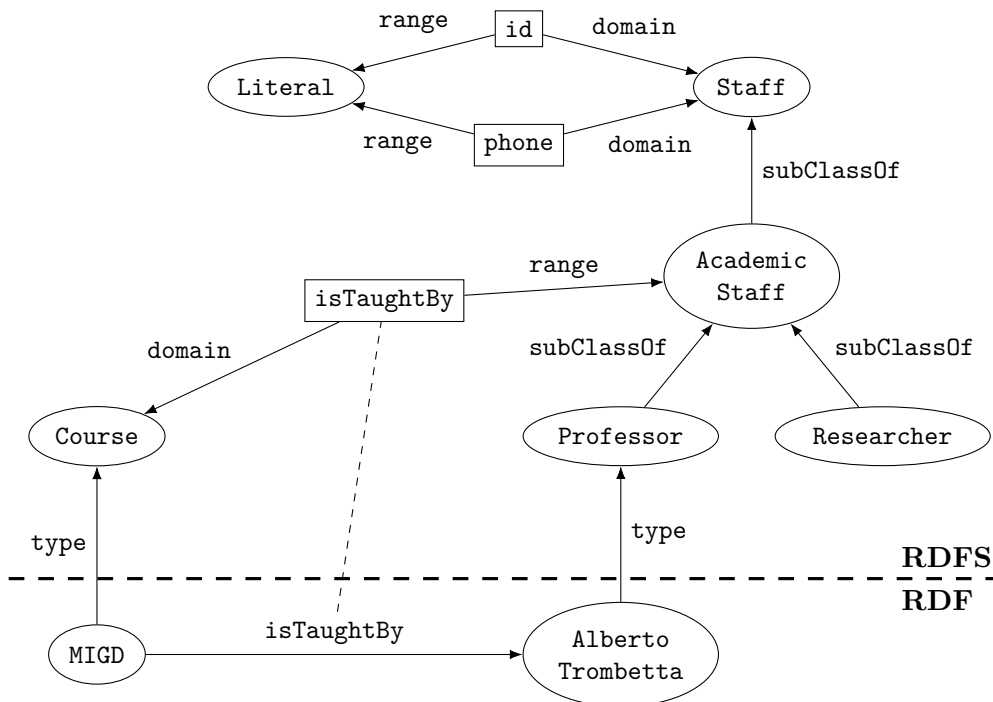
## 1 RDF Schema

Il modello RDF definisce un modo standard per rappresentare, in generale, le relazioni che possono esistere tra entità di qualunque tipo, ma non stabilisce il significato di tali entità e relazioni, cioè rimane del tutto indipendente dal dominio.

**RDF Schema (RDFS)** fissa il significato di alcune risorse (“classe”, ecc.) e proprietà/predicati (“appartenenza” di un’istanza a una classe, “inclusione” di una classe in un’altra, ecc.) RDF, fornendo così un modo standard di esprimere relazioni tra insiemi / **classi** di entità, e quindi di descrivere il significato delle entità e relazioni di un dominio.

## 2 Esempio

Un esempio di utilizzo di RDF/RDFS è il seguente grafo:



In questo grafo compaiono alcune risorse e proprietà definite da RDFS:

- Il predicato `type` denota la relazione di appartenenza di un'istanza ad esempio `MIGD` a una classe (corrispondente alla relazione  $\in$  tra insiemi).
- `subClassOf` denota la relazione di inclusione / sottoinsieme ( $\subseteq$ ).
- `domain` (dominio) indica la classe dei soggetti ai quali può essere applicato un predicato, mentre `range` (codominio) indica i possibili oggetti. Ad esempio, gli statement con il predicato `isTaughtBy` (“insegnato da”) devono avere un `Course` come soggetto, e un `Professor` come oggetto.

*Nota:* La descrizione di un ambito / contesto attraverso l'uso di un vocabolario condiviso (come in questo esempio, dove il vocabolario condiviso è quello di RDFS) è chiamata **ontologia**.<sup>1</sup>

### 3 Principali classi e proprietà

Alcune delle classi principali (“core”) definite da RDFS sono:

- `rdfs:Resource`: la classe di tutte le risorse;
- `rdfs:Class`: la classe di tutte le classi;
- `rdfs:Literal`: la classe di tutti i letterali (stringhe);
- `rdf:Property`:<sup>2</sup> la classe di tutte le proprietà.

Invece, le proprietà principali (“core”) fornite da RDFS sono:

- `rdf:type`:<sup>2</sup> associa una risorsa / istanza alla sua classe;
- `rdfs:subClassOf`: associa una classe a una sua superclasse;
- `rdfs:subPropertyOf`: associa una proprietà a una sua superproprietà;
- `rdfs:domain`: specifica il dominio di una proprietà;
- `rdfs:range`: specifica il codominio di una proprietà.

*Nota:* Le proprietà `rdfs:subClassOf` e `rdfs:subPropertyOf` sono transitive.

Un esempio di uso di alcune di queste classi e proprietà, rappresentato nella serializzazione RDF/XML, è il seguente, nel quale si descrive la classe `Professor`, indicando che è una sottoclasse di `AcademicStaff` (come nel grafo raffigurato in precedenza):

---

<sup>1</sup>Tale significato del termine “ontologia” è diverso da quello che si intende in ambito filosofico.

<sup>2</sup>Alcune di queste classi e proprietà sono in realtà definite da RDF: esso ha già un minimo di funzionalità di rappresentazione dei tipi delle risorse, che RDFS riprende e amplia.

```
<rdfs:Class rdf:about="Professor">
  <rdfs:subClassOf rdf:resource="AcademicStaff"/>
</rdfs:Class>
```

## 4 Regole di inferenza

In RDFS valgono alcune regole, chiamate formalmente **regole di inferenza**, che permettono di “derivare” (*inferire*) nuove triple (statement) a partire da un insieme di triple date. Ad esempio:

- Se  $u$  è una sottoclasse di  $v$ , e  $v$  è una sottoclasse di  $w$ , allora  $u$  è una sottoclasse di  $w$  (cioè, come già detto, la proprietà `subClassOf` è transitiva).
- Se  $i$  è un’istanza di  $v$ , e  $v$  è sottoclasse di  $w$ , allora  $i$  è un’istanza di  $w$ .

Le nuove triple ricavate in questo modo, anche se magari non vengono “materializzate” nel database, possono essere sfruttate nelle interrogazioni: così, ad esempio, un’interrogazione che richiede tutte le istanze di una classe otterrà (correttamente) anche le istanze delle sottoclassi.

Siccome definisce queste regole di inferenza, RDFS può essere inteso come una realizzazione di un “frammento” della logica del primo ordine.