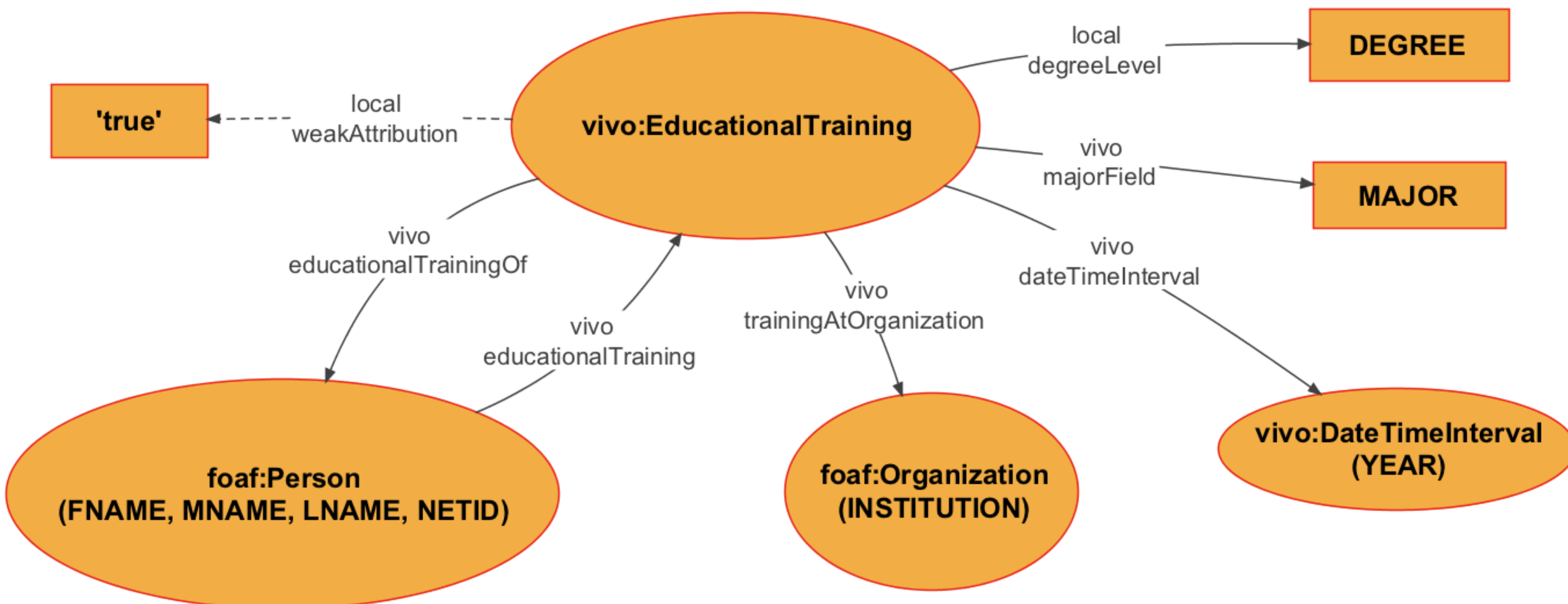


Abstract

The purpose of this example is to illustrate XSLT based techniques that have been used successfully to ingest data from more than a dozen sources into VIVO at Cornell. Instead of a simplified 'toy' example, the source data used will display many of the data quality problems often found in practice. The goal is to transform this source data into RDF that conforms to a specific data model and can be loaded into a VIVO instance. The example is based on educational credentials and the central objects in the RDF data model are instances of the class **vivo:EducationalTraining**. In addition, we want to prevent duplication of Person and Organization RDF. Experience has taught us that this XSLT transform methodology performs well in terms of processing time and is scalable to tens of thousands of source data records.

Tutorial and full download on VIVO wiki at <https://wiki.duraspace.org/x/4dwQAq>

Target RDF Classes and Properties



Sample input data

```

<ROWS>
<ROW id="1011001">
<NETID>dag065</NETID>
<FNAME>David</FNAME>
<MNAME>Augustus</MNAME>
<LNAME>Green</LNAME>
<DEGREE>Master</DEGREE>
<YEAR>1985</YEAR>
<INSTITUTION>University of Kansas</INSTITUTION>
<MAJOR>Oriental Art History</MAJOR>
<MINOR></MINOR>
<LAST_UPDATED>2012-10-24 13:10:59.0</LAST_UPDATED>
</ROW>
<ROW id="1011002">
<NETID>dah3507</NETID>
<FNAME>Don</FNAME>
<MNAME>A</MNAME>
<LNAME>Horsham</LNAME>
<DEGREE>Master</DEGREE>
<YEAR>2005</YEAR>
<INSTITUTION>Cornell University</INSTITUTION>
<MAJOR></MAJOR>
<MINOR></MINOR>
<LAST_UPDATED>2012-08-29 17:09:42.0</LAST_UPDATED>
</ROW>
    
```

Existing persons

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX hr: <http://vivo.cornell.edu/ns/hr/0.9/hr.owl#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX vivo: <http://vivoweb.org/ontology/core#>
PREFIX vivoc: <http://vivo.library.cornell.edu/ns/0.1#>
PREFIX aka: <http://vivoweb.org/ontology/aka#>

construct {
  ?personURI rdfs:label ?personLabel .
  ?personURI hr:netId ?netId .
  ?personURI foaf:firstName ?firstName .
  ?personURI vivo:middleName ?middleName .
  ?personURI foaf:lastName ?lastName .
  ?personURI aka:nameParts ?akaParts .
} where {
  ?personURI a foaf:Person .
  ?personURI foaf:lastName ?lastName .
  ?personURI foaf:firstName ?firstName .
}
    
```

Typical data issues

```

<ROW id="1011004">
<NETID>aweb10</NETID>
<FNAME>Denise</FNAME>
<MNAME>H</MNAME>
<LNAME>Valencia</LNAME>
<DEGREE>Master</DEGREE>
<YEAR>1992</YEAR>
<INSTITUTION>SUNY Oswego</INSTITUTION>
<MAJOR>Vocational Technical Education</MAJOR>
<MINOR></MINOR>
<LAST_UPDATED>2012-07-03 13:56:40.0</LAST_UPDATED>
</ROW>
<ROW id="1011005">
<NETID></NETID>
<FNAME>Denise</FNAME>
<MNAME>Westmore</MNAME>
<LNAME>Valencia</LNAME>
<DEGREE>Bachelor</DEGREE>
<YEAR>1988</YEAR>
<INSTITUTION>Cortland</INSTITUTION>
<MAJOR>Education</MAJOR>
<MINOR></MINOR>
<LAST_UPDATED>2012-07-03 13:56:40.0</LAST_UPDATED>
</ROW>
<ROW id="1011006">
<NETID>daa065</NETID>
    
```

Mapping URIs onto a group of groups

