



INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Aula prática 1

RDF

Representação e Processamento do Conhecimento

Bruno Costa, n.º 36868

Hugo Gil, n.º 37974

João Silva, n.º 42086

Docente: Paulo Trigo

Abril, 2021

1. Iniciar uma descrição RDF (grafo)

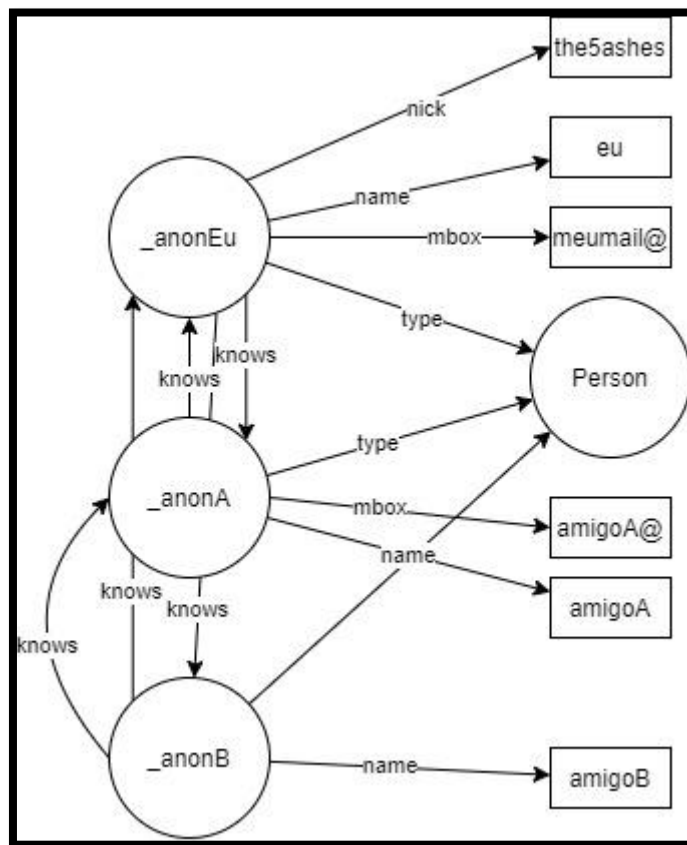


Figura 1 - Grafo FAOF

2. Associar um “namespace” aos recursos RDF

Alterações no código

```
namespace_foaf = "http://xmlns.com/foaf/0.1/"
```

Resultados

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
>
  <rdf:Description rdf:nodeID="eu">
    <rdf:type rdf:resource="http://xmlns.com/foaf/0.1/" />
    <foaf:name>o meu nome</foaf:name>
    <foaf:nick>o meu cognome</foaf:nick>
  </rdf:Description>
  <rdf:Description rdf:nodeID="amigoA">
    <foaf:name>nome do amigo A</foaf:name>
    <rdf:type rdf:resource="http://xmlns.com/foaf/0.1/" />
  </rdf:Description>
</rdf:RDF>
```

Figura 2 - Representação XML do grafo FOAF, com nós identificados por ID

3. Construir um sujeito com BNode ou qualificador[valor]

Alterações no código

```
namespace_mns = "http://meuMS/#"  
MNS = Namespace( namespace_mns )
```

Resultados

```
<?xml version="1.0" encoding="UTF-8"?>  
<rdf:RDF  
  xmlns:foaf="http://xmlns.com/foaf/0.1/"  
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"  
>  
  <rdf:Description rdf:about="http://meuMS/#amigoA">  
    <foaf:name>nome do amigo A</foaf:name>  
    <rdf:type rdf:resource="http://xmlns.com/foaf/0.1/Person"/>  
  </rdf:Description>  
  <rdf:Description rdf:about="http://meuMS/#eu">  
    <foaf:nick>o meu cognome</foaf:nick>  
    <rdf:type rdf:resource="http://xmlns.com/foaf/0.1/Person"/>  
    <foaf:name>o meu nome</foaf:name>  
  </rdf:Description>  
</rdf:RDF>
```

Figura 3 - Representação XML do grafo FOAF, com nós identificados por um NAMESPACE

4. Construir um triplo <sujeito, predicado, objecto>

Alterações no Código

```
def s_knows_o( self, s, o ):  
    p = FOAF[ "knows" ]  
    return self.grafo.add( (s, p, o) )
```

5. Usar uma ontologia já existente e descrever novos indivíduos

Alterações no Código

```
gf.grafo.add( (eu, FOAF[ "mbox" ], Literal( "meumail@" )) )  
gf.grafo.add( (amigoA, FOAF[ "mbox" ], Literal( "amigoA@" )) )  
(amigoB, _, _) = gf.adicionarPessoa( "nome do amigo B", "amigoB" )  
gf.s_knows_o( eu, amigoA )  
gf.s_knows_o( amigoA, eu )  
gf.s_knows_o( eu, amigoB )  
gf.s_knows_o( amigoB, eu )  
gf.s_knows_o( amigoB, amigoA )
```

Resultados

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
>
  <rdf:Description rdf:about="http://meuMS/#eu">
    <rdf:type rdf:resource="http://xmlns.com/foaf/0.1/Person"/>
    <foaf:Knows rdf:resource="http://meuMS/#amigoA"/>
    <foaf:name>o meu nome</foaf:name>
    <foaf:Knows rdf:resource="http://meuMS/#amigoB"/>
    <foaf:mbox>meumail@</foaf:mbox>
    <foaf:nick>o meu cognome</foaf:nick>
  </rdf:Description>
  <rdf:Description rdf:about="http://meuMS/#amigoB">
    <foaf:Knows rdf:resource="http://meuMS/#amigoA"/>
    <foaf:Knows rdf:resource="http://meuMS/#eu"/>
    <foaf:name>nome do amigo B</foaf:name>
    <rdf:type rdf:resource="http://xmlns.com/foaf/0.1/Person"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://meuMS/#amigoA">
    <foaf:Knows rdf:resource="http://meuMS/#eu"/>
    <foaf:mbox>amigoA@</foaf:mbox>
    <rdf:type rdf:resource="http://xmlns.com/foaf/0.1/Person"/>
    <foaf:name>nome do amigo A</foaf:name>
  </rdf:Description>
</rdf:RDF>
```

Figura 4 - Representação XML do grafo FOAF, com predicado "knows" e "amigoB"

6. Percorrer o grafo à procura de relações

Alterações no Código

```
for s in gf.grafo.subjects( RDF.type, FOAF[ "Person" ] ):
    for mbox in gf.grafo.objects( s, FOAF[ "mbox" ] ):
        print( "e-mail do \"%s\": \"%s\"" % (s, mbox) )

    for name in gf.grafo.objects( s, FOAF[ "name" ] ):
        print( "name do \"%s\": \"%s\"" % (s, name) )
```

Resultados

```
name do "http://meuMS/#amigoB": "nome do amigo B"
e-mail do "http://meuMS/#amigoA": "amigoA@"
name do "http://meuMS/#amigoA": "nome do amigo A"
e-mail do "http://meuMS/#eu": "meumail@"
name do "http://meuMS/#eu": "o meu nome"
```

Figura 5 - Output dos predicados nome e email dos objectos pessoa

7. Construir representações RDF (“serialize”) XML, N3 e Triples

Alterações no Código

```
gf.grafo.serialize( file, format="pretty-xml", max_depth=1 )

file = open( "foaf.nt", "wb" )
gf.grafo.serialize( file, format="nt" )
file.close()

file = open( "foaf.n3", "wb" )
gf.grafo.serialize( file, format="n3" )
file.close()
```

Resultados

```
<http://meuMS/#amigoB> <http://xmlns.com/foaf/0.1/knows> <http://meuMS/#amigoA> .
<http://meuMS/#amigoA> <http://xmlns.com/foaf/0.1/name> "nome do amigo A" .
<http://meuMS/#amigoB> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://xmlns.com/foaf/0.1/Person> .
<http://meuMS/#amigoB> <http://xmlns.com/foaf/0.1/name> "nome do amigo B" .
<http://meuMS/#amigoA> <http://xmlns.com/foaf/0.1/mbox> "amigoA@" .
<http://meuMS/#eu> <http://xmlns.com/foaf/0.1/knows> <http://meuMS/#amigoA> .
<http://meuMS/#amigoA> <http://xmlns.com/foaf/0.1/knows> <http://meuMS/#eu> .
<http://meuMS/#eu> <http://xmlns.com/foaf/0.1/knows> <http://meuMS/#amigoB> .
<http://meuMS/#amigoA> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://xmlns.com/foaf/0.1/Person> .
<http://meuMS/#amigoB> <http://xmlns.com/foaf/0.1/knows> <http://meuMS/#eu> .
<http://meuMS/#eu> <http://xmlns.com/foaf/0.1/mbox> "meumail@" .
<http://meuMS/#eu> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://xmlns.com/foaf/0.1/Person> .
<http://meuMS/#eu> <http://xmlns.com/foaf/0.1/nick> "o meu cognome" .
<http://meuMS/#eu> <http://xmlns.com/foaf/0.1/name> "o meu nome" .
```

Figura 6 - Representação em NT do Grafo FOAF

```

<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
>
  <foaf:Person rdf:about="http://meuMS/#eu">
    <foaf:name>o meu nome</foaf:name>
    <foaf:knows rdf:resource="http://meuMS/#amigoB"/>
    <foaf:mbox>meumail@</foaf:mbox>
    <foaf:nick>o meu cognome</foaf:nick>
    <foaf:knows rdf:resource="http://meuMS/#amigoA"/>
  </foaf:Person>
  <foaf:Person rdf:about="http://meuMS/#amigoB">
    <foaf:knows rdf:resource="http://meuMS/#eu"/>
    <foaf:knows rdf:resource="http://meuMS/#amigoA"/>
    <foaf:name>nome do amigo B</foaf:name>
    <foaf:mbox>amigoB@</foaf:mbox>
  </foaf:Person>
  <foaf:Person rdf:about="http://meuMS/#amigoA">
    <foaf:knows rdf:resource="http://meuMS/#eu"/>
    <foaf:knows rdf:resource="http://meuMS/#amigoB"/>
    <foaf:name>nome do amigo A</foaf:name>
    <foaf:mbox>amigoA@</foaf:mbox>
  </foaf:Person>
</rdf:RDF>

```

Figura 7 - Representação XML do Grafo FOAF

```

@prefix foaf: <http://xmlns.com/foaf/0.1/> .

<http://meuMS/#amigoB> a foaf:Person ;
  foaf:knows <http://meuMS/#amigoA>,
    <http://meuMS/#eu> ;
  foaf:name "nome do amigo B" .

<http://meuMS/#amigoA> a foaf:Person ;
  foaf:knows <http://meuMS/#eu> ;
  foaf:mbox "amigoA@" ;
  foaf:name "nome do amigo A" .

<http://meuMS/#eu> a foaf:Person ;
  foaf:knows <http://meuMS/#amigoA>,
    <http://meuMS/#amigoB> ;
  foaf:mbox "meumail@" ;
  foaf:name "o meu nome" ;
  foaf:nick "o meu cognome" .

```

Figura 8 - Representação N3 do Grafo FOAF

8. Reconstruções de Grafos a partir de RDF

Fui testado a reconstrução de grafo utilizando o ficheiro `02_interrogarRDF_grafo.py` e verificamos que em qualquer dos formatos utilizados (.xml, .n3 e .nt) o grafo tem a mesma informação, obtendo o seguinte retorno.

```
Conhecidos mutuos: '5'  
knows( 'nome do amigo A', 'o meu nome' )  
knows( 'o meu nome', 'nome do amigo B' )  
knows( 'nome do amigo B', 'o meu nome' )  
knows( 'o meu nome', 'nome do amigo A' )  
knows( 'nome do amigo B', 'nome do amigo A' )
```

Figura 9 - Output do Grafo FOAF, com conhecidos

9. Construir percursos sobre o grafo RDF

Alterações no Código

```
for (s, o) in lista_s_o:  
    if lista_s_o.count((o, s)) == 1:  
        resultado.append( (s, o) )  
return resultado
```

Resultados

```
Conhecidos mutuos: '4'  
knows( 'nome do amigo A', 'o meu nome' )  
knows( 'nome do amigo B', 'o meu nome' )  
knows( 'o meu nome', 'nome do amigo A' )  
knows( 'o meu nome', 'nome do amigo B' )
```

Figura 10 - Output do Grafo FOAF, com conhecidos mutuos