Things you can do dumping your Invenio database into a flat file

Invenio User Group Workshop 2017
Heinz Maier-Leibnitz Zentrum (MLZ)
21-24 March 2017

Ferran Jorba
Universitat Autònoma de Barcelona
Ferran.Jorba@uab.cat

Summary

- Goal: easing database maintenance tasks.
- Automatic record cleaning.
- Automatic record enrichment.
- The tool.
- Strategies to search the flat file.
- Questions?

Search (librarian's style)

Examples:

- Records without a value (ex: rights, fulltext, DOI, etc.).
- All values for a subfield (ex: 980 $a, journal names, etc.).
- Records per year of publication.
- Check URLs.
- List of possible authors matching ORCIDs.
- Authors without affiliation.

Search & replace (librarian's style)

- Add subfields automatically (ex.: language note from language code).
- Normalise dates (ex: ?, -, X, etc.).
- Normalise Marc indicators.
- Add PubMedID to records.
- Add ORCID to authors.
- Normalise typographic characters.
- Delete empty subfields.

Evolution of a solution

1. Simplest solution: dumping web output
    a. Small database.
    b. Newbie Invenio admin.
2. Moderate solution: use of Invenio API
    a. Moderate database.
    b. Sophisticated wannabe Invenio admin.
3. Current solution: cache the results
    a. Larger database: simpler solutions *just don't work*.
    b. Build a better and stable workflow.

First run may take some time...

```
ifmuc@taltabull:/tmp$ marcdump.py ifmuc.db
ifmuc.db database does not exist.  Creating...
Updating 10358 records...
100 of 10358 records updated. Remaining time: 13m40s (0.08 seconds per record)
200 of 10358 records updated. Remaining time:13m32s (0.08 seconds per record)
300 of 10358 records updated. Remaining time: 13m24s (0.08 seconds per record)
400 of 10358 records updated. Remaining time: 13m16s (0.08 seconds per record)
500 of 10358 records updated. Remaining time: 13m8s (0.08 seconds per record)

ddd@taltabull:/tmp$ marcdump.py ddd.db
ddd.db database does not exist.  Creating...
Updating 150162 records...
100 of 150162 records updated. Remaining time: 7h5m10s (0.17 seconds per record)
200 of 150162 records updated. Remaining time:6h14m54s (0.15 seconds per record)
300 of 150162 records updated. Remaining time: 6h14m39s (0.15 seconds per record)
400 of 150162 records updated. Remaining time: 6h14m24s (0.15 seconds per record)
500 of 150162 records updated. Remaining time: 6h14m9s (0.15 seconds per record)
```

But next times, it take seconds

- It uses Invenio API to search from last modified time:

```
since_mtime = os.path.getmtime(dbname)
[...]
recids = perform_request_search(
                        dt='m', d1y=year, d1m=month, d1d=day)
```

- Dumping from SQLite to flat file is also very fast.

How do we use it at UAB

- Daily dump, for each database.
- Daily git commit.
- Expose the flat file via web address.
- Input file for most daily (12+) maintenance jobs.
- Sometimes, librarians download the file and use it directly (~250 Mb, using Notepad++).

Maintenance scripts

Two pass strategy:

1. First, read the file and get a list of candidate recids for each job (a few seconds).
2. Then, read real records from the database via Invenio API, as they may have changed, and do the work.

Create a valid MarcXML output file to be uploaded via bibupload (-a or -r).

Questions?

https://github.com/fjorba/marcdump