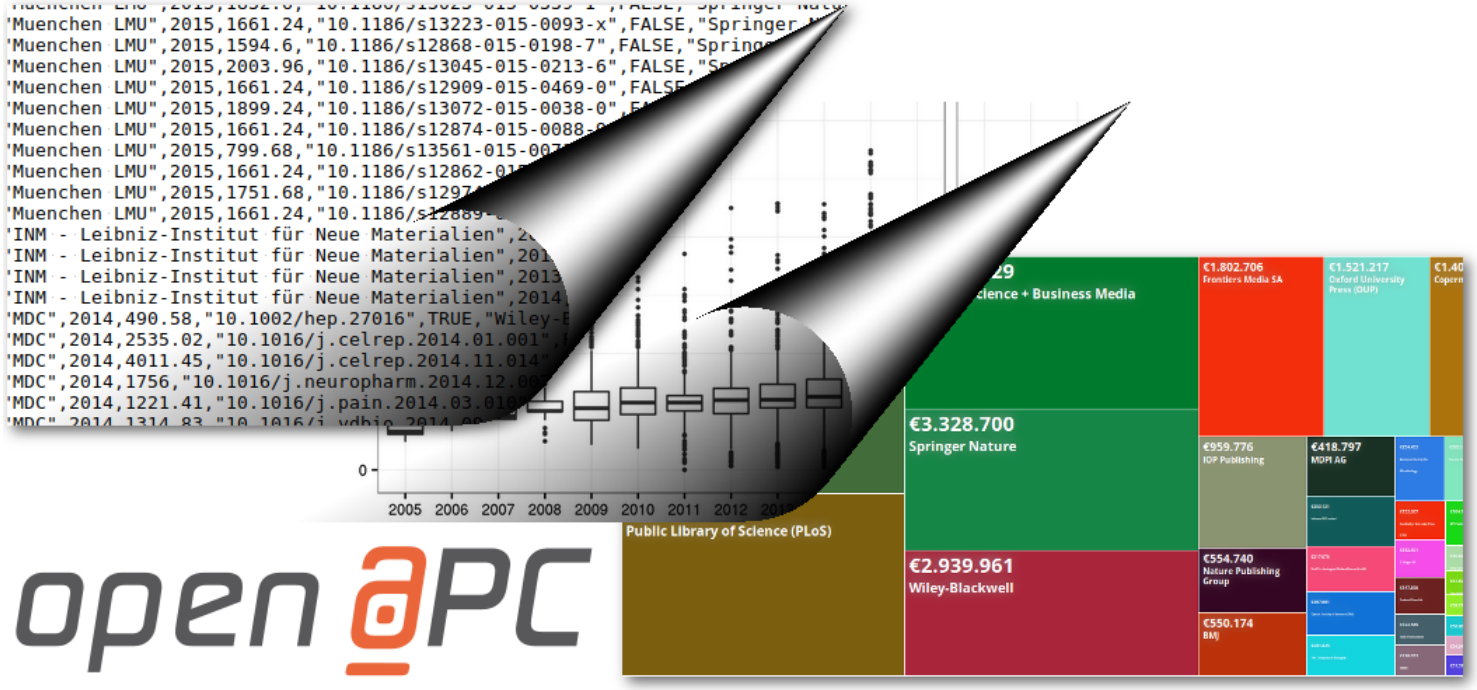
OpenAPC - A contribution to transparency in fee-based Open Access publishing



Christoph Broschinski, Bielefeld University Library, <broschinski@uni-bielefeld.de>

This presentation is also available at **http://www.ub.uni-bielefeld.de/~cbroschinski**

OpenAPC in a nutshell

OpenAPC is an **Open Data** project focused on Article Processing Charges (APCs)

- APC data is collected from participating institutions (universities, research centers, funding organisations)
- Data is being normalised, enriched and processed
- Results are made available for reuse in different ways (raw data, visualisations, OLAP service)

Key properties

- Extensive openness approach: Make not only the data available, but also intermediate results, processing scripts, server applications and materials
- No red tape involved: Submitting a single schema-conforming csv file is enough to become part of OpenAPC

Statistics (as of 09/16)

- 40 participants
- 18,370 articles
- Total APCs amount: 32,802,770€

Timeline

2014: OpenAPC started as a private endeavour by our colleague Najko Jahn (Project Manager at Bielefeld University Library)

- First Participants: German University Libraries receiving funding for paying APCs from the DFG Open Access Publishing Programme
- Good opportunity to start an open data project on APCs: Compiled APC data already existing

2015: Application for DFG funding together with 2 partnering initiatives:

- ESAC at the Max Planck Digital Library (MPDL), Munich (workflows and management connected to OA publishing)
- OA Analytics at the Institute for Interdisciplinary Studies of Science (I²SoS), Bielefeld University (OA bibliometrics)

Funding granted (for 3 years, starting in October 2015), partners teamed up under new label INTACT

General Structure

OpenAPC is designed as a **git** project and hosted on **GitHub**

- git: Version control system / management tool usually employed for software development
- GitHub: Web-based hosting service for git repositories with extended tool support

Why git?

- Revision control (history, tracking of changes)
- Reverting mistakes
- Branching (Testing scripts and changed data in separate branches without influencing the main version)

Why GitHub?

- Ensure project's open approach (Everything is readable by everyone at every point of time)
- Dynamic project status report (via **R Markdown**)
- Wikis, graphical code comparison, pull requests

In addition, OpenAPC operates a blog on github pages to report recent data submissions.

Data: Schema, Submission and Storage

- The Open APC data is stored and maintained in a core data file on GitHub ("data/apc_de.csv")
- Core data file format is described by the Open APC Schema definition.

column	description	source	input_required
institution	Top-level organisation which covered the fee, e.g. Bielefeld University	none	mandatory
period	Year of APC payment (YYYY)	none	mandatory
euro	The amount that was paid in EURO. Includes VAT and additional fees	none	mandatory
doi	Digital Object Identifier	none	mandatory
is_hybrid	Has the article been published in a toll access journal?	none	mandatory
publisher	Name of publication house that has charged the fee	CrossRef	optional
journal_full_title	Full name of periodical that contains the article	CrossRef	optional
issn	International Standard Serial Number	CrossRef	optional
issn_print	International Standard Serial Number - print version	CrossRef	no
issn_electronic	International Standard Serial Number - electronic version	CrossRef	no
issn_l	Linking International Standard Serial Number	ISSN International Centre	no
license_ref	License under which the research paper has been published	CrossRef	no
indexed_in_crossref	checks if the contribution is registered with the DOI agency CrossRef	CrossRef	no
pmid	id for metadata records indexed in Europe Pubmed Central (Europe PMC)	Europe PMC	no
pmcid	id for articles available in Europe PubMed Central full text collection	Europe PMC	no
ut	Web of Science unique item id	Web of Science	no
url	URL to article if no DOI is available	none	optional
doaj	Is the journal indexed in the Directory of Open Access Journals (DOAJ) ?	DOAJ	no

Institutions are required to follow the schema when contributing data. However:

- Only the 5 mandatory fields are always required
- The 4 optional fields are required if (and only if) the article in question does **not have a DOI**
- Every other field can (and should) be omitted, because these values are determined automatically by our metadata enrichment scripts

Institutions may submit their data in one of two ways:

- Via E-Mail (Attached CSV file or Excel/LibreOffice Spreadsheet)
- Via GitHub (By putting a CSV file into their institutional directory and making a pull request)

Open Data: The next Level

Managing a large data aggregation project like OpenAPC is time-consuming and prone to mistakes. What techniques can be employed to reduce the workload and improve results?

Problem 1: Submitted files are often inconsistent, yet they have to be enriched with metadata from several external sources

- Solution: A sophisticated script (python-based) that handles the whole enrichment process
- Handles different file encodings, monetary formats, wrong column names and unsorted columns automatically
- Queries external services to fetch metadata for every article
- Output is compliant to the OpenAPC data schema and can be directly pasted into the core data file

Problem 2: Statistics and Numbers on the GitHub front page have to be kept up-to-date whenever the core data file changes.

- Solution: Utilize R Markdown to regenerate the page whenever necessary ("Dynamic reporting")
- Simply call `R -e "knitr::knit('README.Rmd')"` from the OpenAPC main directory
- The same method is used to build our blog posts on the github pages blog

Problem 3: Even with the enrichment script taking care of the syntax, every submission may still introduce semantical errors into the core data set

- Solution: A test script checks the whole data set for such errors automatically after every commit ("Continuous Integration"). Examples:
- Are there any DOI duplicates?
- Are field values correct (f.e. "is_hybrid" must be TRUE or FALSE, nothing else)?
- Are journal titles/publisher names consistent across same ISSNs?
- Test results are automatically sent to the OpenAPC maintainers

Exploration and reuse: The OLAP server

The core data file is in CSV format. While this makes the data platform independent and easily processable by software tools, it is not really suited for data exploration or querying.

- To adress this problem, an **OLAP** service was set up to provide an alternative means of accessing the OpenAPC data
- OLAP (Online analytical Processing) is a standard tool in business intelligence which models financial data in so-called **cubes**
- A cube can then be divided into a smaller subunits (via **drilldowns** and **cuts**), with their data being aggregated to answer questions

Examples:

- "Show the whole cube without any drilldowns."
- http://olap.intact-project.org/cube/openapc/aggregate
- "Show the APC spendings for all institutions in the year 2014."
- http://olap.intact-project.org/cube/openapc/aggregate?drilldown=institution&cut=period:2014
- "List every hybrid journal (by title and publisher) the Wellcome Trust paid for in 2015 and order them by the average APC amount paid."
- http://olap.intact-project.org/cube/openapc/aggregate?drilldown=publisher|journal_full_title&cut=is_hybrid:TRUE|period:2015|institution:Wellcome+Trust&order=apc_amount_avg:desc

Visualisation: Interactive treemaps

OLAP is a mighty tool for data analysis, but it still unsuited to gain a simple overview over the data or identify interesting facts.

To enable a visual approach for data exploration, a web site providing treemap visualisations was set up: http://treemaps.intact-project.org

- Provides interactive treemaps either for the whole data set or for individual institutions
- Uses the OLAP server as backend and is functionally equivalent
- Based on the offenerHaushalt/openspending project by the OKFN.

Further plans & lookout

More participants, more data

- Increasing number of institutions (both national and international)
- Next candidate: FP7 post-grant Open Access publishing funds pilot (estimated: 600 additional articles)
- Current automatisation approach must be enforced and improved to keep up with increased workload
- **Example:** Provide a web-based frontend to our enrichment script so that participants may process (and correct) their data themselves up front

Increase reuse options

- Participants should be offered appropriate counter-value for submitting their data
- At the moment: Embedding treemap visualisations into institutional web sites
- **In development:** Automatically generated plots and charts which can be downloaded as image files