The PKP XML Service and meTypeset: a new Word-to-JATS solution

Alex Garnett
Digital Preservation & Data Curation Librarian
Juan Pablo Alperin
John Willinsky
Simon Fraser University

The Public Knowledge Project

- Developers of Open Journal Systems and Open Monograph Press
- Based at Stanford University and Simon Fraser University
- Journal hosting services and editorial training initiatives for Global South

XML and Us

- OJS is review and publication-focused, not document-focused
- The handful of OJS journals that do PMC deposit have XML contractors; the others only create PDF

Supervised Solutions

- Word plugin for authoring XML
- Lemon-8 XML
- Manual markup (OxygenXML / LaTeX / other)
  –> Outsourcing
- **Probably most of the other presentations at this conference**

Unsupervised Solutions

- OxGarage
- Word/LibreOffice HTML output (oy)
- Word/LibreOffice PDF output (remarkably popular!)
- Pandoc (nice development, not production-quality)

Perennial Existential Crisis

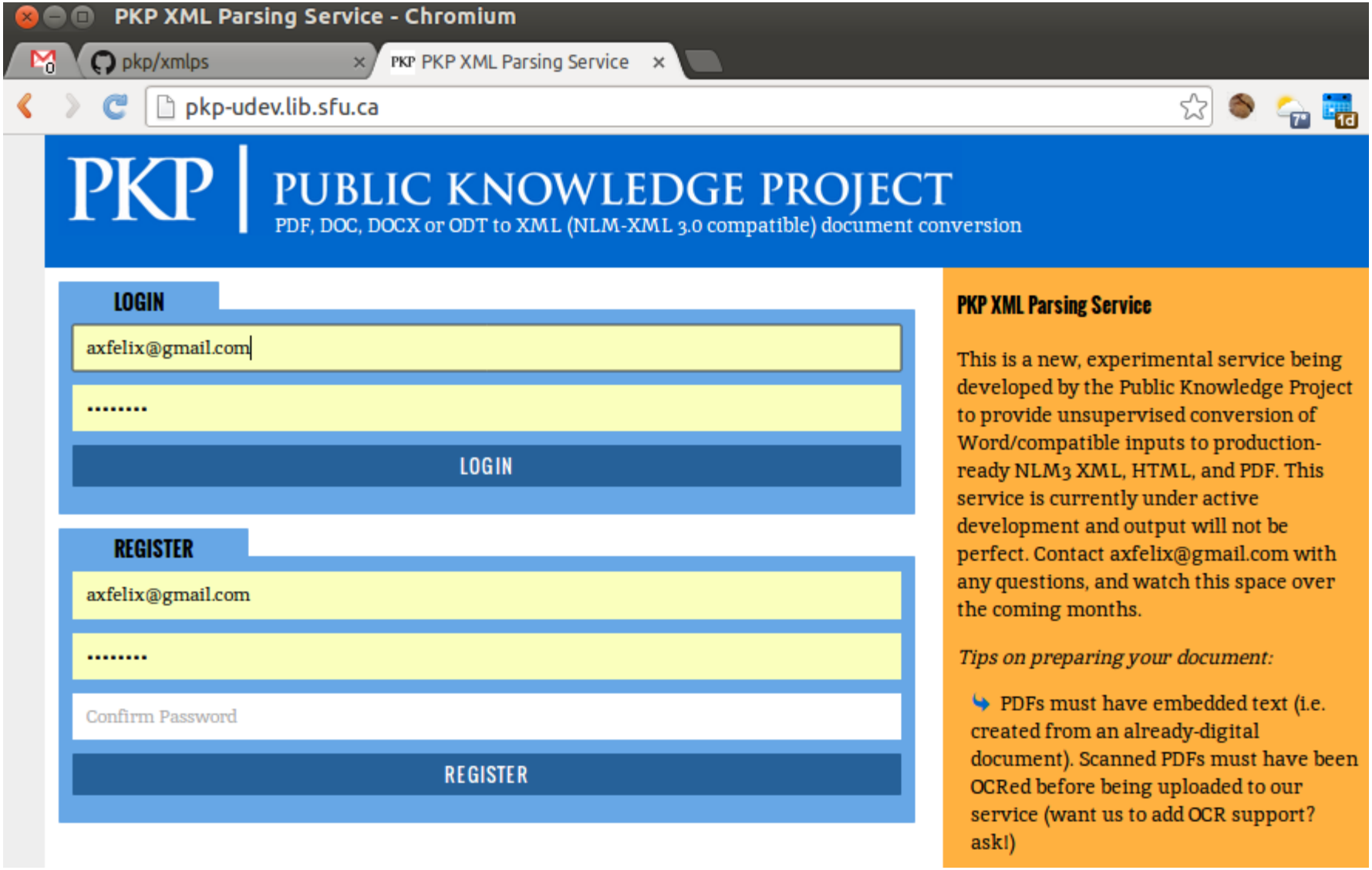
- XML has never, ever, been pleasant for humans to edit by hand. In 2015, no free software makes it better.
- People still don’t use Markdown (or other system X) for some reason.
- Journals that are unwilling or unable to outsource **do not publish** structured articles.

Word, Why?

- The economics of preparing Word manuscripts for publication are awful
- A fully automated solution has not been possible and won't be possible because of Word being Word.
- But we can still help.

The PKP M.O.

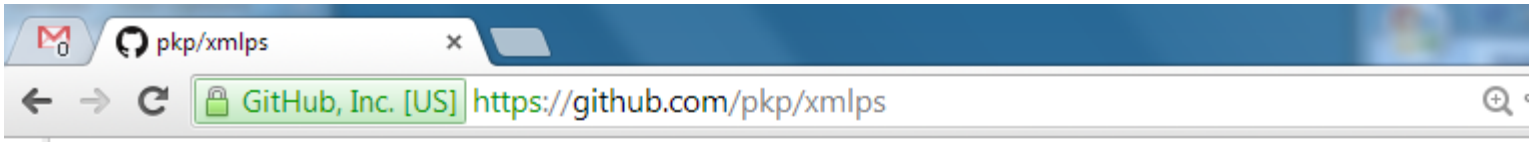
- Many of the journals running Open Journal Systems fit into the “unwilling or unable to outsource” category.
- We won't have any more luck getting authors off of Word or building its replacement than others have.
- But we can make a sane free solution


PKP XML Parsing Service - Chromium
pkp/xmlps
PKP XML Parsing Service
pkp-udev.lib.sfu.ca
PKP | PUBLIC KNOWLEDGE PROJECT
PDF, DOC, DOCX or ODT to XML (NLM-XML 3.0 compatible) document conversion
LOGIN
axfelix@gmail.com
LOGIN
REGISTER
axfelix@gmail.com
Confirm Password
REGISTER
PKP XML Parsing Service
This is a new, experimental service being developed by the Public Knowledge Project to provide unsupervised conversion of Word/compatible inputs to production-ready NLM3 XML, HTML, and PDF. This service is currently under active development and output will not be perfect. Contact axfelix@gmail.com with any questions, and watch this space over the coming months.
Tips on preparing your document:
PDFs must have embedded text (i.e. created from an already-digital document). Scanned PDFs must have been OCRed before being uploaded to our service (want us to add OCR support? ask!)

Fully open source

- Modular solution linking several open solutions to article parsing and rendering
  - ParsCit, wkhtmltopdf, LibreOffice, Pandoc & more

- Core parsing engine “meTypeset” developed in partnership with Open Library of Humanities



    - Handles job distribution to queues
- DocxConversion
    - Converts documents to DocX format
- NlmxmlConversion
    - Converts documents to NLMXML format
- ReferenceConversion
    - Parses references from DocX document into a seperate XML file
- BibtexConversion
    - Converts references from the previous step into Bibtex
- BibtexreferenceConversion
    - Converts Bibtex references into NLMXML and merges the converted references into the NLMXML document
- HtmlConversion
    - Converts the NLMXML document into HTML
- CitationStyleConversion
    - Formats the citations in the HTML document according to the citationstyle requested by the user
- PdfConversion
    - Converts the HTML document into PDF
- XmpConversion
    - Adds an XMP sidecar with metadata from the NLMXML to the PDF document
- ZipConversion
    - Zips all documents
- API
    - Simple REST API to submit and retrieve jobs and to provide functionality for the frontends AJAX callbacks.

Universal Office Converter - Convert between any document format supported by LibreOffice/OpenOffice.
http://dag.wieers.com/home-made/unoconv/

191 commits 1 branch 6 releases 18 contributors

branch: master unoconv / +

Merge pull request #200 from mmariani/version ...

dagwieers authored on 23 Sep 2014 latest commit 2a5cfb8133

doc	Fix typo: "posible" -> "possible"	2 years ago
packaging/rpm	Prepare release v0.6	3 years ago
tests	Improve how FilterOptions are being handled (incompatible changes)	3 years ago
AUTHORS	Initial import of universal OpenOffice convertor	8 years ago
COPYING	Initial import of universal OpenOffice convertor	8 years ago
ChangeLog	Prepare release v0.6	3 years ago
Makefile	Reorganize Makefile, packaging and documentation	3 years ago
README.asciidoc	Update README.asciidoc	2 years ago
WISHLIST	Initial import of universal OpenOffice convertor	8 years ago
unoconv	fixed indent	11 months ago

Code

Issues 109

Pull requests 21

Wiki

Pulse

Graphs

HTTPS clone URL

https://github

You can clone with HTTPS, SSH, or Subversion.

Download ZIP

README.asciidoc

Automated conversion and styling using LibreOffice

< Text Encoding Initiative >

OxGarage Conversion

Select the format into which you want to convert your document

Convert from: ?

Convert to: ?

Documents

- Cocoa tagging
- Compiled TEI ODD
- DocBook Document
- Markdown tagging
- Microsoft Word (.doc)
- Microsoft Word (.docx)
- ODD Document
- OpenOffice 1.0 Text (.sxw)
- OpenOffice Text (.odt)
- Plain Text (.txt)
- Rich Text Format (.rtf)
- TCP XML Document
- TEI P4 XML Document
- TEI P5 XML Document
- TEI Tite XML Document
- WordPerfect (.wpd)
- Wordpress RSS feed of blog
- xHTML

- Comma-Separated Values (.csv)
- DocBook Document
- ePub
- LaTeX
- Markdown tagging
- Microsoft Word (.doc)
- National Library of Medicine (NLM) DTD 3.0
- OpenOffice 1.0 Text (.sxw)
- OpenOffice Text (.odt)
- PDF
- Plain text
- RDF XML
- Rich Text Format (.rtf)
- TEI P5 XML Document
- VerbatimXML tagging
- xHTML
- XML Document
- XSL-FO

meTypeset

- OxGarage fork
- Manually-derived fuzzy parsing logic designed around Word-to-JATS conversion (TEI used as interchange)
  –We barely had to touch the OOXML
- Huge improvement for article body. Focus on parsing free text to nesting.

```
198                                     u'{0}'.format(manipulate.get_stripped_text(title).strip())
199         manipulate.save_tree(tree)
200
201     def nest_headings(self, manipulate, tree):
202         tree = manipulate.load_dom_tree()
203         stack = []
204         message = {}
205
206         for div in tree.xpath('//tei:div', namespaces={'tei': 'http://www.tei-c.org/ns/1.0'}):
207             title = div.xpath('tei:head', namespaces={'tei': 'http://www.tei-c.org/ns/1.0'})
208
209             if len(title) == 0:
210                 size = 100
211                 message[div] = 'No title found in this block'
212             else:
213                 size = title[0].attrib['meTypesetSize']
214                 message[div] = manipulate.get_stripped_text(title[0]).strip()
215
216             stack.append((size, div))
217         first = True
218         position = 0
219         root_size = None
220         root_div = None
221         dict_thresholds = {}
222         for element in stack:
223             if first:
224                 first = False
225                 root_size, root_div = element
226                 self.debug.print_debug(self, u'Set root size as {0}'.format(root_size))
227             else:
228                 size, div = element
229
230                 previous, previous_div = stack[position - 1]
231
232                 if float(size) > float(root_size):
233                     size = float(root_size)
234
235                 # handle an element that is the root size
236                 if float(size) == float(root_size):
```

```
230
231 class ReferenceLinker(Debuggable):
232     def __init__(self, global_variables):
233         self.gv = global_variables
234         self.debug = self.gv.debug
235         self.ibid = None
236         Debuggable.__init__(self, 'Reference Linker')
237
238     def process_ibid_authors(self, ref_items):
239         parsed = 0
240         # this checks for items beginning with "---." and replaces them with the real author name
241         for ref in ref_items:
242             if ref.text is not None and ord(ref.text[0]) == 8212 and ord(ref.text[1]) == 8212 and \
243                             ord(ref.text[2]) == 8212 and ref.text[3] == '.':
244                 try:
245                     current = ref
246
247                     while True:
248                         previous = current
249                         current = current.getprevious()
250
251                         if current is None:
252                             break
253
254                         if current.text is not None and ord(current.text[0]) != 8212:
255                             authorname = current.text.split('.')[0]
256
257                             ref.text = authorname + ref.text[3:]
258                             parsed += 1
259                             break
260
261                 except:
262                     pass
263
264             elif ref.text is not None and ref.text.startswith('_'):
265                 ref.text = ref.text.strip('_')
266
267                 try:
```

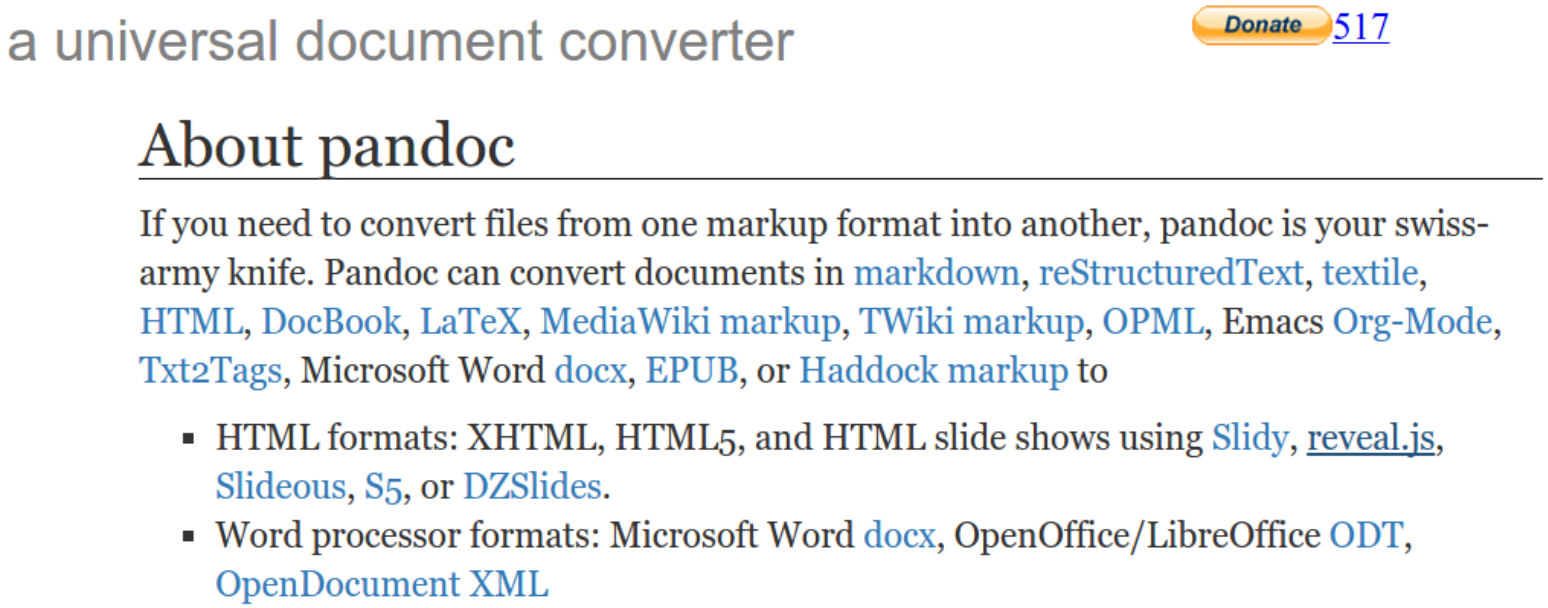
More meTypeset

- Interactive CLI reference checking
- Can supply known-good front matter as additional input parameter
- Handling of embedded Zotero, Mendeley, and Endnote citations
- Mostly Python and XSL.

ParsCit: An open-source CRF Reference String and Logical Document Structure Parsing Package

is the home page of the ParsCit project, which performs two tasks: 1) reference string
sing, sometimes also called citation parsing or citation extraction, and 2) logical structure
sing of scienfific documents. It is architected as a supervised machine learning procedure

- Actively developed (as of 2013)
- Easy to run locally (Perl / CRF++)



- Native Word support added 2014
- Used for reference linking & styling

```
235         * Add the XMP sidecar to the PDF document
236         *
237         * @return void
238         */
239        protected function addXmpSidecar()
240        {
241            $command = new Command;
242
243            // Set the base command
244            $command->setCommand($this->config['exiftool']['command']);
245
246            // Allow duplicates to be extracted
247            $command->addSwitch('-duplicates');
248
249            // Be verbose
250            $command->addSwitch('-verbose');
251
252            // Read tags from XMP sidecar
253            $command->addSwitch('-TagsFromFile');
254
255            // The XMP file
256            $command->addArgument($this->outputFileXmp);
257
258            // The PDF file
259            $command->addArgument($this->inputFilePdf);
260
261            // Redirect STDERR to STDOUT to captue it in $this->output
262            $command->addRedirect('2>&1');
263
264            $this->logger->debugTranslate(
265                'xmpconversion.exiftool.executePdfCommandLog',
266                $command->getCommand()
267            );
268
269            // Add the XMP sidecar
270            $command->execute();
271            $this->status = $command->isSuccess();
272            $this->output = $command->getOutputString();
273
274            $this->logger->debugTranslate(
275                'xmpconversion.exiftool.executePdfCommandOutputLog',
```

management: 6

3 Initial User Inquiry

Too difficult to enter information & data collection concerns	20
Privacy: not wanting to share unhealthy habits or any health information	13
Too much material	10
Digital assets/personal library does not fit	7
Lack the motivation to use the system	4
Concerned about how recipes are shared	2
Privacy Concerns: Not wanting to share the information	2

6 other items mentioned once each

Table #4

Initially we used a questionnaire to test the validity of the ABC Framework (Kamal, Fels, & Ho, 2010) and then used the framework to provide points of inquiry for the design of the VivoSpace social network application. Our aim is to gain insight into end-user motivation through well-tested models. Online and paper questionnaires were used to obtain feedback on both: 1) their motivations in using online social networks and 2) their motivation in changing health behavior.

scalp is 0.1 rT, whereas the earth's magnetic field is ~50 mT. The use of superconducting quantum interference devices (SQID) has enabled these fields to be recorded from the brain with the patient in a magnetically shielded room. These recordings are known as a magnetoencephalogram (MEG).

There are some theoretical and practical differences between EEG and MEG. Although the same electrical currents produce the MEG, it can provide complementary information to EEG.

Electroneurograms (ENG), i.e. recordings made from nerves while stimulating them by electrical pulses or other means, may be used in conjunction with an EEG to test for peripheral and central nerve defects. Peripheral nerve defects may be found in vitamin deficiency, poisoning by alcohol or other substances or after injury. Central nerve defects may be found in demyelinating diseases, such as MS. Defects are shown by the lack of a response or the delay in response to a stimulus, which may be measured at any point along the pathway in the peripheral or central nervous system.

References

Atwood, H. L., & MacKay, W. A. (1989). Hamilton, Canada.: Decker,.

Bickford, R. D. (1987). In G. A. I. E. ed. (Ed. & Trans.), *Encyclopedia of Neuroscience, Birkhauser* (pp. 371–373). Cambridge (USA),.

Bronzino, J. D. (1995). Principles of Electroencephalography. In B. J. D. I. ed. (Ed. & Trans.), *The Biomedical Engineering Handbook* (pp. 201–212). Florida.: CRC Press,.

Brunet, D., & Young, G. (2000). *Electroencephalography, Guidelines for Clinical*

This Journal is Getting Testy

Document Markup Plugin

Settings

Select Citation Style (CSL): American Psychological Association 6th edition

Set preferred citation style for markup. ("Document Markup Server" needs to be set before this can be set).

Stylesheets: This CSS stylesheet affects the layout of articles generated by this plugin. Default styles are automatically provided. You may replace these with your own customized versions using the FILE MANAGER

article.css

Installation Requirements

This plugin can use a guest account (limited to 10 conversions per week) or log in for unlimited use. Leave the User ID field blank for guest account use. For more information, visit PKP Document Markup Service.

User ID: axfelix@gmail.com

Password:

Document Markup Server: http://pkp-udev.lib.sfu.ca/

The journal will retrieve converted documents from this server URL. Normally you

Job Id	User	Status	Creation Date	Upload File Name	Actions
1479	shiminliu2010@gmail.com	Completed	2015/04/13 06:48:58	Strategies for therapeutic hypometabothermia_revised72512.docx	
1478	shiminliu2010@gmail.com	Completed	2015/04/13 06:26:13	Liu.docx	
1477	shiminliu2010@gmail.com	Completed	2015/04/13 06:22:45	respiratory-distress-in-als.docx	
1476	info@diagnomx.eu	Failed	2015/04/13 05:19:16	1-6-1-pb.doc	
1475	g.king@spandidos-publications.com	Completed	2015/04/11 03:28:48	MS-142086-EndNoteCits.doc	
1474	g.king@spandidos-publications.com	Completed	2015/04/11 03:16:01	manuscript.doc	
1473	pradip.jadhav@northgate-is.com	Completed	2015/04/10 03:01:06	es505116p.docx	
1472	pradip.jadhav@northgate-is.com	Completed	2015/04/10 02:43:12	sample-document.odt	
1471	pradip.jadhav@northgate-is.com	Failed	2015/04/09 09:18:08	enquiries (26).csv	
1470	elton.drego@northgate-is.com	Completed	2015/04/09 05:30:47	ie5038769.docx	
1469	journals@indiana.edu	Completed	2015/04/08 12:29:24	test.docx	
1468	lholmes@ariessys.com	Completed	2015/04/08 10:18:02	PONE-D-13-12347R6.doc	
1467	journals@indiana.edu	Completed	2015/04/08 09:59:17	007_Erickson_JFR_copyedit.docx	
1466	journals@indiana.edu	Completed	2015/04/08 08:42:19	007_Erickson_JFR_copyedit.docx	
1465	Scott.Abbott@uts.edu.au	Completed	2015/04/07 19:54:30	GOI Cleall wednesday GALLEY.doc	

Submit

Submit a job to the server. The citationStyleHash is an internal identifier for the requested citaton style. A list of hashes can be retrieved through the citationStyleList API. The API will return the job id which can be used to retrieve the completed job later or to query the server for the job status.

URL: api/job/submit Request type: POST Parameters:

- email
- password
- fileName
- fileContent
- citationStyleHash

I.e.

```
http://example.com/api/job/submit
POST parameters:
    'email' => 'user@example.com'
    'password' => 'passowrd'
    'fileName' => 'document.docx'
    'citationStyleHash' => 'c6de5efe3294b26391ea343053c19a84',
    'fileContent' => '...'
```

Example response:

```
{"status":"success","id":123}
```

Future Work

- Continuing to improve meTypeset
  - Open Library of the Humanities Funding
- Modularizing our stack to use multiple parsing engines & compare results
  - Likewise for reference parsing
- Better automated evaluation
  - Corpus management / regression testing

http://cermine.ceon.pl/index.html

CERMINE

Content ExtRactor and MINEr

CERMINE Home | Tasks and results | About CERMINE | CERMINE at GitHub

Welcome to CERMINE - Content ExtRactor and MINEr

Upload PDF file

Upload a PDF file containing scientific article:

Choose file | No file chosen | Upload

Or process one of the example files: Example #1 (PDF), Example #2 (PDF), Example #3 (PDF)

About the service

CERMINE is a Java library and a web service for extracting metadata and content from scientific articles in born-digital form. The system analyses the content of a PDF file and attempts to extract information such as:

- Title of the article
- Journal information (title, etc.)
- Bibliographic information (volume, issue, page numbers, etc.)
- Authors and affiliations
- Keywords
- Abstract
- Bibliographic references

Limitations

Feedback

%	precision	recall	F1
abstract	76.79 (+30.42, +29.54)	88.97 (+35.18, +33.38)	82.43 (+32.62, +31.35)
title	91.95 (+1.14, +6.74)	89.31 (+4.14, +5.86)	90.61 (+2.71, +6.29)
journal	88.72 (+9.64, +0.16)	72.30 (+26.67, +0.23)	79.67 (+21.80, +0.20)
authors	90.05 (+6.56, +5.58)	86.16 (+19.62, +2.71)	88.06 (+14.00, +4.11)
affiliation	78.70 (+9.88, +1.98)	74.57 (+26.10, +12.16)	76.58 (+19.70, +7.76)
year	98.41 (+7.03, +4.64)	92.53 (+34.02, +4.37)	95.38 (+24.04, +4.50)
volume	97.26 (+1.74, +0.26)	85.75 (+29.43, +0.23)	91.14 (+20.28, +0.24)
issue	96.30 (+0.31, +0.34)	62.87	76.08 (+21.12, +0.11)
average	**89.77 ()**	**81.56** (+24.94, +7.39)	**84.99** (+19.53, +6.82)

The Pitch

- meTypeset has already been adopted by the University of Michigan's *mPach* HathiTrust preparation system

- We've been out of money for this project for most of a year (no improvements and minimal advertising) and we're still getting several testers each day

Abstract	**Title:** *empty* This is a sample abstract that forms part of the *bookMetadataSample.xml* file.
Permissions	All rights reserved © 2014, Heidelberg University

Next ❯

Debug Area


```
{
  "book": {
    "@xmlns:mml": "http://www.w3.org/1998/Math/MathML",
    "@xmlns:xlink": "http://www.w3.org/1999/xlink",
    "@xmlns:xsi": "http://www.w3.org/2001/XMLSchema-instance",
    "book-meta": {
      "book-id": {
        "@pub-id-type": "other",
        "#text": "handbook1"
      },
      "book-title-group": {
        "book-title": {
          "@xml:lang": "en",
          "#text": "A Sample Book"
        }
      },
      "volume": "1",
      "series": "Heidelberg Studies in Transculturality Open Access Book Serie
s",
      "edition": "1st",
      "contrib-group": {
        "contrib": [
```

Professional Use

- We're building our system with post-parse cleanup hooks in mind
- Heidelberg University is developing a WYSIWYG TEI editor which our system could hand off results to for production prep; others welcome!

Want to Help?

- The stack is open source, but owing to the myriad components, it’s not GPL-licensed

- If you want to help us modularize the kit so that you can add your own proprietary module – by all means do!
  - Cash is nice too

2016 Deliverables

- Efficiency study of time savings when a full-XML-production shop uses our system at the start of markup

- Full, free testing and evaluation corpus of articles in both original author Word format and full JATS; parsing evaluation tests written in the Robot framework.

Thanks!

- **Martin Eve**, Open Library of the Humanities – meTypeset maintainer and all-around great guy and collaborator. This wouldn't have gone nearly as far as it did without him.

- **Stanford University MediaX Incubator and Konica-Minolta**, for funding $250k of work to date.

Questions?

garnett@sfu.ca

http://github.com/pkp/xmlps