XML/RDF Query By Example

- http://dyomedea.com/papers/2006-XMP-Prague/
- XML Prague 2006

– Eric van der Vlist, Dyomedea (vdv@dyomedea.com)

Context

- LDAP as a tree

```
<?xml version="1.0" encoding="utf-8"?>
<annuaire>
 <fr>
  <insee>
   <Personnes>
    <inseePerson
dn="uid=R3D2,ou=Personnes,o=insee,c=fr">
     <telephoneNumber>0123456789</telephoneNumber>
     <cn>Laurent Dupondt</cn>
```

LDAP as a graph

```
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
----------------------------
 <inseePerson
rdf:about="http://xml.insee.fr/ldap/uid=R2D2,ou=Personn
es,o=insee,c=fr">
  <l:dn>uid=R2D2,ou=Personnes,o=insee,c=fr</l:dn>
  <l:parent
rdf:resource="http://xml.insee.fr/ldap/ou=Personnes,o=i
nsee,c=fr"/>
----------------------------
  <cn>Laurent Dupondt</cn>
  <inseeFonction
rdf:resource="http://xml.insee.fr/ldap/uid=GS10,ou=Fonc
tions,o=insee,c=fr"/>
```

LDAP as a graph AND as a tree

```
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
 xmlns="http://xml.insee.intra/schema/annuaire/"
xmlns:a="http://xml.insee.intra/schema/annuaire/"
 xmlns:l="http://xml.insee.intra/schema/ldap/">
 <l:root>
  <l:node rdf:parseType="Resource">
   <c>fr</c>
   <l:node rdf:parseType="Resource">
    <o>insee</o>
    <l:node rdf:parseType="Resource">
     <ou>Personnes</ou>
     <inseePerson rdf:parseType="Resource">
      <telephoneNumber>0123456789</telephoneNumber>
```

Problem and candidates

- Low RDF tax RDF/XML Query language
- LDAP filters (RFC 2254)

```
<?xml version="1.0" encoding="utf-8"?>
<ldapSearch>
  <base scope="subTree">Personnes,o=insee,c=fr</base>
  <filter>(inseeRoleApplicatif=RP$$P$SUP*)</filter>
</ldapSearch>
```

W3C XQuery

```
xquery version "1.0";
declare namespace a =
"http://xml.insee.intra/schema/annuaire/";
<rdf:RDF xmlns:rdf="
http://www.w3.org/1999/02/22-rdf-syntax-ns#">
{
   for $x in /rdf:RDF/a:inseePerson
   where $x[a:inseeRoleApplicatif[ starts-with(.,
'RP$$P$SUP') ]]
   return $x }
</rdf:RDF>
```

W3C SPARQL (or any other RDF query language)

```
PREFIX : http://xml.insee.intra/schema/annuaire/
SELECT ?inseePerson
WHERE {
   ?inseePerson rdf:type :inseePerson;
   ?inseePerson :inseeRoleApplicatif ?
inseeRoleApplicatif.
   FILTER regex(str(?inseeRoleApplicatif), "^RP$$P$SUP
")

}
```

Home bred

Query By Example (QBE)

- Basic query structure

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
    xmlns="http://xml.insee.intra/schema/annuaire/"
    xmlns:q="http://xml.insee.intra/schema/qbe/">
    <q:select>
        <q:what rdf:resource="#what"/>
        <q:where rdf:resource="#where"/>
    </q:select>
    <inseePerson rdf:ID="what"/>
    <inseePerson rdf:ID="where">
        <mail>jean.dupondt@insee.fr</mail>
    </inseePerson>
</rdf:RDF>
```

Omitted what clause

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
    xmlns="http://xml.insee.intra/schema/annuaire/"
    xmlns:q="http://xml.insee.intra/schema/qbe/">
    <q:select>
        <q:where rdf:resource="#where"/>
    </q:select>
    <inseePerson rdf:ID="where">
        <mail>jean.dupondt@insee.fr</mail>
    </inseePerson>
</rdf:RDF>
```

Russian doll

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
    xmlns="http://xml.insee.intra/schema/annuaire/"
    xmlns:q="http://xml.insee.intra/schema/qbe/">
    <q:select>
        <q:where>
            <inseePerson>
                <mail>jean.dupondt@insee.fr</mail>
            </inseePerson>
        </q:where>
    </q:select>
</rdf:RDF>
```

Introducing functions

- What we've not done

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
    xmlns="http://xml.insee.intra/schema/annuaire/"
    xmlns:q="http://xml.insee.intra/schema/qbe/">
    <q:select>
        <q:where>
            <inseePerson>
                <mail>ends-with(@insee.fr)</mail>
            </inseePerson>
        </q:where>
    </q:select>
</rdf:RDF>
```

What we've almost done

```
        <q:where>
            <inseePerson>
                <mail>
                    <q:ends-with>@insee.fr</q:ends-
with>
                </mail>
            </inseePerson>
        </q:where>
--------------------------
_:jA3 <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://xml.insee.intra/schema/qbe/ends-with> .
Error: file:///home/vdv/cvs-
private/presentations/en/extreme2005/query4.rdf[11:43]:
{E202} Expected whitespace found: '@insee.fr'. Maybe a
missing rdf:parseType='Literal', or a striping problem.
```

Looking back at RDF

```
<rdf:RDF>
  <foo>
   <bar>
    <baz>
      <bat>XXX</bat>
    </baz>
  </bar>
 </foo>
<rdf:RDF>
---------------------------
Subj.  Pred.   Obj.
<foo> <bar> <baz>
<baz> <bat> "XXX"
```

What we've not done (again)

```
        <q:where>
            <inseePerson>
                <mail rdf:parseType="Resource">
                    <q:ends-with>@insee.fr</q:ends-
with>
                </mail>
            </inseePerson>
        </q:where>
```

What we've done (finally)

```
            <inseePerson>
                <mail>
                    <q:conditions>
                        <q:ends-with>@insee.fr</q:ends-
with>
                    </q:conditions>
                </mail>
            </inseePerson>
```

Joins

```
    <insee:Person>
        <inseeUnite rdf:resource="xxx"/>
    </insee:Person>
    <inseeUnite rdf:about="xxx">
        <ou>DG75-C460</ou>
    </inseeUnite>
--------------------------
    <insee:Person>
        <inseeUnite rdf:parseType="Resource">
            <ou>DG75-C460</ou>
        </inseeUnite>
    </insee:Person>
```

Joins in queries

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
    xmlns="http://xml.insee.intra/schema/annuaire/"
    xmlns:q="http://xml.insee.intra/schema/qbe/">
    <q:select>
        <q:where>
            <inseePerson>
                <inseeUnite rdf:parseType="Resource">
                    <ou>DG75-C460</ou>
                </inseeUnite>
            </inseePerson>
        </q:where>
    </q:select>
</rdf:RDF>
```

Join and condition

```
        <q:where>
            <inseePerson>
                <inseeUnite rdf:parseType="Resource">
                    <ou>
                        <q:conditions>
                            <q:starts-
with>DG75</q:starts-with>
                        </q:conditions>
                    </ou>
                </inseeUnite>
            </inseePerson>
        </q:where>
```

What about several conditions?

- By default: and

```
<q:where>
    <inseePerson>
        <cn>
            <q:conditions>
                <q:contains>a</q:contains>
            </q:conditions>
        </cn>
        <mail>
            <q:conditions>
                <q:contains>o</q:contains>
            </q:conditions>
        </mail>
    </inseePerson>
</q:where>
```

More verbose for other cases

```
<inseePerson>
   <q:if>
      <q:all>
          <cn>
              <q:conditions>
                  <q:contains>a</q:contains>
               </q:conditions>
          </cn>
          <mail>
              <q:conditions>
                  <q:contains>o</q:contains>
              </q:conditions>
          </mail>
      </q:all>
  </q:if>
</inseePerson>
```

Search base

- Additions to the RDF model

```
  <l:dn>uid=R2D2,ou=Personnes,o=insee,c=fr</l:dn>
  <l:parent
rdf:resource="http://xml.insee.fr/ldap/ou=Personnes,o=i
nsee,c=fr"/>
  <l:ancestorOrSelf
rdf:resource="http://xml.insee.fr/ldap/uid=R2D2,ou=Pers
onnes,o=insee,c=fr"/>
  <l:ancestorOrSelf
rdf:resource="http://xml.insee.fr/ldap/ou=Personnes,o=i
nsee,c=fr"/>
  <l:ancestorOrSelf
rdf:resource="http://xml.insee.fr/ldap/o=insee,c=fr"/>
```

LDAP “subtree”

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
    xmlns:q="http://xml.insee.intra/schema/qbe/"
    xmlns="http://xml.insee.intra/schema/annuaire/"
    xmlns:l="http://xml.insee.intra/schema/ldap/">
    <q:select>
        <q:where>
            <inseePerson>
                <l:ancestorOrSelf
rdf:parseType="Resource">
                    <l:dn>o=insee,c=fr</l:dn>
                </l:ancestorOrSelf>
            </inseePerson>
        </q:where>
    </q:select>
</rdf:RDF>
```

LDAP “one level”

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
    xmlns:q="http://xml.insee.intra/schema/qbe/"
    xmlns="http://xml.insee.intra/schema/annuaire/"
    xmlns:l="http://xml.insee.intra/schema/ldap/">
    <q:select>
        <q:where>
            <inseePerson>
                <l:parent rdf:parseType="Resource">
                    <l:dn>o=insee,c=fr</l:dn>
                </l:parent>
            </inseePerson>
        </q:where>
    </q:select>
</rdf:RDF>
```

LDAP “base”

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
    xmlns:q="http://xml.insee.intra/schema/qbe/"
    xmlns="http://xml.insee.intra/schema/annuaire/"
    xmlns:l="http://xml.insee.intra/schema/ldap/">
    <q:select>
        <q:where>
            <inseePerson>
                <l:dn>o=insee,c=fr</l:dn>
            </inseePerson>
        </q:where>
    </q:select>
</rdf:RDF>
```

Current status

- Proof of concept
- Final implementation in progress

Conclusion

- This could be generalized for cases where:
  - **The source to query is expressed (or can be expressed) as RDF/XML.**
  - **The query language needs to make sense for both XML and RDF heads.**
  - **The query language needs to look like the source to query (QBE).**