

# RDF a RDFS interpretácie

## Spracované podľa

dokumentov W3C

- <http://www.w3.org/TR/rdf11-mt/>

a prednášok

- <http://www.inf.tu-dresden.de/content/institutes/ki/cl/study/summer14/fswt/slides/FSWT2014-L4-RDSF-Semantics.pdf>

# Triedy interpretácií - opakovanie

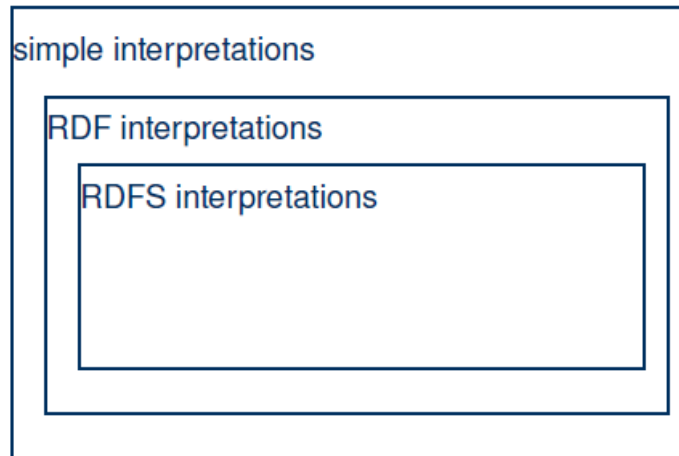
Ked sa zaoberáme interpretáciami nejakého formálneho jazyka, môžeme uvažovať o akýchkoľvek interpretáciách ale môžeme požadovať aby interpretácie mali isté vlastnosti.

## Sémanticke podmienky

- vyjadrené v jazyku **predikátového počtu prvého rádu** (FOL)

## Semantics of RDF(S)

- we proceed stepwise:



- the more we restrict the set of interpretations, the stronger the consequence relation becomes

# Triedy interpretácií - opakovanie

**Jednoduchá interpretácia** môže menám slovníka priradiť akýkoľvek význam:

- Univerzum môže byť akákoľvek množina prvkov, pojmov, vzťahov...
- Hodnoty zobrazenia  $I_S$  a  $I_L$  môžu byť akékoľvek objekty/pojem z univerza
- Hodnoty funkcie  $I_{EXT}$  akékoľvek binárne relácie nad univerzom

Nekladie na tieto množiny a zobrazenia žiadne podmienky či obmedzenia.

V ďalšom sa budeme zaoberať triedami interpretácií, ktoré spĺňajú určité **sémanticke podmienky – axiómy**, ktoré sa považujú za platné (v danej triede interpretácií) S využitím týchto axióm a sémantických podmienok je možné z daného predpokladu (RDF-grafu) dedukovať viac dôsledkov ako pri jednoduchej interpretácii.

Nech  $\mathcal{I}$  je nejaká trieda interpretácií , potom platí:

Ak  $E$  je jednoduchým dôsledkom  $G$ , potom je  $E$  dôsledkom aj v triede interpretácií  $\mathcal{I}$  .

# RDF interpretácia

## RDF slovník:

- `rdf:type`
- `rdf:Property`
- `rdf:langString`
- `rdf:subject rdf:predicate rdf:object`
- `rdf:first rdf:rest rdf:value rdf:nil rdf:List rdf:_1 rdf:_2 ...`

**RDF interpretácia** je taká jednoduchá interpretácia, ktorá dáva menám zo **slovníka RDF** konkrétny význam formálne popísaný sémantickými podmienkami a RDF-axiómami.

## Sémantická podmienka pre RDF interpretácie:

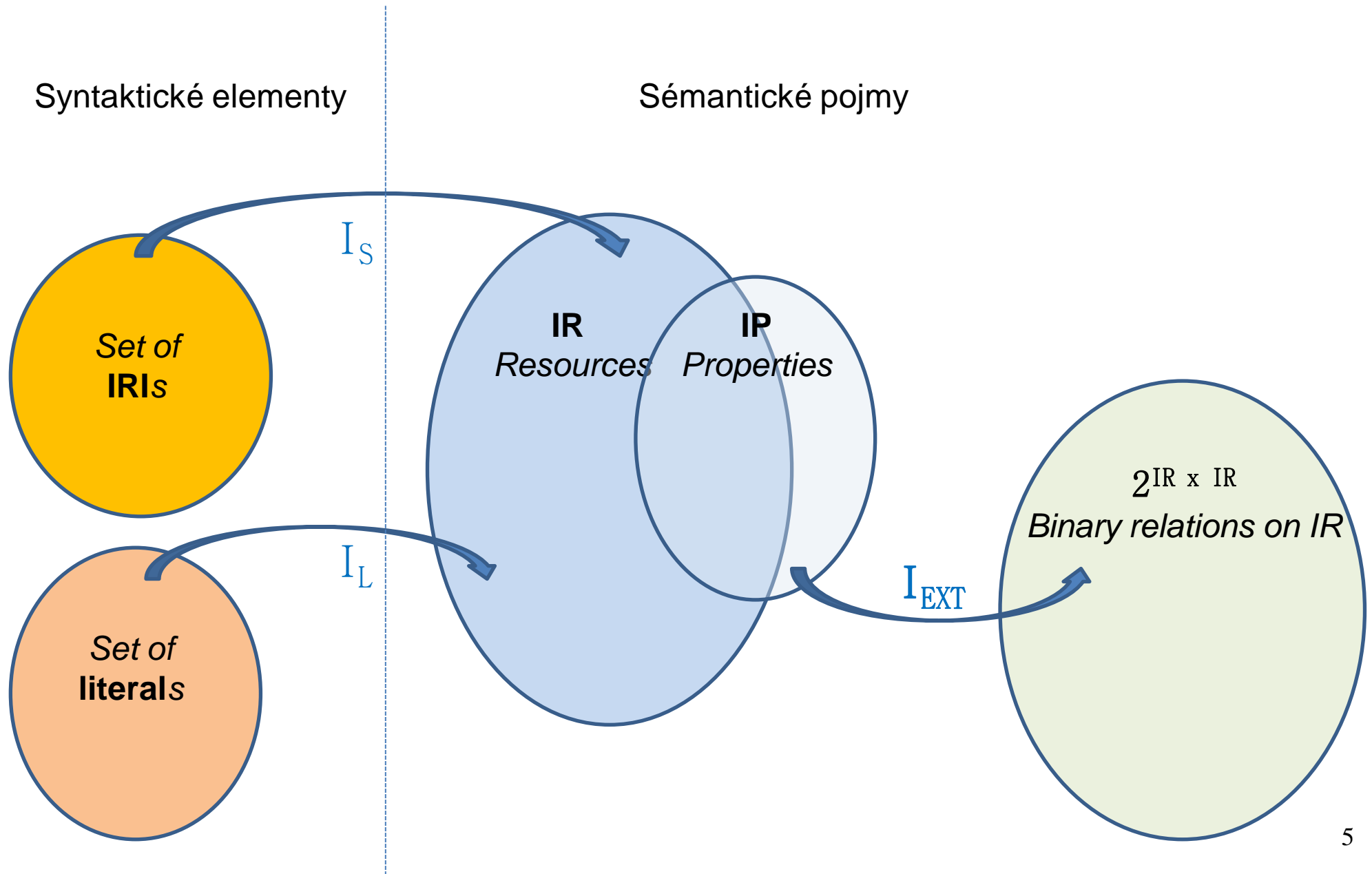
$IP \subseteq IR$  a  $I_S(p) \in IP$  práve vtedy keď  $\langle I_S(p), I(\text{rdf:Property}) \rangle \in I_{\text{EXT}}(I(\text{rdf:type}))$ .

Podmienka hovorí, že každá property musí byť prvkom IR a má `rdf:type` `rdf:Property`.

Teda IRI `rdf:Property` je identifikátorom množiny všetkých properties.

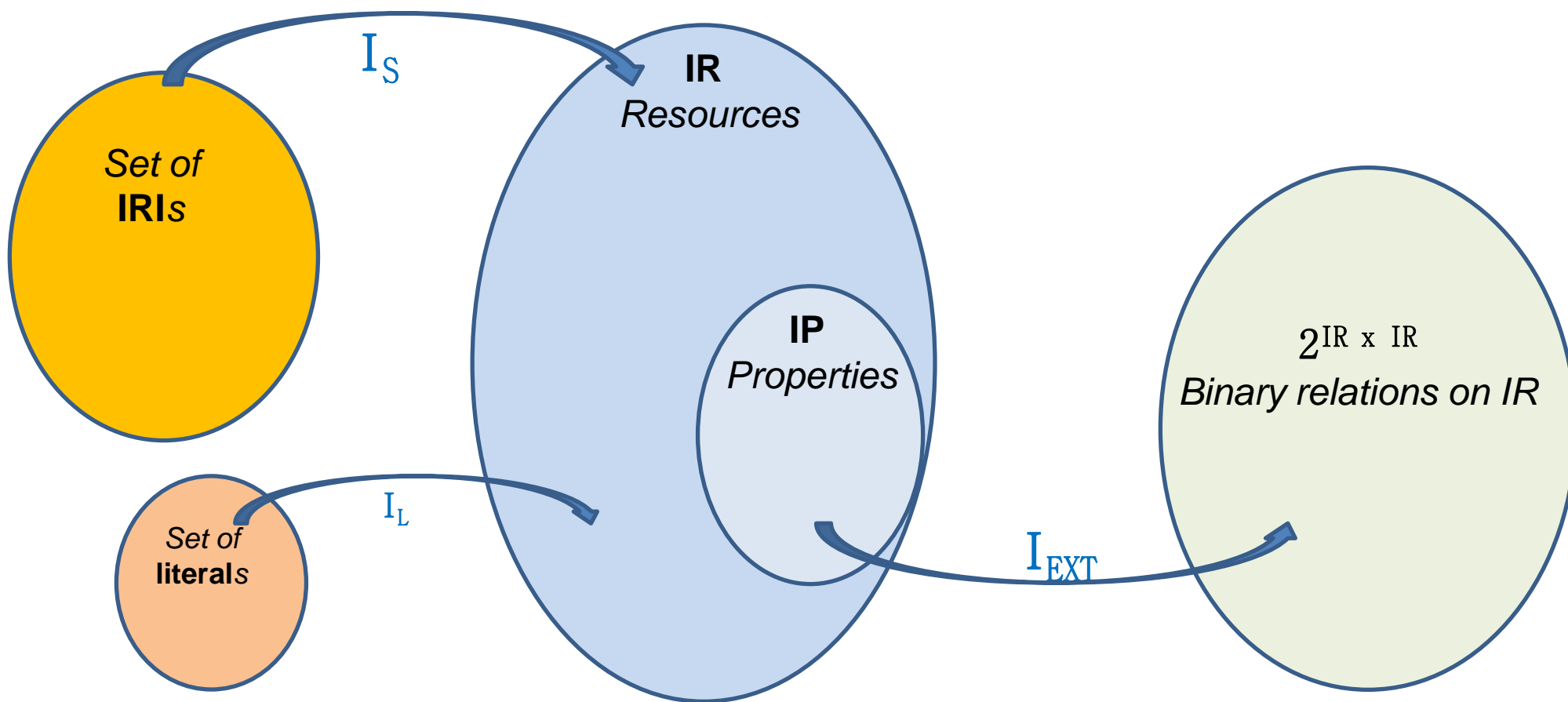
# Jednoduchá interpretácia – diagram

Opakovanie



# RDF interpretácia - diagram

V jednoduchá interpretácia nevyžaduje aby boli množiny zdrojov IR a properties IP v nejakom vzťahu. V RDF interpretácii sú všetky properties prvkami univerza IR. Teda v RDF interpretácii platí  $IP \subseteq IR$



# Axiómy a dedukčné pravidlá pre RDF

Zo sémantickej podmienky vyplýva jednoduché **dedukčné pravidlo** pre akúkoľvek trojicu  $s p o$  :

$$\text{rdf1} \quad \frac{s \quad p \quad o}{p \quad \text{rdf:type} \quad \text{rdf:Property}}$$

*Nasledujúce axiómy predpisujú typ, ktorý musia mať ostatné IRI zo slovníka RDF.*

rdf:type	rdf:type	rdf:Property .
rdf:subject	rdf:type	rdf:Property .
rdf:predicate	rdf:type	rdf:Property .
rdf:object	rdf:type	rdf:Property .
rdf:first	rdf:type	rdf:Property .
rdf:rest	rdf:type	rdf:Property .
rdf:value	rdf:type	rdf:Property .
rdf:nil	rdf:type	rdf:List .
rdf:_1	rdf:type	rdf:Property .
rdf:_2	rdf:type	rdf:Property .

...

*Terminologická poznámka: Axiómy sú sémantické podmienky, ktoré je možné vyjadriť v jazyku RDF.*

Keďže tieto trojice sú axiómy, musia byť vždy splnené, čo možno vyjadriť **dedukčným pravidlom**:

$$\text{rdfax} \quad \frac{}{s \quad p \quad o}$$

kde  $s p o$  je axiomatická trojica

# RDF dôsledok

Hovoríme, že graf E je **RDF dôsledkom** grafu G ak je E pravdivý vo všetkých RDF interpretáciách, v ktorých je pravdivý G.

**Veta:** Nech graf G neobsahuje žiadne literály s dátovým typom ani jazykovo lokalizované literály, potom platí:

Graf E je RDF-dôsledkom grafu G práve vtedy keď existuje graf G0 taký, že

- G0 môže byť odvodený z grafu G pomocou dedukčných pravidiel ***rdf1*** a ***rdfax***
- E je jednoduchým dôsledkom G0.

Poznámka: V materiáloch TUD je uvedené tvrdenie, ktoré nemá obmedzenie na literály použité v grafoch. Na literály s typom alebo jazykovou lokalizáciou sa vzťahujú ďalšie sémantické podmienky a z nich plynú dodatočné dedukčné pravidlá. Tieto podmienky pravidlá sme vynechali pretože problematika literálov je veľmi široká a živá (zmeny aj v štandardoch). Informatívne sa dotkneme tejto témy na konci prednášky.



# RDFS interpretácia

Meno *rdf:type* bolo definované v slovníku RDF a slúžilo na definovanie množiny properties ako podmnožiny univerza.

RDFS umožňuje definovať aj iné podmnožiny univerza a použiť *rdf:type* na vyjadrenie príslušnosti k nim. Teda *rdf:type* má sémantiku vzťahu „byť prvkom množiny“ tak ako ho poznáme v matematike.

**Priklad:** trojica *ex:rex rdf:type ex:Pes* vyjadruje, že rex je pes, t.j. patrí do množiny psov.

**Terminologická poznámka:** v RDF používame inú terminológiu ako v teórii množín

TM	RDF
množina	trieda (class)
prvok	objekt (resource)
byť prvkom	mať typ
binárna relácia	vzťah (property)

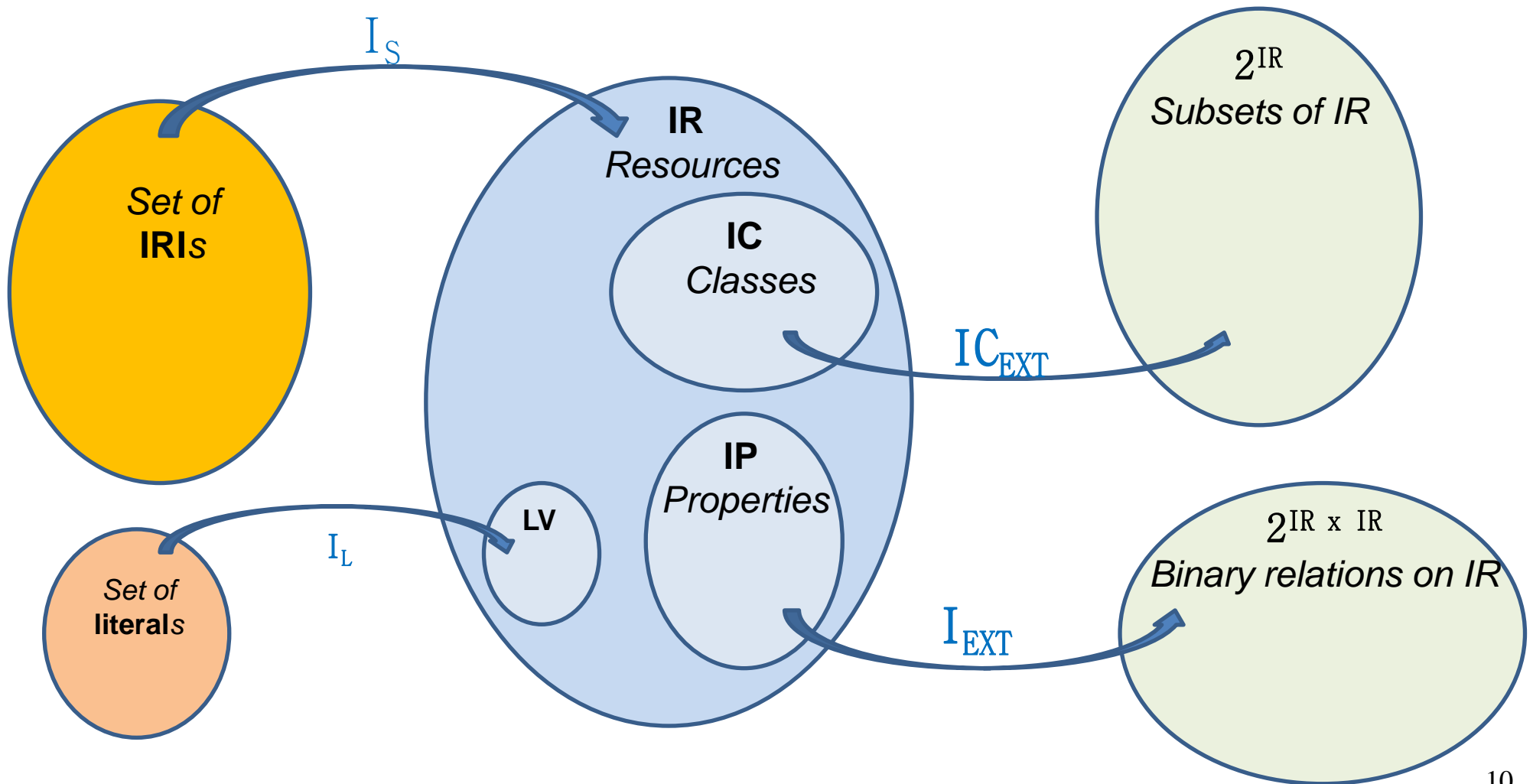
**RDFS interpretácia** je **RDF interpretácia**, ktorá navyše definuje:

- množinu  $IC$  všetkých tried. Triedy sú objekty univerza typu *rdfs:Class*.
- a funkciu  $IC_{EXT} : IR \rightarrow 2^{IR}$ , ktorá každej triede *t* priradí množinu jej prvkov t.j. objektov, ktoré majú rdf-typ *t*.

Okrem typu *rdfs:Class* definuje RDFS slovník aj typy:

- *rdfs:Resource* pre všetky objekty univerza. T.j. Každý prvok *IR* má aj tento typ.
- *rdfs:Literal* je typom pre hodnoty literálov.

# RDFS interpretácia - diagram



# Sémantické podmienky pre RDFS interpretáciu

Formálne musia byť teda splnené nasledujúce sémantické podmienky:

- $IC_{EXT}(t) := \{ x : \langle x, t \rangle \in I_{EXT}(I(\mathit{rdf:type})) \}$

Teda  $IC_{EXT}(t)$  je množina všetkých prvkov univerza ktoré majú  $\mathit{rdf:type} t$

- $IR := IC_{EXT}(I(\mathit{rdfs:Resource}))$

$\mathit{rdfs:Resource}$  sa musí interpretovať ako trieda a to ako trieda, ktorá obsahuje všetky resourcy. Teda každý objekt univerza má typ  $\mathit{rdfs:Resource}$ .

- $IC := IC_{EXT}(I(\mathit{rdfs:Class}))$

$\mathit{rdfs:Class}$  sa musí interpretovať ako trieda všetkých tried. Teda každá trieda je objekt typu  $\mathit{rdfs:Class}$ .

- $LV := IC_{EXT}(I(\mathit{rdfs:Literal}))$

Teda každá hodnota literálu je typu  $\mathit{rdfs:Literal}$ . A  $\mathit{rdfs:Literal}$  je IRI triedy všetkých hodnôt literálov

Keďže RDFS interpretácia je zároveň RDF interpretácia, musí platiť

$$IP := \{ x : \langle x, I(\mathit{rdf:Property}) \rangle \in I_{EXT}(I(\mathit{rdf:type})) \}$$

a zároveň aj 1. podmienka RDFS interpretácie

$$IC_{EXT}(y) := \{ x : \langle x, y \rangle \in I_{EXT}(I(\mathit{rdf:type})) \}$$

Z toho pre  $y=I(\mathit{rdf:Property})$  teda plynie dôsledok

- $IP = IC_{EXT}(I(\mathit{rdf:Property}))$

Teda každá property je typu  $\mathit{rdf:Property}$ . A  $\mathit{rdfs:Property}$  je IRI triedy všetkých propertií.

# Slovník RDFS

RDFS interpretácia dáva presne definovanú sémantiku aj ďalším menám zo slovníka RDFS.

## RDFS slovník:

- **rdfs:Class** – trieda všetkých tried
- **rdfs:subClassOf** - property, ktorá reprezentuje vzťah inklúzie medzi triedami t.j. analógia vzťahu „byť podmnožinou“  $\subseteq$ .
- **rdfs:subPropertyOf** - property, ktorá reprezentuje vzťah inklúzie medzi properties. Keďže vzťahy (property) reprezentujú binárne relácie, t.j. množiny usporiadaných dvojíc je možné hovoriť aj o vzťahu inklúzie medzi vzťahmi. (pomocou inklúzie môžeme vyjadriť istý druh implikácie)
- **rdfs:Resource** **rdfs:Literal** **rdfs:Datatype** – sú triedy všetkých zdrojov, literálov, resp. dátových typov. (t.j. sú to prvky triedy **rdfs:Class**.)
- **rdfs:domain** **rdfs:range** - properties, ktoré slúžia na definovanie definičného oboru (koobor) a oboru hodnôt property. t.j. Definujú do ktorých tried patria prvé resp. druhé položky v usporiadaných dvojiciach binárnej relácie danej property.
- **rdfs:member** **rdfs:Container** **rdfs:ContainerMembershipProperty**
- **rdfs:comment** **rdfs:seeAlso** **rdfs:isDefinedBy** **rdfs:label**

Sémantika všetkých mien zo slovníka RDFS je presne definovaná ďalšími sémantickými podmienkami a axiómami RDFS

# Sémantické podmienky pre rdfs:domain a rdfs:range

## Definičný obor a obor hodnôt vzťahu/property

- Ak  $\langle p, x \rangle \in I_{EXT}(I(rdfs:domain))$  a  $\langle u, v \rangle \in I_{EXT}(p)$  potom  $u \in IC_{EXT}(x)$
- Ak  $\langle p, x \rangle \in I_{EXT}(I(rdfs:range))$  a  $\langle u, v \rangle \in I_{EXT}(p)$  potom  $v \in IC_{EXT}(x)$

t.j. ak resource x je definičným oborom vzťahu p a ďalej resource u je vo vzťahu p s resourcesom v, potom resource u je prvkom množiny reprezentovanej resourcesom x, teda u má typ x.  
(Podobne pre obor hodnôt a v)

Z týchto podmienok vyplývajú nasledujúce dedukčné pravidlá:

**rdfs2** ppp rdfs:domain xxx .  
uuu ppp vv .  
-----  
uuu rdf:type xxx .

**rdfs3** ppp rdfs:range xxx .  
uuu ppp vv .  
-----  
vvv rdf:type xxx .

# Sémantické podmienky pre rdfs:subClassOf

## Vlastnosti inklúzie medzi triedami

- $I_{EXT}(I(rdfs:subClassOf))$  je tranzitívna a reflexívna binárna relácia na  $IC$
- Ak  $\langle x, y \rangle \in I_{EXT}(I(rdfs:subClassOf))$  potom  $x \in IC, y \in IC$  a  $IC_{EXT}(x) \subseteq IC_{EXT}(y)$   
t.j. ak resource  $x$  je subClassOf resource  $y$ , potom sú  $x$  aj  $y$  triedy a množina, ktorú reprezentuje resource  $x$  je podmnožinou množiny reprezentovanej  $y$ .
- Ak  $x \in IC$  potom  $\langle x, I(rdfs:Resource) \rangle \in I_{EXT}(I(rdfs:subClassOf))$

Každá trieda je podtriedou triedy zdrojov. T.j. trieda zdrojov je koreň v strome všetkých tried

Z druhej podmienky vyplýva okrem iného aj nasledujúce dedukčné pravidlo:

```
rdfs9  xxx rdfs:subClassOf yyy .  
        zzz rdf:type xxx .  
-----  
        zzz rdf:type yyy .
```

A z tretej

```
rdfs8  xxx rdf:type rdfs:Class .  
-----  
        xxx rdfs:subClassOf rdfs:Resource .
```

# Sémantické podmienky pre rdfs:subPropertyOf

## Vlastnosti inklúzie medzi vzťahmi

- $I_{EXT}(I(rdfs:subPropertyOf))$  je tranzitívna a reflexívna binárna relácia na  $IP$
- Ak  $\langle x, y \rangle \in I_{EXT}(I(rdfs:subPropertyOf))$  potom  $x \in IP, y \in IP$  a  $I_{EXT}(x) \subseteq I_{EXT}(y)$
- Ak  $x \in IC_{EXT}(I(rdfs:ContainerMembershipProperty))$   
potom  $\langle x, I(rdfs:member) \rangle \in I_{EXT}(I(rdfs:subPropertyOf))$

*Tretia podmienka hovorí: Každý vzťah/property triedy všetkých container-membership vzťahov je pod-vzťah member vzťahu. T.j. Vzťah member je koreňom stromu všetkých container-membership vzťahov.*

Z druhej podmienky vyplýva okrem iného aj nasledujúce dedukčné pravidlo:

```
rdfs7  aaa rdfs:subPropertyOf bbb .  
        xxx aaa yyy .  
-----  
        xxx bbb yyy .
```

A z prvej

```
rdfs5  aaa rdfs:subPropertyOf bbb .  
        bbb rdfs:subPropertyOf ccc .  
-----  
        aaa rdfs:subPropertyOf ccc .
```

# RDFS axiomy

rdf:type	rdfs:domain	rdfs:Resource .
<b>rdfs:domain</b>	<b>rdfs:domain</b>	<b>rdf:Property .</b>
<b>rdfs:range</b>	<b>rdfs:domain</b>	<b>rdf:Property .</b>
<b>rdfs:subPropertyOf</b>	<b>rdfs:domain</b>	<b>rdf:Property .</b>
<b>rdfs:subClassOf</b>	<b>rdfs:domain</b>	<b>rdfs:Class .</b>
rdf:subject	rdfs:domain	rdf:Statement .
rdf:predicate	rdfs:domain	rdf:Statement .
rdf:object	rdfs:domain	rdf:Statement .
rdfs:member	rdfs:domain	rdfs:Resource .
rdf:first	rdfs:domain	rdf:List .
rdf:rest	rdfs:domain	rdf:List .
rdfs:seeAlso	rdfs:domain	rdfs:Resource .
rdfs:isDefinedBy	rdfs:domain	rdfs:Resource .
rdfs:comment	rdfs:domain	rdfs:Resource .
rdfs:label	rdfs:domain	rdfs:Resource .
rdf:value	rdfs:domain	rdfs:Resource .



# RDFS axiomy

<b>rdf:type</b>	<b>rdfs:range</b>	<b>rdfs:Class .</b>
<b>rdfs:domain</b>	<b>rdfs:range</b>	<b>rdfs:Class .</b>
<b>rdfs:range</b>	<b>rdfs:range</b>	<b>rdfs:Class .</b>
<b>rdfs:subPropertyOf</b>	<b>rdfs:range</b>	<b>rdf:Property .</b>
<b>rdfs:subClassOf</b>	<b>rdfs:range</b>	<b>rdfs:Class .</b>
rdf:subject	rdfs:range	rdfs:Resource .
rdf:predicate	rdfs:range	rdfs:Resource .
rdf:object	rdfs:range	rdfs:Resource .
rdfs:member	rdfs:range	rdfs:Resource .
rdf:first	rdfs:range	rdfs:Resource .
rdf:rest	rdfs:range	rdf:List .
rdfs:seeAlso	rdfs:range	rdfs:Resource .
rdfs:isDefinedBy	rdfs:range	rdfs:Resource .
rdfs:comment	rdfs:range	rdfs:Literal .
rdfs:label	rdfs:range	rdfs:Literal .
rdf:value	rdfs:range	rdfs:Resource .

# RDFS axiomy

rdfs:isDefinedBy

rdfs:subPropertyOf

rdfs:seeAlso .

rdfs:Datatype

rdfs:subClassOf

rdfs:Class .

rdf:Alt

rdfs:subClassOf

rdfs:Container .

rdf:Bag

rdfs:subClassOf

rdfs:Container .

rdf:Seq

rdfs:subClassOf

rdfs:Container .

rdfs:ContainerMembershipProperty

rdfs:subClassOf

rdf:Property .

rdf:\_1

rdf:type

rdfs:ContainerMembershipProperty .

rdf:\_1

rdfs:domain

rdfs:Resource .

rdf:\_1

rdfs:range

rdfs:Resource .

rdf:\_2

rdf:type

rdfs:ContainerMembershipProperty .

rdf:\_2

rdfs:domain

rdfs:Resource .

rdf:\_2

rdfs:range

rdfs:Resource .

...

# Dedukčné pravidlá pre RDFS

RDFS axiómy sú vždy platné, čo možno vyjadriť dedukčným pravidlom:

***rdfsax*** \_\_\_\_\_  
                  s p o

kde **s p o** je RDFS axióma

Z RDFS axióm a sémantických podmienok vyplývajú aj ďalšie dedukčné pravidlá. Všetky sú zhrnuté v nasledujúcej tabuľke:

	<b>If S contains:</b>	<b>then S RDFS entails recognizing D:</b>
<b><i>rdfs1</i></b>	any IRI <i>aaa</i> in D	<i>aaa</i> <code>rdf:type</code> <code>rdfs:Datatype</code> .
<b><i>rdfs2</i></b>	<i>aaa</i> <code>rdfs:domain</code> <i>XXX</i> . <i>yyy</i> <i>aaa</i> <i>ZZZ</i> .	<i>yyy</i> <code>rdf:type</code> <i>XXX</i> .
<b><i>rdfs3</i></b>	<i>aaa</i> <code>rdfs:range</code> <i>XXX</i> . <i>yyy</i> <i>aaa</i> <i>ZZZ</i> .	<i>ZZZ</i> <code>rdf:type</code> <i>XXX</i> .
<b><i>rdfs4a</i></b>	<i>xxx</i> <i>aaa</i> <i>yyy</i> .	<i>XXX</i> <code>rdf:type</code> <code>rdfs:Resource</code> .
<b><i>rdfs4b</i></b>	<i>xxx</i> <i>aaa</i> <i>yyy</i> .	<i>yyy</i> <code>rdf:type</code> <code>rdfs:Resource</code> .

# Dedukčné pravidlá pre RDFS

## Pokračovanie

	If S contains:	then S RDFS entails recognizing D:
<b><i>rdfs5</i></b>	XXX <code>rdfs:subPropertyOf</code> yyy . yyy <code>rdfs:subPropertyOf</code> ZZZ .	XXX <code>rdfs:subPropertyOf</code> ZZZ .
<b><i>rdfs6</i></b>	XXX <code>rdf:type</code> <code>rdf:Property</code> .	XXX <code>rdfs:subPropertyOf</code> XXX .
<b><i>rdfs7</i></b>	aaa <code>rdfs:subPropertyOf</code> bbb . xxx aaa yyy .	xxx bbb yyy .
<b><i>rdfs8</i></b>	XXX <code>rdf:type</code> <code>rdfs:Class</code> .	XXX <code>rdfs:subClassOf</code> <code>rdfs:Resource</code> .
<b><i>rdfs9</i></b>	XXX <code>rdfs:subClassOf</code> yyy . ZZZ <code>rdf:type</code> XXX .	ZZZ <code>rdf:type</code> yyy .
<b><i>rdfs10</i></b>	XXX <code>rdf:type</code> <code>rdfs:Class</code> .	XXX <code>rdfs:subClassOf</code> XXX .
<b><i>rdfs11</i></b>	XXX <code>rdfs:subClassOf</code> yyy . yyy <code>rdfs:subClassOf</code> ZZZ .	XXX <code>rdfs:subClassOf</code> ZZZ .
<b><i>rdfs12</i></b>	XXX <code>rdf:type</code> <code>rdfs:ContainerMembershipProperty</code> .	XXX <code>rdfs:subPropertyOf</code> <code>rdfs:member</code> .
<b><i>rdfs13</i></b>	XXX <code>rdf:type</code> <code>rdfs:Datatype</code> .	XXX <code>rdfs:subClassOf</code> <code>rdfs:Literal</code> .

# Dôkazy niektorých dedukčných pravidiel

*rdfs2* ppp **rdfs:domain** xxx .  
uuu ppp vvv .  
-----  
uuu **rdf:type** xxx .

## Dôkaz:

Z pravdivosti trojice uuu ppp vvv vyplýva (podľa definície jednoduchšej interpretácie)

$$\langle I(uuu), I(vvv) \rangle \in I_{EXT}(I(ppp))$$

Rovnako z pravdivosti trojice ppp **rdfs:domain** xxx vyplýva

$$\langle I(ppp), I(xxx) \rangle \in I_{EXT}(I(\mathbf{rdfs:domain}))$$

Teda zo sémantickej podmienky pre definičný obor

$$\text{Ak } \langle p, x \rangle \in I_{EXT}(I(\mathbf{rdfs:domain})) \text{ a } \langle u, v \rangle \in I_{EXT}(p) \text{ potom } u \in IC_{EXT}(x)$$

vyplýva

$$I(uuu) \in IC_{EXT}(I(xxx))$$

A z toho vďaka sémantickej podmienke pre triedy

$$IC_{EXT}(t) = \{ \mathbf{x} : \langle \mathbf{x}, t \rangle \in I_{EXT}(I(\mathbf{rdf:type})) \}$$

dostaneme

$$\langle I(uuu), I(xxx) \rangle \in I_{EXT}(I(\mathbf{rdf:type}))$$

Čiže podľa definície jednoduchšej interpretácie je trojica

$$uuu \mathbf{rdf:type} xxx$$

pravdivá.

# Dôkazy niektorých dedukčných pravidiel

***rdfs9***   `xxx rdfs:subClassOf yyy .`  
          `aaa rdf:type xxx.`  
-----  
          `aaa rdf:type yyy.`

**Dôkaz:**

Ak `aaa rdf:type xxx` z definície  $IC_{EXT}(t) := \{ x : \langle x, t \rangle \in I_{EXT}(I(rdf:type)) \}$  vyplýva, že  $I(aaa) \in \{ x : \langle x, I(xxx) \rangle \in I_{EXT}(I(rdf:type)) \} = IC_{EXT}(I(xxx))$

Ďalej zo sémantickej podmienky

Ak  $\langle x, y \rangle \in I_{EXT}(I(rdfs:subClassOf))$  potom  $x \in IC$ ,  $y \in IC$  a  $IC_{EXT}(x) \subseteq IC_{EXT}(y)$  uplatnenej pre  $x = I(xxx)$ ,  $y = I(yyy)$  vyplýva, že  $IC_{EXT}(I(xxx)) \subseteq IC_{EXT}(I(yyy))$ .  
Teda  $I(aaa) \in \{ x : \langle x, I(yyy) \rangle \in I_{EXT}(I(rdf:type)) \} = IC_{EXT}(I(yyy))$

Čo značí, že `aaa rdf:type yyy`.

Úplne analogickým spôsobom môžeme dokázať aj ostatné pravidlá

**DÚ:** dokážate *rdfs7* a *rdfs3*

# Niektoré ďalšie vždy platné RDFS trojice

Pomocou dedukčných pravidiel a axióm je možné dokázať, že aj ďalšie trojice musia byť platné v každej RDFS interpretácii.

rdfs:Resource	rdf:type	rdfs:Class .
<b>rdfs:Class</b>	<b>rdf:type</b>	<b>rdfs:Class .</b>
rdfs:Literal	rdf:type	rdfs:Class .
rdf:XMLLiteral	rdf:type	rdfs:Class .
rdf:HTML	rdf:type	rdfs:Class .
rdfs:Datatype	rdf:type	rdfs:Class .
rdf:Seq	rdf:type	rdfs:Class .
rdf:Bag	rdf:type	rdfs:Class .
rdf:Alt	rdf:type	rdfs:Class .
rdfs:Container	rdf:type	rdfs:Class .
rdf:List	rdf:type	rdfs:Class .
rdfs:ContainerMembershipProperty	rdf:type	rdfs:Class .
rdf:Property	rdf:type	rdfs:Class .
rdf:Statement	rdf:type	rdfs:Class .

# Niektoré ďalšie vždy platné RDFS trojice

## pokračovanie

rdfs:domain	rdf:type	rdf:Property .
rdfs:range	rdf:type	rdf:Property .
rdfs:subPropertyOf	rdf:type	rdf:Property .
rdfs:subClassOf	rdf:type	rdf:Property .
rdfs:member	rdf:type	rdf:Property .
rdfs:seeAlso	rdf:type	rdf:Property .
rdfs:isDefinedBy	rdf:type	rdf:Property .
rdfs:comment	rdf:type	rdf:Property .
rdfs:label	rdf:type	rdf:Property .



# Dôkazy niektorých platných trojíc

Dôkaz pravdivosti trojice `rdfs:Class rdf:type rdfs:Class`.

Dedukčné pravidlo ***rdfs3***:

`aaa` `rdfs:range` `xxx` .

`yyy` `aaa` `zzz` .

-----  
`zzz` `rdf:type` `xxx` .

uplatníme na RDFS axiómy:

`rdfs:subClassOf`

`rdfs:Datatype`

-----  
`rdfs:Class`

`rdfs:range`

`rdfs:subClassOf`

`rdf:type`

`rdfs:Class` .

`rdfs:Class` .

`rdfs:Class` .

# RDFS dôsledok

Hovoríme, že graf E je **RDFS dôsledkom** grafu G, ak je E pravdivý vo všetkých RDFS interpretáciách, v ktorých je pravdivý G.

**Tvrdenie:** Nech graf G neobsahuje žiadne literály s dátovým typom ani jazykovo lokalizované literály, potom platí:

Graf E je RDFS-dôsledkom grafu G práve vtedy keď existuje graf G0 taký, že

- G0 môže byť odvodený z grafu G pomocou dedukčných pravidiel ***rdf1, rdfsax, rdfs1 – rdfs13***
- E je jednoduchým dôsledkom G0.

Poznámka: V materiáloch TUD je uvedené tvrdenie, ktoré nekladie podmienky na literály použité v grafoch. Na literály s typom alebo jazykovou lokalizáciou sa vzťahujú ďalšie sémantické podmienky a z nich plynú dodatočné dedukčné pravidlá. Tieto podmienky pravidlá sme vynechali pretože problematika literálov je veľmi široká a živá (zmeny aj v štandardoch). Informatívne sa dotkneme tejto témy na konci prednášky.

**Pozn.** Toto tvrdenie bohužiaľ nie je pravdivé pretože pri štandardizácii sa na niektoré dedukčné pravidlá zabudlo. Napr.:

**p rdfs:subPropertyOf r. r rdfs:range x. => p rdfs:range x.**

A podobne pre domain.

# Sémantické podmienky

## pre literály, literály s jazykovou lokalizáciou a dátové typy

Všeobecne RDFS interpretácia nemusí dávať špeciálnu sémantiku reťazcom s dátovým typom. Štandard predpisuje len podporu isté dátové typy. Pokiaľ by sme pracovali s len s dokumentami, ktoré neobsahujú reťazce s dátovým typom alebo jazykovo lokalizované nepotrebujeme využívať sémantické podmienky a axiómi, ktorými sú viazané.

Keďže ide o pomerne komplexnú problematiku a navyše štandard samotný sa v tejto oblasti stále vyvíja, nebudeme sa im v rámci RZZ podrobnejšie venovať. Len pre úplnosť zhrnieme tieto podmienky ako su uvedené vo rezii REC-rdf11-mt-20140225. Kompletnú špecifikáciu D-interpretácií nájdete v

<http://www.w3.org/TR/rdf11-mt/#literals-and-datatypes>

Nech  $D$  je množina podporovaných dátových typov **RDF interpretáciou  $I$** .

Potom pre každé IRI dátového typu  $aaa$  z  $D$  musí platiť:

- $\langle x, I(aaa) \rangle \in I_{EXT}(I(rdf:type))$  práve vtedy keď  $x$  je prvkom priestoru hodnôt  $I(aaa)$

a **RDFS interpretácia** musí spĺňať navyše:

- $LV := IC_{EXT}(I(rdfs:Literal))$  –  $LV$  je teda trieda všetkých hodnôt literálov (t.j. Všetkých reťazcov s označením typu, jazyka alebo bez nich) a  $rdfs:Literal$  je jej IRI

- $IC_{EXT}(I(rdfs:langString)) := \{I(E) : E \text{ a language-tagged string} \}$  – trieda usporiadaných dvojíc  $\langle sss, ttt' \rangle$  reťazec  $sss$ , jazykové označenie  $ttt$  konvertované na malé.

- for every other IRI  $aaa$  in  $D$ ,  $IC_{EXT}(I(aaa))$  is the value space of  $I(aaa)$

- for every IRI  $aaa$  in  $D$ ,  $I(aaa)$  is in  $IC_{EXT}(I(rdfs:Datatype))$

- If  $x$  is in  $IC_{EXT}(I(rdfs:Datatype))$

then  $\langle x, I(rdfs:Literal) \rangle$  is in  $IC_{EXT}(I(rdfs:subClassOf))$