Getty Vocabularies - Loading and Publishing Technical Update

GREGG GARCIA / SOFTWARE ARCHITECT / J. PAUL GETTY TRUST
AAT Translation Activity  
(since last ITWG)

Dutch AAT  
- yearly update routines developed (adLib XML format)  
- updates to complete translation December, 2014 and February, 2016  
- 2990 newly translated records loaded May, 2015  
- 222 new AAT candidates loaded March, 2015  
- 146 new AAT candidates loaded December, 2014  
- 219 new AAT candidates in-progress

Chinese AAT  
- 4774 newly translated records loaded January, 2015

German AAT  
- 2205 newly translated records loaded October, 2015  
- new sources added March, 2016  
- 5298 newly translated records in-progress
Other data loads

TGN Countries loaded
- China (636995), Pakistan (158027), Ireland (23908), Cyprus/Dekelia (6298), Australia (107074), Portugal (23725), Turkey (71725), France (73426)

ULAN
- 2439 artists and 12008 owner loaded from Getty Provenance
- 400 records loaded from online contribution form
LOD Activity

- TGN launched in LOD format February, 2015
- ULAN launched in LOD format August, 2015
- CONA LOD development 80% complete
- Getty Museum LOD development in-progress
- Research using Sparql endpoint for web search
Using Sparql Endpoint for Online Search
AAT Search

Term Search:

(lcase(REPLACE(REPLACE(str(?pTerm), '<', ''), '>', ''))) AS ?pTermSort
(lcase(REPLACE(REPLACE(str(?mTerm), '<', ''), '>', ''))) AS ?mTermNorm
{
?Termuri luc:term 'rhyta'; a ?typ; xl:literalForm ?mTerm
filter (?typ = xl:Label)
}
ORDER BY ?pTermSort ?Suburi

Add Note Search:

(lcase(REPLACE(REPLACE(str(?pTerm), '<', ''), '>', ''))) AS ?pTermSort
(lcase(REPLACE(REPLACE(str(?mTerm), '<', ''), '>', ''))) AS ?mTermNorm
{
?Termuri luc:term 'rhyta'; a ?typ; xl:literalForm ?mTerm filter(?typ = xl:Label)
?Scopeuri luc:text 'vessel'; a ?typ2 filter(?typ2 = gvp:ScopeNote)
gvp:parentStringAbbrev ?Parents
}
ORDER BY ?pTermSort ?Suburi
Lessons Learned
AAT Search

- Lucene index does not handle stemming

- Could not replicate current search functionality without “extra” keywords generated during the publish:

Term “bow-back” generates “bowback bow-back bow back” all as keywords
ULAN Search

Name search:

(coalesce(?engTerm,?pTerm) as ?engTermFinal)
(lcase(REPLACE(REPLACE(str(?mTerm), '<', ''), '>', '')) AS ?mTermNorm)
(lcase(REPLACE(REPLACE(str(?pTerm), '<', ''), '>', '')) AS ?pTermSort)
(lcase(REPLACE(REPLACE(str(?engTerm), '<', ''), '>', '')) AS ?engTermSort)
(coalesce(?engTermSort,?pTermSort) as ?TermSort)
{
?Termuri luc:term 'bacon'; a ?typ; xl:literalForm ?mTerm filter (?typ = xl:Label)
optional {?Suburi xl:prefLabel [ dcterms:language aat:300388277; xl:literalForm ?engTerm ]}
} ORDER BY ?pTermSort ?Suburi

Add Role:
?Suburi gvp:agentType ?role filter(?role=aat:300025136)

Add Nationality:
TGN Search

Place Search:
(coalesce(?engTerm,?pTerm) as ?engTermFinal)
(lcase(REPLACE(REPLACE(str(?mTerm), '<', ""), '>', "")) AS ?mTermNorm)
(lcase(REPLACE(REPLACE(str(?pTerm), '<', ""), '>', "")) AS ?pTermSort)
(lcase(REPLACE(REPLACE(str(?engTerm), '<', ""), '>', "")) AS ?engTermSort)
(coalesce(?engTermSort,?pTermSort) as ?TermSort)
{
?Termuri luc:term 'paris'; a ?typ; xl:literalForm ?mTerm filter (?typ = xl:Label)
} ORDER BY ?TermSort ?Suburi

Add Nation:

Add Place Type:
?Suburi gvp:placeType ?ptype filter(?ptype=aat:300387219)
Lessons Learned
TGN and ULAN Search

- LOD searching worked great!
- Very easy to cross-reference and add to search linked fields like role, nationality, and place type
- No need for matching compound words and stemming with Names/Places

But ..... 
- Normalization missing from keywords
- Search could not find Names/Places with diacritics unless exact diacritics were used in the search
LOD Search Conclusion

- Because the LOD search resulted in less functionality than currently exits with the Vocabularies website, we reverted back to using RDBMS text index

- Unlike RDBS, the LOD data set does not have hidden fields that can store expanded search keywords

- Future search implementations could focus on using systems such as Solr or Elasticsearch which could contain the expanded keywords without exposing them to the user
Basic Sparql Queries

Get all nations:
select * {?s gvp:placeType aat:300128207; gvp:prefLabelGVP/xl:literalForm ?pTerm}

Get count of subjects having a Chinese term:

Get count of subjects having a Chinese term in AAT: