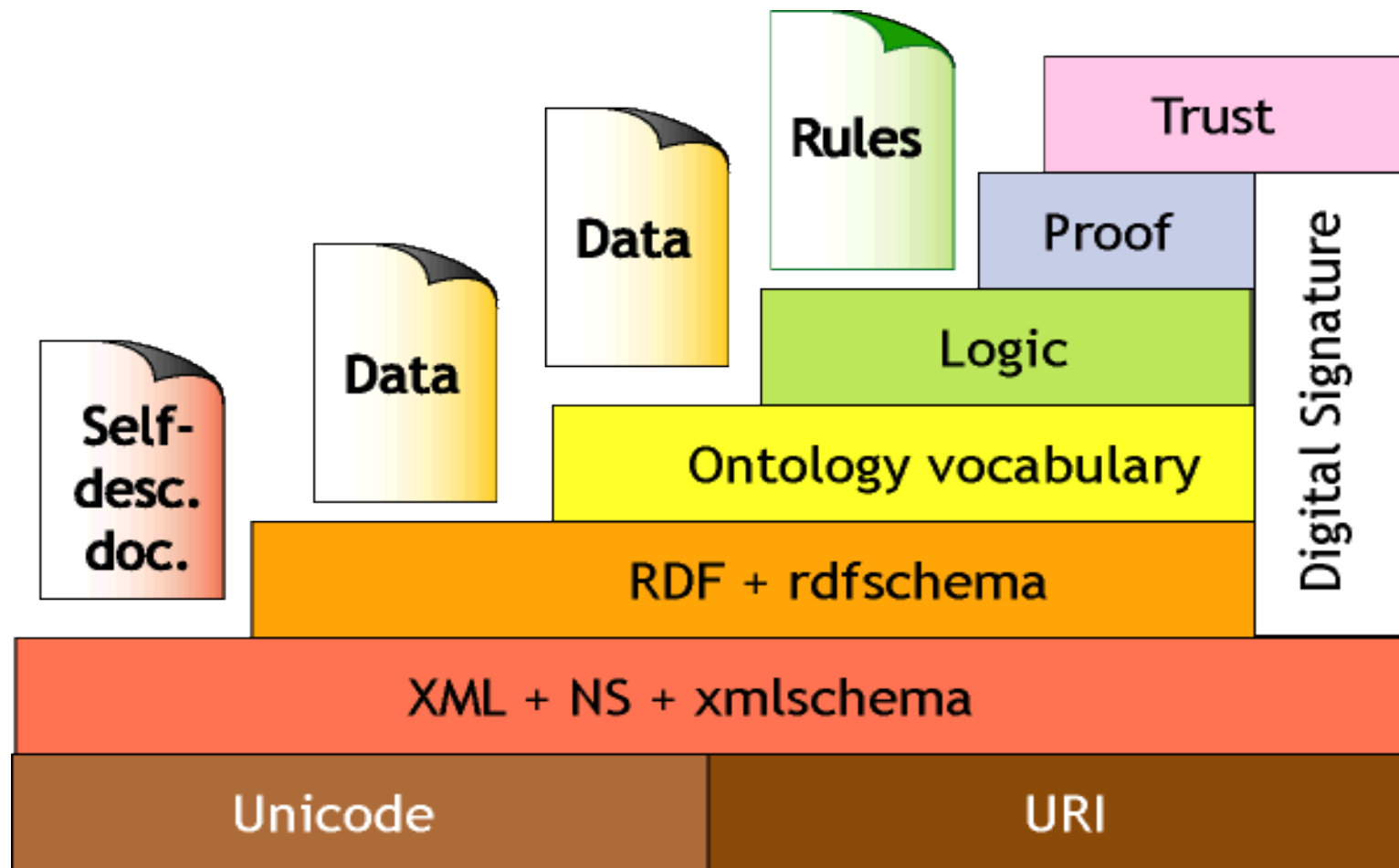


# RDF

Iztok Sarnik

# Stolp semantičnega spleta



# Sintaksa in semantika

- **Sintaksa**: struktura podatkov
- **Semantika**: pomen podatkov
- Dva pogoja potrebna za **skupno delo**:
  - Skupna sintaksa: programi lahko razčlenjujejo podatke
  - Skupen način razumevanja pomena: programi lahko uporabljajo podatke.

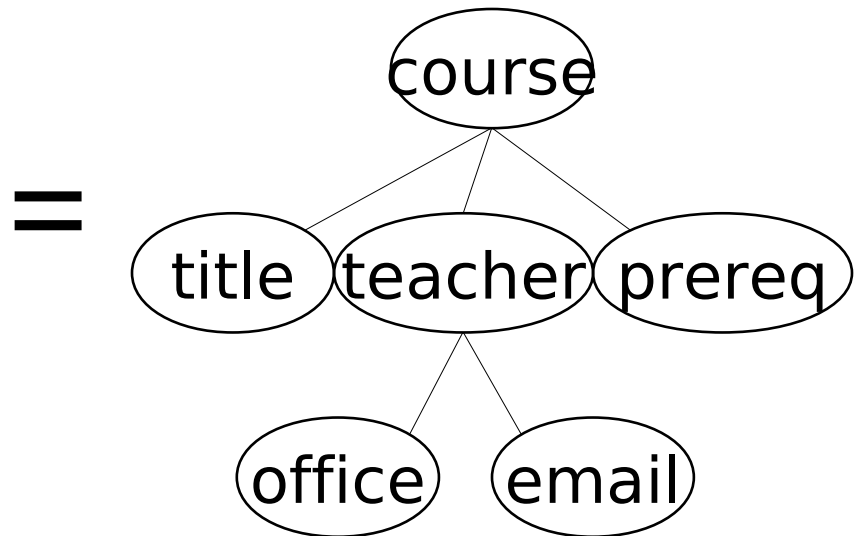
# XML

- XML: eXtensible Mark-up Language
- XML dokumenti so napisani z uporabniško definiranimi značkami
- Oznake so uporabljene za izražanje “pomena” delov podatkov

# XML

- XML: dokument = označeno drevo
- vozlišče = oznaka + atributi/vrednosti + vsebina

```
<course date="...">  
  <title>...</title>  
  <teacher>  
    <office>...</office>  
    <email>...</email>  
  </teacher>  
  <room>...</room>  
  <prereq>...</prereq>  
</course>
```



# XML

- XML Schema = slovnica za opis dreves in podatkovnih tipov
- Lahko uporabljamo XML za predstavitev semantike?

# XML in pomen

<Predator>

...

</Predator>

- Predator: srednje višine, velika vztrajnost, letalna naprava.
- Predator: tisti, ki terorizira, razbija in uničuje še posebno za neko korist.
- Predator: organizem, ki živi na osnovi ulova drugih organizmov.
- ...

# Omejitve pri opisu pomena

- XML ne opisuje:
  1. Besednjak specifičen za neko domeno
  2. Ontološki primitivi za modeliranje podatkov
- Zahteva dogovor glede 1 in 2
- Uporabna rešitev za lokalne projekte:
  - Agenti v manjšem stabilnem okolju
  - Strani na manjšem in stabilnem intranet
- Ni primerno za predstavitev spletnih virov



# RDF

- RDF je **podatkovni model**
  - Model je neodvisen od domene, aplikacije in se lahko internacionalizira
  - Model lahko vidimo kot usmerjen, označen graf ali kot objekten model (objekti/atributi/vrednosti)
- RDF podatkovni model je abstrakten, konceptualni nivo **neodvisen** od XML
  - XML je lahko sintaksa za RDF in ne del RDF
  - RDF podatki se lahko sploh ne pojavijo v XML obliki

# RDF model

RDF model = množica RDF **trojic**

trojica = izraz (stavek)

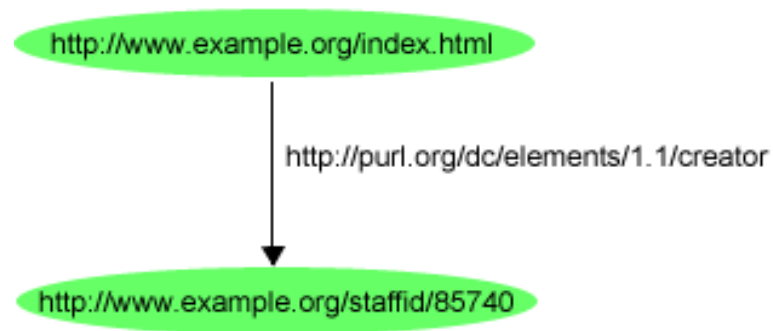
(subjekt, predikiat, objekt)

- subjekt = vir
- predikat = lastnost (vira)
- objekt = vrednost (lastnosti)



# RDF model

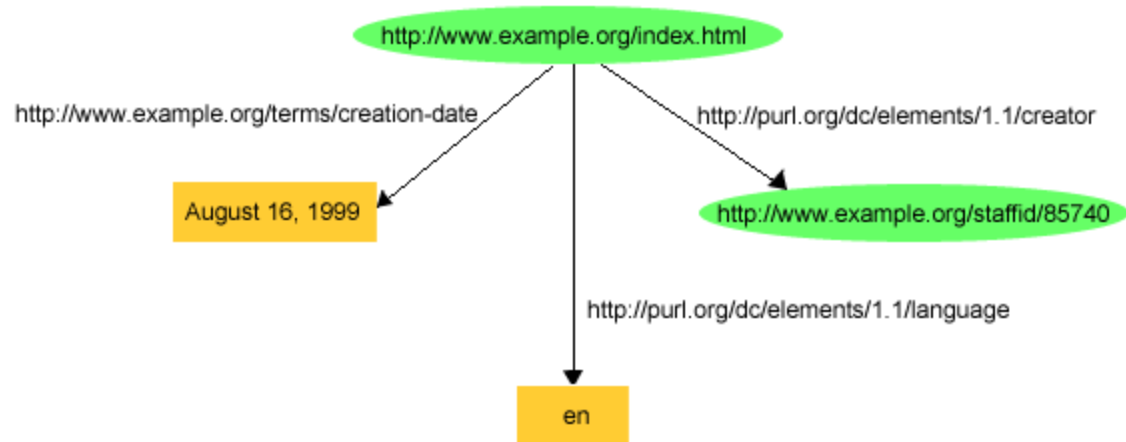
- Subjekt: `http://www.example.org/index.html`
- Predikat: `http://purl.org/dc/elements/1.1/creator`
- Objekt: `http://www.example.org/staffid/85740`
  
- Vsaka trojica ustreza eni povezavi v grafu



# Primer RDF opisa

```
<http://www.example.org/index.html> <http://purl.org/dc/elements/1.1/creator> <http://www.example.org/staffid/85740> .  
<http://www.example.org/index.html> <http://www.example.org/terms/creation-date> "August 16, 1999" .  
<http://www.example.org/index.html> <http://purl.org/dc/elements/1.1/language> "en" .
```

- Uporabljamo **polne poti** !
- **Koncepti** so ovali
- **Literali** so pravokotniki



# RDF sintaksa

RDF model = graf z označenimi povezavami  
= množica trojic

- Grafična notacija (graf)
- Notacija (neformalna) na osnovi trojic, npr.:  
(subject, predicate, object)
- Formalna sintaksa:
  - N3 notacija
  - Konkretna sintaksa: RDF/XML

# Prostori imen

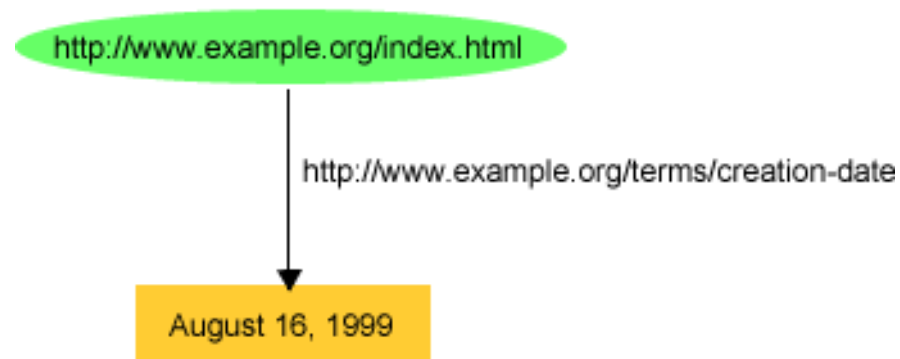
- Uporabljali bomo okrajšana imena URL naslovov
- **Imena** so definirana na **določenih URL-jih**
- Imena, ki jih uporabljamo so **pripone**
- **Qname prefiksi:**

```
prefix rdf:, namespace URI: http://www.w3.org/1999/02/22-rdf-syntax-ns#  
prefix rdfs:, namespace URI: http://www.w3.org/2000/01/rdf-schema#  
prefix dc:, namespace URI: http://purl.org/dc/elements/1.1/  
prefix owl:, namespace URI: http://www.w3.org/2002/07/owl#  
prefix ex:, namespace URI: http://www.example.org/ (or http://www.example.com/)  
prefix xsd:, namespace URI: http://www.w3.org/2001/XMLSchema#
```

- Poglejmo si zdaj okrajšan zapis primerov

# En stavek

`ex:index.html exterms:creation-date "August 16, 1999" .`



```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:exterms="http://www.example.org/terms/">
  <rdf:Description rdf:about="http://www.example.org/index.html">
    <exterms:creation-date>August 16, 1999</exterms:creation-date>
  </rdf:Description>
</rdf:RDF>
```

# Dva stavka

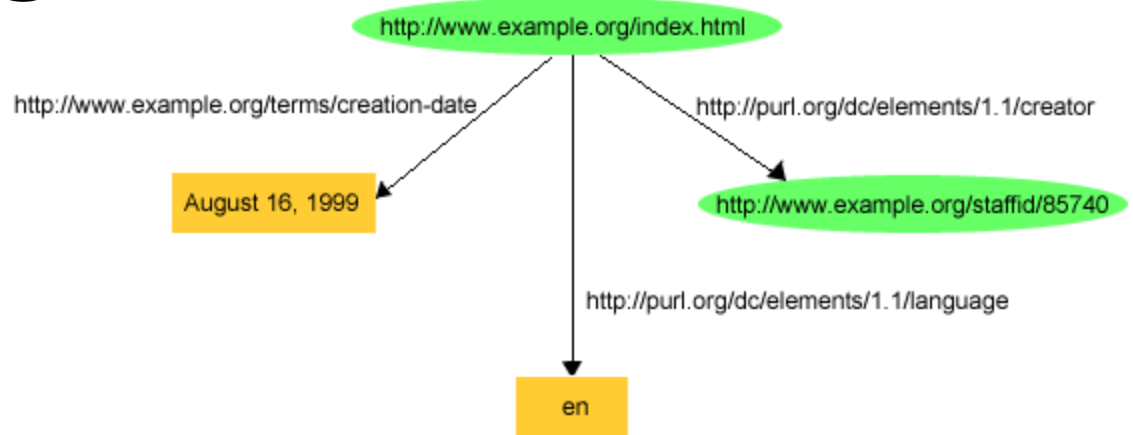
```
ex:index.html exterms:creation-date "August 16, 1999" .  
ex:index.html dc:language "en" .
```

```
<?xml version="1.0"?>  
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"  
  xmlns:dc="http://purl.org/dc/elements/1.1/"  
  xmlns:exterms="http://www.example.org/terms/">  
  
  <rdf:Description rdf:about="http://www.example.org/index.html">  
    <exterms:creation-date>August 16, 1999</exterms:creation-date>  
  </rdf:Description>  
  
  <rdf:Description rdf:about="http://www.example.org/index.html">  
    <dc:language>en</dc:language>  
  </rdf:Description>  
  
</rdf:RDF>
```



```
ex:index.html dc:creator exstaff:85740 .
ex:index.html exterms:creation-date "August 16, 1999" .
ex:index.html dc:language "en" .
```

# Trije stavki 😊



```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:dc="http://purl.org/dc/elements/1.1/"
        xmlns:exterms="http://www.example.org/terms/">
  <rdf:Description rdf:about="http://www.example.org/index.html">
    <exterms:creation-date>August 16, 1999</exterms:creation-date>
    <dc:language>en</dc:language>
    <dc:creator rdf:resource="http://www.example.org/staffid/85740"/>
  </rdf:Description>
</rdf:RDF>
```

# Trije stavki

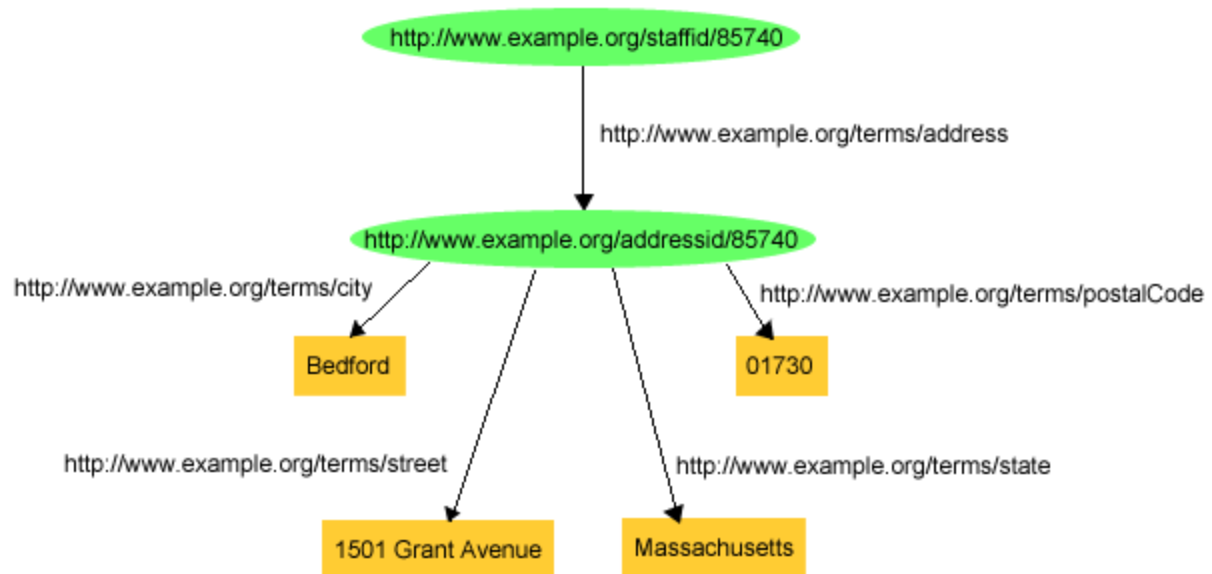
```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
          xmlns:dc="http://purl.org/dc/elements/1.1/"
          xmlns:extterms="http://www.example.org/terms/">
  <rdf:Description rdf:about="http://www.example.org/index.html">
    <extterms:creation-date>August 16, 1999</extterms:creation-date>
    <dc:language>en</dc:language>
    <dc:creator rdf:resource="http://www.example.org/staffid/85740"/>
  </rdf:Description>
</rdf:RDF>
```

# Kompleksne vrednosti

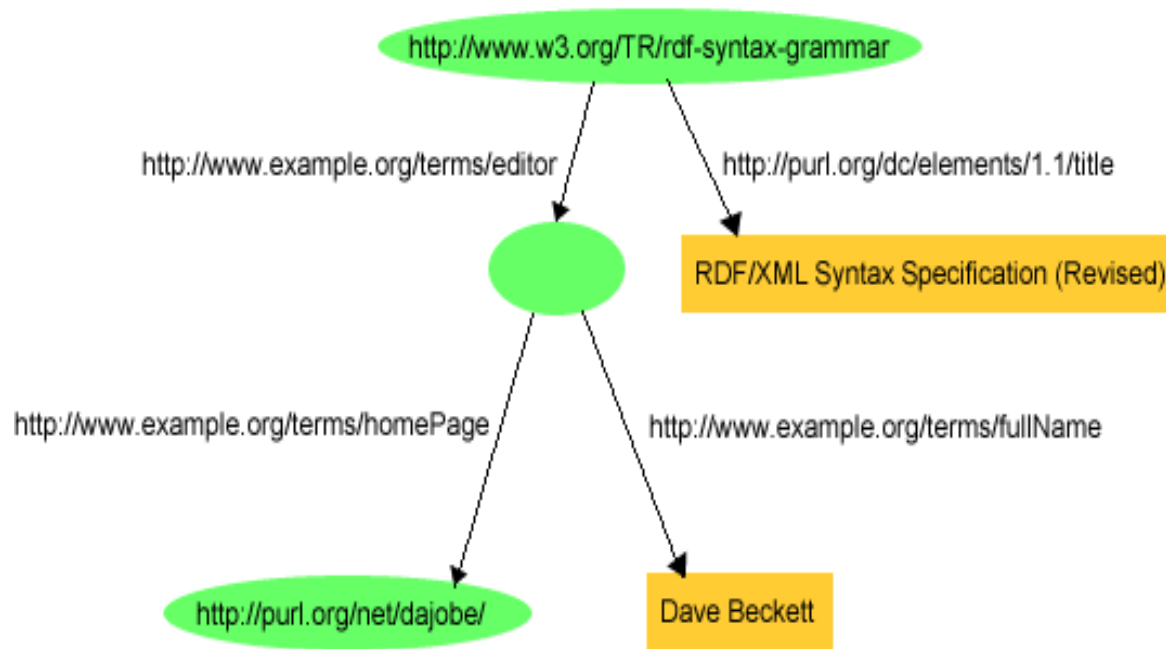
- Vrednosti lastnosti niso samo enostavni nizi
- Vrednost lastnosti je lahko tudi vozlišče grafa (ki ustreza viru)
  - Možna so poljubno kompleksna drevesa in grafi
  - Sintaktično: vrednosti so lahko vgnezdene (leksikalno) ali referencirane (povezane)

# Strukturirane vrednosti

```
exstaff:85740  exterm:address  exaddressid:85740 .  
exaddressid:85740  exterm:street "1501 Grant Avenue" .  
exaddressid:85740  exterm:city  "Bedford" .  
exaddressid:85740  exterm:state  "Massachusetts" .  
exaddressid:85740  exterm:postalCode "01730" .
```



# Prazno vozlišče



Na prazno vozlišče se lahko skličemo večkrat !

# Prazno vozlišče

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:dc="http://purl.org/dc/elements/1.1/"
         xmlns:exterms="http://example.org/stuff/1.0/">

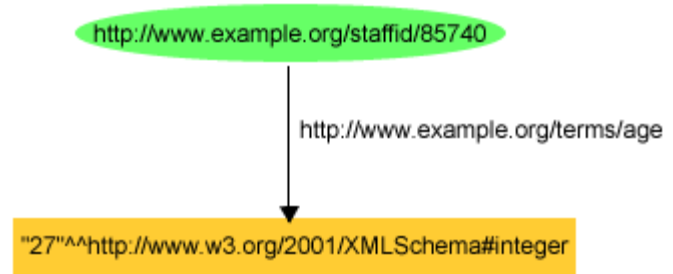
  <rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar">
    <dc:title>RDF/XML Syntax Specification (Revised)</dc:title>
    <exterms:editor rdf:nodeID="abc"/>
  </rdf:Description>

  <rdf:Description rdf:nodeID="abc">
    <exterms:fullName>Dave Beckett</exterms:fullName>
    <exterms:homePage rdf:resource="http://purl.org/net/dajobe/">
  </rdf:Description>

</rdf:RDF>
```

# Tip skalarja

- Skalarju definiramo tip
- 27 pomeni celo število in ne znaka “2” in “7”

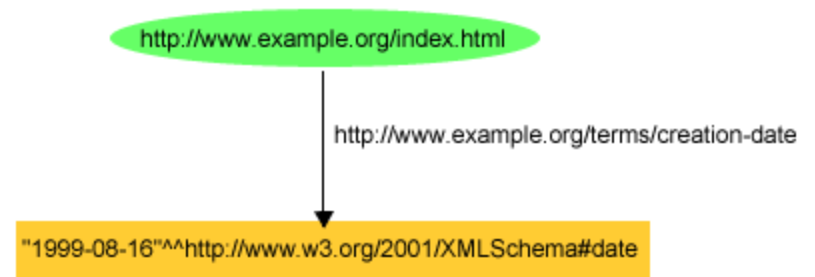


exstaff:85740 exterms:age "27"^^xsd:integer .

<http://www.example.org/staffid/85740> <http://www.example.org/terms/age>  
"27"^^<http://www.w3.org/2001/XMLSchema#integer> .

- RDF nima vgrajenih tipov
- Tipi so definirani izven RDF: **datatype URI**
- **XML Schema enostavni tipi**
- xsd:integer, xsd:float, xsd:double, xsd:boolean, xsd:boolean, xsd: date, ...

# Tip skalarja



```
ex:index.html exterms:creation-date "1999-08-16"^^xsd:date .
```

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:exterms="http://www.example.org/terms/">
  <rdf:Description rdf:about="http://www.example.org/index.html">
    <exterms:creation-date rdf:datatype=
      "http://www.w3.org/2001/XMLSchema#date">1999-08-16 </exterms:creation-
      date>
  </rdf:Description>
</rdf:RDF>
```



# Tip skalarja in entiteta

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
```

# Uporaba ID

- Do zdaj smo uporabljali `rdf:about` za specifikacijo objekta, ki ga opisujemo
  - Opisovanje virov
- Včasih želimo eksplicitno definirati identifikator objekta
  - Primer: katalog objektov na določenem naslovu
- Podobno XML ID: unikaten znotraj osnovnega URI

# Primer

- Kolekcija objektova na naslovu:  
<http://www.example.com/2007/04/products>

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
```

# Identifikacija RDF objekta

- Vsak izdelek ima svoj identifikator.

<http://www.example.com/2007/04/products#item10245>

- Namesto `rdf:ID="item10245"` bi lahko uporabili tudi `rdf:about="#item10245"`
- Če bi opisovali objekt izven konteksta danega naslova bi se opis začel:

```
<rdf:Description rdf:about="http://www.example.com/2002/04/products#item10245">
```

- Osnovni arhitekturni princip spleta:
  - lahko dodajamo podatke o nekem objektu!
- **XML Base:**
  - definiramo osnovni vir, ki ga opisujemo

# Uporaba XML Base

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [<!ENTITY xsd "http://www.w3.org/2001/XMLSchema#">]>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:exterms="http://www.example.com/terms/"
        xml:base="http://www.example.com/2002/04/products">

  <rdf:Description rdf:ID="item10245">
    <exterms:model rdf:datatype="&xsd:string">Overnighter</exterms:model>
    <exterms:sleeps rdf:datatype="&xsd:integer">2</exterms:sleeps>
    <exterms:weight rdf:datatype="&xsd:decimal">2.4</exterms:weight>
    <exterms:packedSize rdf:datatype="&xsd:integer">784</exterms:packedSize>
  </rdf:Description>

  ...other product descriptions...

</rdf:RDF>
```

# Tipi virov

- Vire oz. objekte lahko **klasificiramo v razrede**
  - Podobno programskim jezikom
- Vnaprej definiran koncept `rdf:type`
  - Vrednost te lastnosti predstavlja **kategorijo ali razred stvari**
  - Koncept definiran s tipom je instanca tipa (razreda)

# Opis primerka tipa

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
```

# Definicija tipov

- Sam RDF nima gradnikov za definicijo novih tipov
- Definicijo tipov omogoča **RDF Schema**
  - razširitev slovarja za delo s tipi
- Instance razreda lahko definiramo na enostavnejši način
  - `rdf:type` lastnost je odstranjena
  - Uporabimo QName ime za definicijo značk



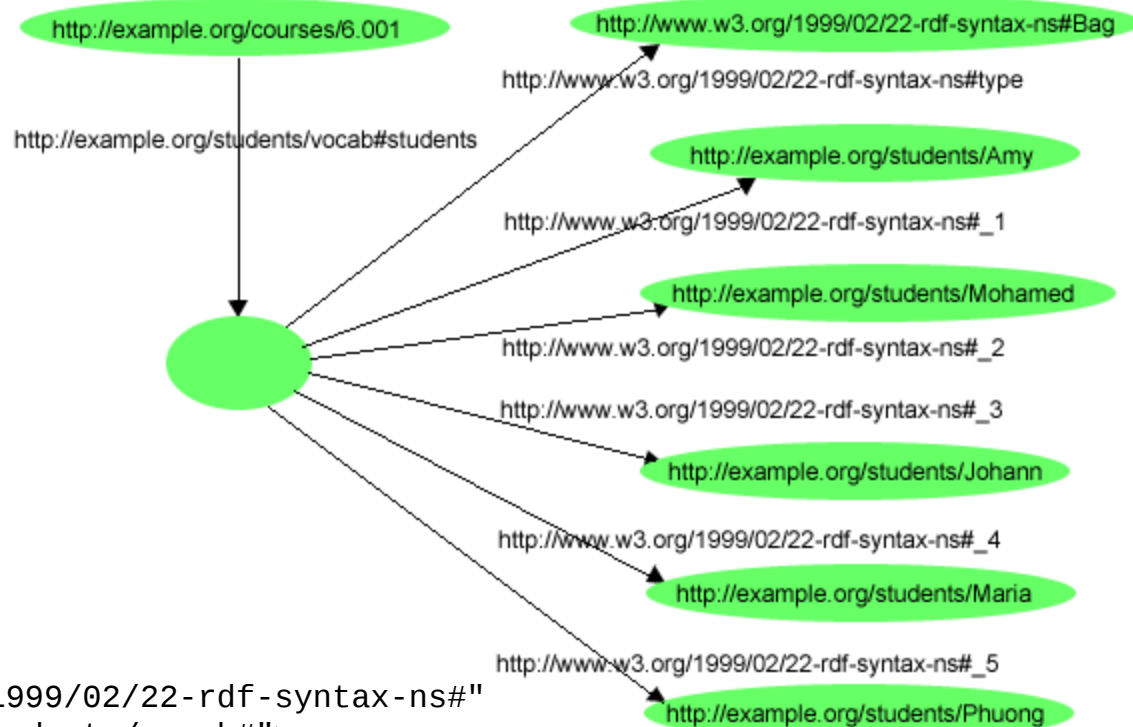
# Krajša predstavitev primerkov

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [http://www.w3.org/1999/02/22-rdf-syntax-ns#
  xmlns:exterms="http://www.example.com/terms/"
  xml:base="http://www.example.com/2002/04/products">
  <exterms:Tent rdf:ID="item10245">
    <exterms:model
      rdf:datatype="&xsd:string">Overnighter</exterms:model>
    <exterms:sleeps rdf:datatype="&xsd:integer">2</exterms:sleeps>
    <exterms:weight rdf:datatype="&xsd:decimal">2.4</exterms:weight>
    <exterms:packedSize
      rdf:datatype="&xsd:integer">784</exterms:packedSize>
  </exterms:Tent>
  ...other product descriptions...
</rdf:RDF>
```

# Kontejnerji

- Kontejnerji omogočajo **grupiranje virov** (ali besed)
- Napišemo lahko **izjave o kontejnerju** (kot celota) ali individualno o njegovih članih
- Poznamo različne tipe kontejnerjev
  - **Vreča** (rdf:Bag) – neurejena kolekcija
  - **Zaporedje** (rdf:Seq) – urejena kolekcija (= “sekvenca”)
  - **Alternative** (rdf:Alt) – predstavlja alternative
- Možno je tudi kreirati kolekcije na osnovi vzorcev URI
- Dvojniki so dovoljeni (ni mehanizma za zagotavljanje unikatnosti vrednosti)

# Vreča



```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:s="http://example.org/students/vocab#">
  <rdf:Description rdf:about="http://example.org/courses/6.001">
    <s:students>
      <rdf:Bag>
        <rdf:li rdf:resource="http://example.org/students/Amy"/>
        <rdf:li rdf:resource="http://example.org/students/Mohamed"/>
        <rdf:li rdf:resource="http://example.org/students/Johann"/>
        <rdf:li rdf:resource="http://example.org/students/Maria"/>
        <rdf:li rdf:resource="http://example.org/students/Phuong"/>
      </rdf:Bag>
    </s:students>
  </rdf:Description>
</rdf:RDF>
```

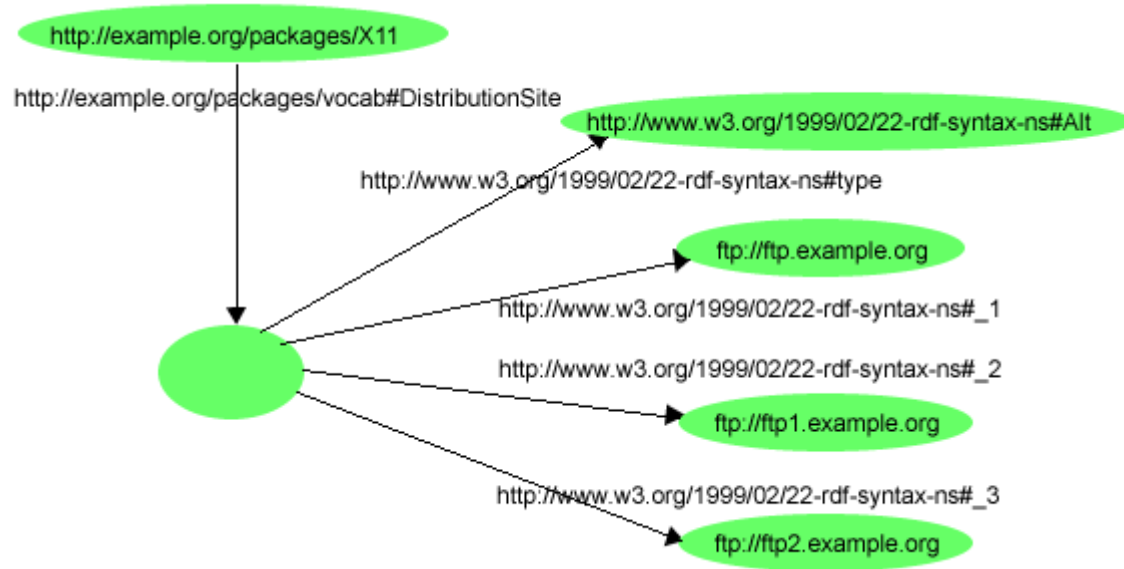
# Dostop do elementov

- Lastnosti, ki omogočajo dostop do članov kontejnerja
  - `rdf:_1`, `rdf:_2`, ...`rdf:_n`
  - Člane definiramo z `rdf:li`
- Lahko definiramo tudi dodatne lastnosti kontejnerjev

# Alternative

- Predstavitev možnosti, ki so na razpolago
- Kontejner mora imeti vsaj en element
- Ni možnosti za definicijo privzete alternative
  - Lahko definiramo sami
  - Pomen je na strani aplikacije!
- Uporabniki lahko definirajo kontejnerje na svoj način
  - **RDF kontejnerji samo nudijo skupno definicijo!**

# Alternative

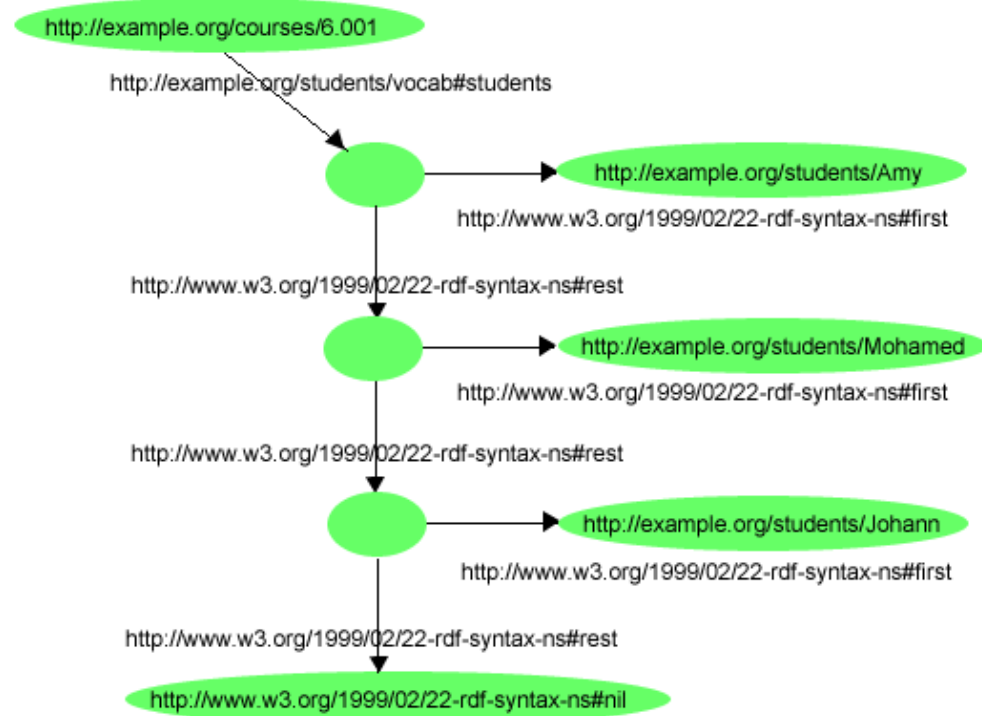


```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:s="http://example.org/packages/vocab#">
  <rdf:Description rdf:about="http://example.org/packages/X11">
    <s:DistributionSite>
      <rdf:Alt>
        <rdf:li rdf:resource="ftp://ftp.example.org"/>
        <rdf:li rdf:resource="ftp://ftp1.example.org"/>
        <rdf:li rdf:resource="ftp://ftp2.example.org"/>
      </rdf:Alt>
    </s:DistributionSite>
  </rdf:Description>
</rdf:RDF>
```

# Kolekcije

- Kontejnerje ne moremo “zapreti”
  - Ne moremo reči: “to so vsi člani kontejnerja”?
  - Kontejner pravi: “nekateri identificirani viri so člani”
  - Ni mogoče povedati, da je nekje še dodaten graf, ki opisuje še druge člane
- Skupina objektov predstavljena z RDF seznamom
- Predefiniran tip **rdf:List**
  - **rdf:first**, **rdf:rest**
- Posebna notacija za opis kolekcij  
**rdf:parseType="Collection"**
  - Seznam članov
  - Vsebina elementa naj se obravnava na poseben način !

# Kolekcija



```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:s="http://example.org/students/vocab#">
  <rdf:Description rdf:about="http://example.org/courses/6.001">
    <s:students rdf:parseType="Collection">
      <rdf:Description rdf:about="http://example.org/students/Amy"/>
      <rdf:Description rdf:about="http://example.org/students/Mohamed"/>
      <rdf:Description rdf:about="http://example.org/students/Johann"/>
    </s:students>
  </rdf:Description>
</rdf:RDF>
```



# Daljša verzija

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:s="http://example.org/students/vocab#">

  <rdf:Description rdf:about="http://example.org/courses/6.001">
    <s:students rdf:nodeID="sch1"/>
  </rdf:Description>

  <rdf:Description rdf:nodeID="sch1">
    <rdf:first rdf:resource="http://example.org/students/Amy"/>
    <rdf:rest rdf:nodeID="sch2"/>
  </rdf:Description>

  <rdf:Description rdf:nodeID="sch2">
    <rdf:first rdf:resource="http://example.org/students/Mohamed"/>
    <rdf:rest rdf:nodeID="sch3"/>
  </rdf:Description>

  <rdf:Description rdf:nodeID="sch3">
    <rdf:first rdf:resource="http://example.org/students/Johann"/>
    <rdf:rest rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#nil"/>
  </rdf:Description>

</rdf:RDF>
```

# Stavki višjega reda

- RDF stavki o RDF stavkih
  - primer: “Tone misli, da splet vsebuje bilijon dokumentov”
- Stavki višjega reda
  - Opišemo lahko lastnosti o lastnostih
  - Pomembni za modele zaupanja, digitalne signature, itd.
  - Meta-podatki
  - Tudi meta-podatki o meta-podatkih
  - RDF lahko opišemo v RDF-ju

⇒ “reification”

# Reifikacija

- Shranjevanje podatkov o drugih stavkih
- Kdo je kreiral stavek, kdaj, ...
- RDF ima vgrajen slovar za opisovanje stavkov
- **Tip:** `rdf:Statement`
  - Definiramo stavek, ki bo služil kot višjenivojski koncept
    1. Definiramo stavek
    2. Dodamo dodatne lastnosti
- **Lastnosti** `rdf:Statement` za definicijo stavka:  
`rdf:subject,rdf:predicate,rdf:object`

# Reifikacija



```
exproducts:triple12345 rdf:type rdf:Statement .  
exproducts:triple12345 rdf:subject exproducts:item10245 .  
exproducts:triple12345 rdf:predicate exterms:weight .  
exproducts:triple12345 rdf:object "2.4"^^xsd:decimal .  
exproducts:triple12345 dc:creator exstaff:85740 .
```

# Reifikacija

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [<!ENTITY xsd ttp://www.w3.org/2001/XMLSchema#>]>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:extterms="http://www.example.com/terms/"
  xml:base="http://www.example.com/2002/04/products">

  <rdf:Description rdf:ID="item10245">
    <extterms:weight rdf:datatype="&xsd;decimal">2.4</extterms:weight>
  </rdf:Description>

  <rdf:Statement rdf:about="#triple12345">
    <rdf:subject rdf:resource="http://www.example.com/2002/04/products#item10245"/>
    <rdf:predicate rdf:resource="http://www.example.com/terms/weight"/>
    <rdf:object rdf:datatype="&xsd;decimal">2.4</rdf:object>
    <dc:creator rdf:resource="http://www.example.com/staffid/85740"/>
  </rdf:Statement>

</rdf:RDF>
```

# RDF shema

RDFS = RDF shema

- Definira majhen slovar za RDF:
  - rdfs:class, rdfs:subClassOf, rdfs:type
  - rdfs:property, rdfs:subPropertyOf
  - rdfs:domain, rdfs:range
- Ustreza množici RDF predikatov:
  - ⇒ meta-nivo
  - ⇒ poseben vnaprej definiran pomen

# RDFS

**Slovar** s vnaprej definiranimi **razredi** in **lastnostmi**

Slovar za razrede:

- `rdfs:Class` – definicija razreda
- `rdf:type` - definicija instance razreda
- `rdfs:subClassOf` - definicija podrazreda drugega razreda

# Razredi



ex:MotorVehicle rdf:type rdfs:Class .  
ex:PassengerVehicle rdf:type rdfs:Class .  
ex:Van rdf:type rdfs:Class .  
ex:Truck rdf:type rdfs:Class .  
ex:MiniVan rdf:type rdfs:Class .

ex:PassengerVehicle rdfs:subClassOf ex:MotorVehicle .  
ex:Van rdfs:subClassOf ex:MotorVehicle .  
ex:Truck rdfs:subClassOf ex:MotorVehicle .

ex:MiniVan rdfs:subClassOf ex:Van .  
ex:MiniVan rdfs:subClassOf ex:PassengerVehicle .



```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [<!ENTITY xsd http://www.w3.org/2001/XMLSchema#>]>

<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xml:base="http://example.org/schemas/vehicles">

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

  <rdf:Description rdf:ID="PassengerVehicle">
    <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
    <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
  </rdf:Description>

  <rdf:Description rdf:ID="Truck">
    <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
    <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
  </rdf:Description>

  <rdf:Description rdf:ID="Van">
    <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
    <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
  </rdf:Description>

  <rdf:Description rdf:ID="MiniVan">
    <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
    <rdfs:subClassOf rdf:resource="#Van"/>
    <rdfs:subClassOf rdf:resource="#PassengerVehicle"/>
  </rdf:Description>

</rdf:RDF>
```

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [<!ENTITY xsd "http://www.w3.org/2001/XMLSchema#">]>
<rdf:RDF
  xmlns:rdf=http://www.w3.org/1999/02/22-rdf-syntax-ns#
  xmlns:rdfs=http://www.w3.org/2000/01/rdf-schema#
  xml:base="http://example.org/schemas/vehicles">

  <rdfs:Class rdf:ID="MotorVehicle"/>

  <rdfs:Class rdf:ID="PassengerVehicle">
    <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
  </rdfs:Class>

  <rdfs:Class rdf:ID="Truck">
    <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
  </rdfs:Class>

  <rdfs:Class rdf:ID="Van">
    <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
  </rdfs:Class>

  <rdfs:Class rdf:ID="MiniVan">
    <rdfs:subClassOf rdf:resource="#Van"/>
    <rdfs:subClassOf rdf:resource="#PassengerVehicle"/>
  </rdfs:Class>

</rdf:RDF>
```

Krajši zapis tipov

# Primerek vozila

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ex="http://example.org/schemas/vehicles#"
  xml:base="http://example.org/things">

  <ex:MotorVehicle rdf:ID="companyCar"/>

</rdf:RDF>
```

# RDFS lastnosti

Slovar za lastnosti:

- RDF razred:
  - `rdfs:Property` – definicija lastnosti
- RDF schema lastnosti:
  - `rdfs:domain` – definicija prve komponente lastnosti
  - `rdfs:range` – definicija druge komponente lastnosti
  - `rdfs:subPropertyOf` - izraža ISA relacijo med lastnostmi

# Opisovanje lastnosti

- Opisujemo **lastnosti razredov** podobno kot razrede v programskih jezikih
- Vse lastnosti so definirane kot primerki `rdf:Property`
- Ni omejitev glede **števnosti** lastnosti!
- Pri definiciji zaloge vrednosti lahko uporabljamo tudi osnovne tipe npr. `xsd:integer`

# Primer

- Lastnost `ex:author` ima domeno `ex:Book` in zalogo vrednosti razred `ex:Person`

```
ex:Book rdf:type rdfs:Class .  
ex:Person rdf:type rdfs:Class .  
ex:author rdf:type rdf:Property .  
ex:author rdfs:domain ex:Book .  
ex:author rdfs:range ex:Person .
```

- Lastnost ima lahko več kot eno zalogo vrednosti:

```
ex:hasMother rdfs:range ex:Female .  
ex:hasMother rdfs:range ex:Person .
```

# Lastnosti vozil

- Avto je registrirala oseba:

```
<rdf:Property rdf:ID="registeredTo">  
  <rdfs:domain rdf:resource="#MotorVehicle"/>  
  <rdfs:range rdf:resource="#Person"/>  
</rdf:Property>
```

- Razdalja med sedeži:

```
<rdf:Property rdf:ID="rearSeatLegRoom">  
  <rdfs:domain rdf:resource="#PassengerVehicle"/>  
  <rdfs:range rdf:resource="&xsd;integer"/>  
</rdf:Property>
```

# Pod-lastnosti

- Primarni voznik je podlastnost voznika:

```
ex:driver rdf:type rdf:Property .
```

```
ex:primaryDriver rdf:type rdf:Property .
```

```
ex:primaryDriver rdfs:subPropertyOf ex:driver .
```

```
<rdf:Property rdf:ID="driver">
```

```
  <rdfs:domain rdf:resource="#MotorVehicle"/>
```

```
</rdf:Property>
```

```
<rdf:Property rdf:ID="primaryDriver">
```

```
  <rdfs:subPropertyOf rdf:resource="#driver"/>
```

```
</rdf:Property>
```



```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [<!ENTITY xsd "http://www.w3.org/2001/XMLSchema#">]>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xml:base="http://example.org/schemas/vehicles">

  <rdfs:Class rdf:ID="MotorVehicle"/>

  <rdfs:Class rdf:ID="PassengerVehicle">
    <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
  </rdfs:Class>

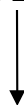
  <rdfs:Class rdf:ID="Truck">
    <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
  </rdfs:Class>

  <rdfs:Class rdf:ID="Van">
    <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
  </rdfs:Class>

  <rdfs:Class rdf:ID="MiniVan">
    <rdfs:subClassOf rdf:resource="#Van"/>
    <rdfs:subClassOf rdf:resource="#PassengerVehicle"/>
  </rdfs:Class>

  <rdfs:Class rdf:ID="Person"/>

  <rdfs:Datatype rdf:about="&xsd;integer"/>
```





```
<rdf:Property rdf:ID="registeredTo">
  <rdfs:domain rdf:resource="#MotorVehicle"/>
  <rdfs:range rdf:resource="#Person"/>
</rdf:Property>

<rdf:Property rdf:ID="rearSeatLegRoom">
  <rdfs:domain rdf:resource="#PassengerVehicle"/>
  <rdfs:range rdf:resource="&xsd;integer"/>
</rdf:Property>

<rdf:Property rdf:ID="driver">
  <rdfs:domain rdf:resource="#MotorVehicle"/>
</rdf:Property>

<rdf:Property rdf:ID="primaryDriver">
  <rdfs:subPropertyOf rdf:resource="#driver"/>
</rdf:Property>

</rdf:RDF>
```

# RDF + RDFS: semantika

- Natančen pomen RDF(S) grafa
  - Začetno formalna semantika ni bila definirana!
  - Glavni problemi:
    - bnode
    - meta-modeliranje
    - Formalna semantika RDFS slovarja
  - Semantika je osnovana na **modelih**
- ⇒ Formalna definicija izpeljav in odgovorov na vprašanja nad RDF(S) grafi

# Nepopolna informacija v RDF grafih

- bnode = eksistenčne vrednosti (null vrednosti)
- ⇒ nepopolna informacija v RDF grafih
- RDF graf lahko vidimo kot **nepopolno podatkovno bazo** predstavljeno s **tabelo**
  - RDF graf je predstavljen z unikatno tabelo T z vrednostmi, ki so bodisi konstante ali imenovane eksistenčne spremenljivke
  - RDF graf lahko vidimo kot **boolov konjuktivni izraz** nad **relacijo T**

# Formalna semantika RDF + RDFS

- Formalna semantika RDFS slovarja
- RDFS stavki = omejitve nad RDF grafi
- Izpeljave RDF + RDFS = sklepanje (odgovarjanje na poizvedbe) nad nepopolno bazo z omejitvami

# Viri

- <http://www.w3.org/TR/rdf-primer/>
- <http://www.w3.org/TR/rdf-mt/>