AUTOMATING PROCESSING AND INTAKE IN THE INSTITUTIONAL REPOSITORY WITH PYTHON

John Beatty
INTRODUCTION
What are we doing here?
Populating the Institutional Repository

- Year 1: Faculty scholarship
- Year 2: Law Journals & Alumni Publications
Disclaimer/Full Disclosure

It’s true I had zero Python programming knowledge at the start of this project. But I was starting with some knowledge:

• General programming knowledge & experience
• VisualBasic (17 years; 4 database applications built)
• Bash (basic knowledge)
• Perl (mostly forgotten knowledge)
• Has used a regular expression
The Law Journal Project: The Journals

- Buffalo Environmental Law Review: 23 volumes, 1-2 issues/volume
- Buffalo Human Rights Law Review: 22 volumes, 1-2 issues/volume
- Buffalo Intellectual Property Journal: 11 volumes, 1-2 issues/volume
- Buffalo Journal of Gender, Law & Social Policy: 24 volumes, 1 issue/volume
- Buffalo Law Review: 65 volumes, 3-5 issues/volume
- Buffalo Public Interest Law Review: 35 volumes, 1 issue/volume
The Law Journal Project: Workflow

- Convert Hein metadata to Digital Commons format
- Load PDFs into Box drive
- Preview files in Box
- Check metadata against PDF and correct where necessary
- Cut and paste Box links into Digital Commons spreadsheet
- Upload
The Law Journal Project: Timeline

- August-November 2018
- All but first 22 volumes of Buffalo Law Review complete in mid-November
THE PROBLEM
What’s so special about the Law Review?
Some types of documents are in the system as a section rather than individual pieces.

Combined files have no individual metadata.

Some documents have no author data.

Some articles missing the last page.
Book Reviews

- In HeinOnline, book reviews in a single BLR issue are all in one file
- All book reviews are signed, but no author data in HeinOnline
- In later volumes (processed first), issues contain 2 book reviews at most; splitting and metadata creation was done by hand
- In early volumes, there are up to five book reviews an issue, so automation helpful

BUFFALO LAW REVIEW

such as the inspection of wearing apparel, breath tests, corporate records, fingerprint examinations, truth serums, scienisis and voice identification. Unquestionably Professor Maguire has assembled a nearly exhaustive body of authority to define, delineate and illuminate each of these areas.

The book is well received, for in scrutinizing these essential rules of evidence, Professor Maguire traces their history from their points of origin to the most recent decisions which take cognizance of special legal phenomena having contemporary relevance. He dramatically demonstrates how the means utilized by some law enforcement officials, erroneous to apprehend those suspected of crimes, have come into collision with our state and federal constitutions. This conflict is epitomized by the McNibly-Maloney doctrine which is meticulously and expertly examined.

Professor Maguire, in dealing with each of the five vital rules of evidence, critically evaluates them within our present day legal framework and then prognosticates as to their survival in the future.

In the reviewer's opinion, one of the book's greatest contributions is the manner in which it leads order and consistency to what otherwise appears to be a maze of conflicting rules and decisions. Through Professor Maguire's keen analysis and careful cataloging of the leading authorities, he is able to show some direction in the development of these less than precise rules.

For the New York practitioner there is an abundance of New York authorities cited in each of the problem areas. Every attorney facing these problems will surely welcome Professor Maguire's book as a much needed guide when charting a client's course through hazardous litigation.

HERALD F. BARSHOVER, JR.
Member of the New York State Bar


Suppose that Citizen is awakened one night by the heavy knock of the gloved hand on his door. A group of uniformed troopers force their way into the house, ignoring his request to see a warrant. The leader informs him that the government has information that he is an athlete, and that by a recent decree atheists were declared enemies of the People and subject to summary execution. He is imprisoned, locked alone in a cell, permitted to see no one, and awaits his execution without even the pretense of a trial.

Has Citizen any legal rights? Did he have any legal rights violated, and, if so, how may they be vindicated? Of course, this raises the question what are legal rights—but that I pass over for this review. We may view Citizen's problem by examining the hierarchy of laws. Moving from the lower to the higher rungs, one first looks to the local subdivision of government and its laws,
Case Notes & Legislative Notes

- Case notes all combined into all case notes for an entire issue
- No individual note or author metadata
- 2-3 issues/volume
- 5-10 case notes/issue
- Same for legislative notes, but only a few issues have them

RECENT DECISIONS

Parties or the registry of the car, however, it seems, therefore, that the Court of Appeals should not find that the weight of post decision precludes it from a reasoned approach to the facts in the instant case.

If in Elling the Court of Appeals elected for the victim of an airplane crash a result that was consistent with its policy, there would seem to be no reason why the same measure of justice should not be extended to victims of other accidents occurring without the state, at least when the interests of New York outweigh the interests of the foreign jurisdiction as greatly as they did in Elling. It seems that Judge Field, at least, would be receptive to an argument for reversal in Babcock because of his opinion in Elling. He favored "the most significant contact or contacts" approach but considered himself bound by the weight of prior decisions to the contrary in wrongful death cases. Such precedent has not been established in actions for common-law negligence like Babcock. Admittedly, Elling has been rendered inapplicable by Daumport v. Webb. Nevertheless, the Elling decision arrives at a just result and the equities seem to justify a proliferation that will reverse and reinstate the complaint in Babcock.

John O. D'Alonzi

OBSCENITY IN NEW YORK: LAW, FACT—OR BOTH?

A number of cases in recent years have defined, redefined and refused the tests of what constitutes actionable obscenity in printed matter under state and federal obscenity statutes. Two fundamental problems form the core of the need for such constant adjustment: (1) the necessarily vague construction of most of the relevant statutes; and (2) the delicate distinctions imposed in striking a balance between objective pornography and freedom of artistic expression. The decision that has had the broadest effect upon each of these considerations, if volume of interpretative and critical comment

Is. 9 N.Y.2d 34, 37, 172 N.E.2d 526, 529, 211 N.Y.S.2d 103, 1118 (1961) (separate opinion). Judge Field concurred with the majority in substance at the present dissenting the same of such, provided a greater, but was given to the dissenting any other. The decision of the court has been reprinted in an action against a printer to avoid a Pennsylvania statute involving recovery. Dye v. Erie, 40 N.Y.2d 327, 337 (1973).

1. See, e.g., Robb v. United States, 344 U.S. 84 (1952); Albert v. California, 344 U.S. 476 (1953); Bailey v. Michigan, 344 U.S. 309 (1952); People v. Hudson County News, 9 N.Y.2d 329, 173 N.E.2d 68, 176 N.Y.S.2d 281 (1957). Two fundamental problems form the core of the need for such constant adjustment: (1) the necessarily vague construction of most of the relevant statutes; and (2) the delicate distinctions imposed in striking a balance between objective pornography and freedom of artistic expression. The decision that has had the broadest effect upon each of these considerations, if volume of interpretative and critical comment

369
Court of Appeals

- Court of Appeals is highest court in New York
- Volumes 3-14 contain case note summaries for the prior year’s Court of Appeals term
- 1 or 2 issues/volume
- Up to 150 case notes/issue
- In most volumes, notes are signed

COURT OF APPEALS, 1956 TERM

ADMINISTRATIVE LAW

Board Member Absent From Administrative Hearing Entitled To Vote

Courts, in reviewing the decisions of administrative tribunals, are limited to ascertaining whether or not there is a rational, legal basis for each determination. This does not mean that the courts will "probe the mental processes" of the body but rather that they will determine only whether there has been an independent appraisal and conclusion based on sufficient knowledge.

In *Yarb v. Perilo*, in an application for a variance from a local zoning ordinance, the Court of Appeals unanimously held that, though a member of the Board was absent from the public hearing and did not read the transcript of the record, he nevertheless possessed adequate knowledge to make an informed decision. The Court pointed out that because the absent member had for years been a resident of the village and a village trustee, and because he had discussed the arguments presented at the hearing before voting, he possessed sufficient information to reach an independent and competent conclusion.

Thus the Court has held that an administrative board member may vote on the disposition of a proceeding, though absent from the hearing therein, provided that he has acquired independent knowledge of the issues involved. It is submitted that this proposition is susceptible of unreserved extension and its application should be scrutinized to similar factual settings.

Hotels Not Special Class Of Housing Accomodations Within Rent Control Law

In *Hotel Association of New York City v. Wacek*, the Court held that the Temporary State Housing Commission did not act in an arbitrary and capricious manner when it ruled that hotels as such did not constitute a "particular class of housing accommodations" within the meaning of the Emergency Housing Rent Control Law.

Petitioners, an association of one hundred and seventy-five hotels in New York City, petitioned the Rent Administrator to find that hotels as such are a particular class of housing accommodations within the meaning of the statute and entitled

---

3. *Yarb*.

---

63
In early volumes, notes did not always start at the top of a page

All page breaks were at the start of the next note

Some notes missing the last page
Why do the extra work?

• In some cases, combined works are substantial (review essays)
• To properly credit alumni and faculty authors
• Some case notes are contemporary coverage of substantial changes in New York or United States law
• Some notes written by prominent alumni
### Implementation Issues

<table>
<thead>
<tr>
<th>Previous Solution</th>
<th>Our Situation</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Requires 2 librarians and a student worker</td>
<td>• Tech services departments busy with massive LSP migration</td>
</tr>
<tr>
<td></td>
<td>• Most departments shorthanded because of retirement</td>
</tr>
<tr>
<td></td>
<td>• No funding for student workers</td>
</tr>
</tbody>
</table>
The Solution: Automation

- Personnel available: 1 Faculty Scholarship Librarian
- Drastically shorten the amount of time needed to generate metadata and split PDFs
- Use generated metadata and split PDFs in established workflow
Timelines

Proposed:

• Learn enough Python to start coding: 1-2 weeks
• Write initial code and test: 1-2 weeks
• Process 22 volumes: 1 month

Actual:

• Learn enough Python to start: 3 days (Thanksgiving week)
• Initial code and test: 5 days (November 26-30)
• Process 22 volumes: 4 weeks (December 3-21, January 3-11)

Note: Processing time included a LOT of code tweaking.
THE PROJECT
First Steps
Learning Python

- John Mueller: Beginning Programming with Python for Dummies
- Kent D. Lee: Python Programming Fundamentals
- T.R. Padmanabhan: Programming with Python
- Python Documentation: https://docs.python.org/3/
- w3schools.com: https://www.w3schools.com/python/default.asp
- Automate the Boring Stuff: https://automatetheboringstuff.com/
Programming Environment

- Laptop computer running Ubuntu Linux 18.04
- PyCharm Community Edition (free!)
- Python 3.6
Identifying Libraries

- PyPDF2: PDF toolkit that can be used to extract data and manipulate PDF files
- pdfminer: A tool for extracting information from PDF files (using pdfminer.six, for Python 3 compatibility)
- openpyxl: Python library to read and write Excel 2010 xlsx/xlsm files
- Standard Python libraries: argparse, os, re, csv, fnmatch, io
- Add-on libraries installed with PIP
Wait… TWO PDF libraries?

- Yes, two PDF libraries
- PyPDF2 has good tools for manipulating PDFs, but the documentation specifically says not to rely on the text extraction functions
- pdfminer is designed to extract information including text and layout from PDF files, so can be relied on for text extraction. But it doesn’t have the manipulation functions.
Initial Workflow: Single Script

- Search through PDF for start page (PDF), end page (PDF), author, title, start page (printed)
- Split PDF into multiple files based on start and end pages
- Export metadata into Excel file to be cut and pasted into Digital Commons batch spreadsheets

Just One Problem:
OCR. It’s not good enough to allow the code to consistently identify the metadata elements.
Scan file

Use appropriate dsplit-XX.py to extract metadata. Use the --write-csv-only option because none of the OCR is good enough to trust that it’s right.
Check metadata

Open the CSV file and check it against the original PDF. Fix titles, authors, and most importantly, start and ending pages for the PDF split.
Split PDF

Feed hand-corrected CSV and original PDF back to dsplit-XX.py to split. For extra fun, hand-correct a couple of volumes, then use a bash script to run through them all while you get coffee.
Convert CSV

Feed that CSV file to dc-convert.py. Copy everything back to the main computer. Cut and paste entries from exported Excel file into DC spreadsheet.
Hand-check as normal

Open split PDFs in Box preview. Check page split. Double-check metadata. Add disciplines. Cut and paste Box link.
journaltools.py

Main Python code—Contains all reusable code

- Author name and title manipulation (splitname, capitalize_title)
- PDF splitting code (splitpdf)
- PDF reading code (getpdf)
- CSV manipulation (importcsv, exportcsvnew, convertcsv)
- Page preparation (doublepages, croppages)
- PDF manipulation code (combinepdf, shiftpage, dirshift)
- Support code (getfilenames)

Most of these code segments called by external files that act as command-line interfaces

- E.g. dir-shift.py: Takes a path and passes it to dirshift
dsplit-XX.py

• This is the main metadata extraction and PDF splitting code.
• Different command line file is used for each type of file scanned.
• Consists of a command-line interface and scanning code.
• Remainder of code is the same for each. Calls to journaltools.py.
Other functions

- **combine-pdf.py**: Used to combine Hein-split volume indexes back into a single file. Takes a path and combines all files in filename order.

- **dc-convert.py**: Exports CSV file to an Excel file, with metadata in the proper columns to be cut and paste into DC upload sheets.

- **dir-shift.py**: Takes a path; copies the first page of every file and adds it as the last page of the previous file in the directory.

- **page-shift.py**: Takes two files and copies the first page of the second file and adds it to the end of the first file (quickly replaced by dir-shift.py).
EXTENSIONS

What else can I do with this thing I built?
New volumes of Buffalo Law Review

- Five new issues a year need to be processed and uploaded
- NO OCR text
- New command line program extracts metadata from a single file
- Bash script used to scan all articles and write to a single CSV
- Total processing time for an issue: About 15 minutes
UB Law Forum

- 38 volumes, 1-2 issues/volume
- OCR text too unpredictable to automatically scan for metadata
- Contents page fairly comprehensive
- Partial automation solution
- Contents text copied and pasted into text editor, cleaned up with search and replace, then copied into Excel file
UB Law Forum

- New code to crop from full magazine page scans to 8.5 x 11
- New code to convert hand-built Excel file to CSV
- PDF splitting and export command lines re-used
Resources

- The journaltools code, at GitHub
  https://github.com/johnrbeatty/journaltools

- These slides
  https://digitalcommons.law.buffalo.edu/law_librarian_other_scholarship/