Invenio Knowledge Bases

Tibor Šimko
<tibor.simko@cern.ch>

Department of Information Technology
CERN

Invenio User Group Workshop 2012
CERN, May 7–9 2012

Overview of Knowledge Base Types

- **KBA**: authority like KB
  - controlled vocabulary
  - use case: list of allowed subjects
- **KBR**: reference like KB
  - `bad1--good`
    `bad2--good`
  - use case: synonyms for searching and indexing
- **KBT**: taxonomy like KB
  - RDF SKOS
  - use case: keyword classification
- **KBD**: dynamic like KB
  - using search functions via KB API
  - use case: advanced auto-suggestion facility (typing “Geneva” yielding “CERN” before “University of Geneva” as per article count)

Knowledge Base Usage

Providers:

- text files
  - KBA, KBR
- BibKnowledge
  - KBA, KBR, KBT reference, KBD definition

Users:

- WebSubmit
- BibConvert
- BibFormat
- BibEdit
- BibClassify
- BibCheck
- BibIndex

Text Files vs BibKnowledge

- old technique: KBA, KBR living as text files
- new technique: living in tables under **BibKnowledge**

```
mysql> SELECT id,name,kbtype FROM knwKB LIMIT 2;

| id | name            | kbtype |
+----+-----------------+--------+
|  1 | DBCOLLID2COLL   | NULL   |
|  2 | EJOURNALS       | NULL   |

mysql> SELECT * FROM knwKBRVAL LIMIT 2;

| id | m_key     | m_value              | id_knwKB |
+----+-----------+----------------------+----------+
|  1 | ARTICLE   | Published Article    |        1 |
|  2 | PREPRINT  | Preprint             |        1 |
```

- import/export

`http://localhost/kb/export?kbname=INDEX-SYNONYM-TITLE&format=jquery`

Use case: WebSubmit/BibConvert

- **KBR** used for “input value” $\rightarrow$ “output value” replacement

```
KB(kb_file, kb_mode)     -     kb_file search

1 - case sensitive / match (default)
2 - not case sensitive / search
3 - case sensitive / search
4 - not case sensitive / match
5 - case sensitive / search (in KB)
6 - not case sensitive / search (in KB)
7 - case sensitive / search (reciprocal)
8 - not case sensitive / search (reciprocal)
9 - replace by _DEFAULT_ only
R - not case sensitive / search (reciprocal) replace
```

Use case: BibFormat

- use of **KBR** in action:
  - `DBCOLLID2COLL`
    collection identifiers and navigation trail detection
  - `DBCOLLID2BIBTEX`
    collection identifiers and BibTeX entry type

Use case: BibEdit

- **KBD** for simple auto suggestion
  (type “ell”, get “Ellis, J”)
- **KBD** for advanced auto suggestion
  (type “gene”, get “CERN” before “University of Geneva”)

Use case: BibClassify

- using **KBT** for ontology

```
<Concept rdf:about="http://cern.ch/thesauri/HEPontology.rdf#activityreport">
  <prefLabel xml:lang="en">activity report</prefLabel>
  <altLabel xml:lang="en">status report</altLabel>
  <related rdf:resource="http://cern.ch/thesauri/HEPontology.rdf#review"/>
</Concept>
```

- bibclassify invocation:

```
sudo -u www-data /opt/invenio/bin/bibclassify \
     -k HEP /tmp/0101001.pdf
```

Use case: BibCheck KBA

- using **KBA** for record checking and auto-correction
  - `check-field-subfield-content-via-kba`
  - `check-field-transform-subfield-content-unless-kba`

```
;;; check-field-subfield-content-via-kba f-tag sf-code kba-filename
;;
;;   Check the content of field tag f-tag and subfield-code sf-code by
;;   comparing the content to the lines of the authority knowledge base
;;   file kba-file. (i.e. of the form ``good'')
;;   1) When no f-tag field is present in the record, nothing happens.
;;   2) If the field is present and matches some KB value, then nothing happens.
;;   3) If no match in KB was found, then the incident will be reported.

(check-field-subfield-content-via-kba
 ("041" $$a "~/private/src/nchkall/kbs/SISC-lang.kba")
 ("65017" $$a "~/private/src/nchkall/kbs/SISC-su.kba")
 ("693" $$a "~/private/src/nchkall/kbs/SISC-ac.kba")
 ("693" $$e "~/private/src/nchkall/kbs/SISC-ex.kba"))
```

Use case: BibCheck KBR

- using **KBR** for record checking and auto-correction
  - `check-field-replace-subfield-content-via-kbr`
  - `check-field-replace-subfield-content-strings-from-kbr`
  - `check-field-replace-field-content-via-kbrs`

```
;;; check-field-replace-subfield-content-via-kbr f-tag sf-code kbr-file action-when-not
;;
;;   Check the content of field tag f-tag and subfield-code sf-code by
;;   comparing the content to the keys of referential knowledge base file
;;   kbr-file. (i.e. of the form ``bad---good'')
;;   1) When no f-tag field is present in the record, nothing happens.
;;   2) If the field is present and matches some KB value, then correct these fields.
;;   3) If no match in KB was found, then behaviour depends on action-when-not-found:
;;       a) 'REPORT' means that the incident will be reported.
;;       b) 'IGNORE' means that nothing happens.
;;       c) 'ADD' means to add value-to-add-when-not-found, that is inserted at the
;;          beginning of the field.

(check-field-replace-subfield-content-via-kbr
 ("041" $$a "~/private/src/nchkall/kbs/SISUC-lang.kb" ignore "")
 ("260" $$a "~/private/src/nchkall/kbs/SISC-implace.kbr" ignore "")
 ("260" $$b "~/private/src/nchkall/kbs/SISUC-univ.kb" ignore "")
 ("65017" $$a "~/private/src/nchkall/kbs/SISC-su.kbr" ignore "XX")
 ("693" $$a "~/private/src/nchkall/kbs/SISUC-ac.kb" ignore "Accelerator? ")
 ("693" $$e "~/private/src/nchkall/kbs/SISUC-ex.kbr" ignore ""))
```

Use case: BibIndex

- **KBR** for search-time synonym expansion

```
CFG_WEBSEARCH_SYNONYM_KBRS = {
  'journal': ['SEARCH-SYNONYM-JOURNAL',
              'leading_to_number'],
}
```

- **KBR** for index-time synonym generation

```
CFG_BIBINDEX_SYNONYM_KBRS = {
  'global': ['INDEX-SYNONYM-TITLE', 'exact'],
  'title': ['INDEX-SYNONYM-TITLE', 'exact'],
}
```