



Samuele Kaplun  
*1st Invenio User  
Workshop 2012*

Invenio RobotUpload

# RobotUpload

- PUSH paradigm (V.s. OAI-PMH PULL paradigm)
- To **POST** (the HTTP way) a record
- Insert/replace/modify/replace
- Upload files via FFT (see talk on BibDocFile)
- BibUpload wrapper



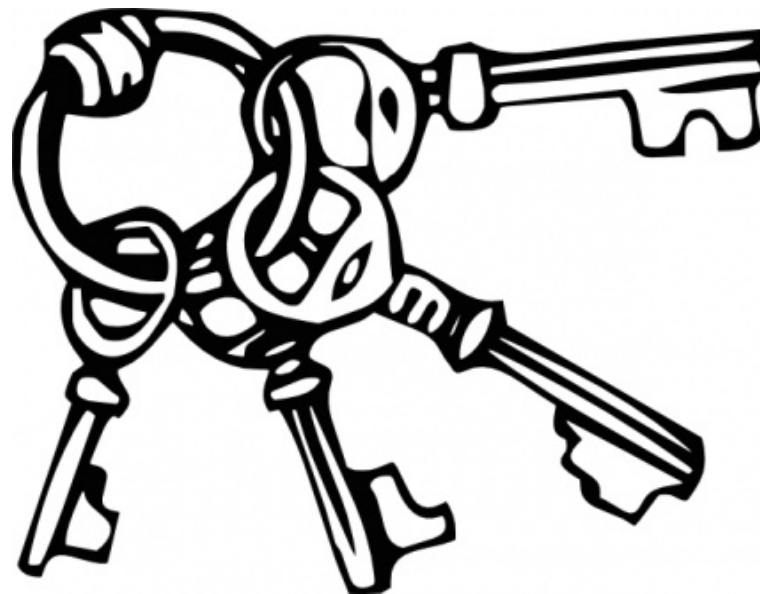
# Usage at CERN



- CMS experiment pushes record automatically to CDS
- Micala software pushes talks held at CERN in CDS automatically
- EDMS (engineering document repository at CERN) will soon push ALICE experiment documents to CDS
- CERN Drupal web infrastructure to push photos and other documents to CDS

# Authorization

- *IP address* based (in 1.0 and newer)
  - Specific IP address ranges authorized to specific collections
- *API key a la Twitter* (in 1.1 and newer)



# Specifications

- *multipart/form-data* (1.0):
  - MARCXML as a form field
  - e.g.: `curl -F 'file=@localfile.xml' -F 'mode=-i' http://cdsweb.cern.ch/batchuploader/robotupload -A invenio_webupload`
- REST-like (1.1):
  - MARXML as body of the HTTP request
  - easier client creation e.g. in PL/SQL
  - e.g.: `curl -T localfile.xml http://cdsweb.cern.ch/batchuploader/robotupload/insert -A invenio_webupload -H "Content-Type: application/marcxml+xml"`

# Feedback

- Immediate HTTP error code in case of non valid MARCXML or other request error
- From 1.1, calls back a given URL, POSTing JSON-encoded results



# Feedback

```
{
  "nonce": "1234",
  "results": [
    {
      "recid": -1,
      "error_message": "ERROR: can not retrieve the record identifier",
      "success": false
    },
    {
      "recid": 1000,
      "error_message": "",
      "success": true,
      "marcxml": "1000...",
      "url": "http://www.example.org/record/1000"
    },
    ...
  ]
}
```

# Tips & Tricks

- Use FFT (see talk on BibDocFile) to push also digital objects with the metadata
  - *The Invenio server need to be authorized to download the digital object*
- Use the *nonce* (1.1) argument to map multiple request to their multiple feedbacks
- Use the feedback to discover allocated record IDs...
- ... or preallocate them with the specific *allocaterecord* API





# Summary

- Simple PUSH interface
- REST friendly
- Directly wraps BibUpload
- More info at:

**<http://.../help/admin/bibupload-admin-guide#4.2>**