Towards Machine-Actionable Modules of a Digital Mathematics Library*

The Example of DML-CZ

Michal Růžička[1,2], Petr Sojka[1], and Vlastimil Krejčíř[1,2]

[1] Masaryk University, Faculty of Informatics
Botanická 68a, 602 00 Brno, Czech Republic
mruzicka@mail.muni.cz, sojka@fi.muni.cz
[2] Masaryk University, Institute of Computer Science
Botanická 68a, 602 00 Brno, Czech Republic
krejcir@ics.muni.cz

Abstract. Publishing and archiving mathematical literature presents its own sets of problems. Reaching the goal of building global digital mathematics library (DML), smaller DMLs play an inevitable role in collecting, validating, digitizing and checking data from smaller publishers.

In this paper, we overview the technical challenges of building a machine-actionable set of modules we have developed over almost a decade of evolution of the Czech Digital Mathematics Library (DML-CZ). Firstly, we survey methods of effective automated data acquisition from the content providers. Then we show OCR processing of mathematical documents and automated segmentation of plain text references for metadata enhancement and effective DOI look up. Finally we describe connection to the European Digital Mathematics Library (EuDML) project and public interfaces of DML-CZ for the best visibility and accessibility.

Keywords: DML-CZ; EuDML; DOI; ParsCit; references; validation; DSpace; OAI-PMH; TeX; LaTeX; Tralics; Infty; machine-actionable digital library; library automation; Google Scholar; webometrics

1 Introduction

Publishing and archiving mathematical literature is a unique and challenging task in many respects. It revolves around handling of mathematical formulae in papers, dealing with the number and diversity of math publishers' size and approaches, as well as the existence of reference databases as Mathematical Reviews and Zentralblatt Math community services. There are specific citation patterns across a great diversity of topic areas throughout their long evolution. Handling all these specifics in a local digital mathematical library (DML) possesses variety of challenges.

* Preprint of a paper published by Springer in J. Carrete et al: Proceedings of CICM 2013 on pages 263–277 to be found at Springer Link. Preprint published with Springer's permission.

One possible approach to run and sustain a project of a small digital library is to try to automate as many processes as possible, while still maintaining high-quality, checked data in the repository. The running costs of such a system can be too high to allow it to survive in the digital publishing ecosystem unless machine-actionable modules are developed for a DML.

This paper provides an overview of the technical challenges we have coped with in the design, development and adoption of technologies and technical solutions during almost a decade of evolution of the Czech Digital Mathematics Library (DML-CZ, `http://dml.cz/`).

The structure of the paper is as follows: The following section reports on the interfaces and formats we have settled on in the DML-CZ with our data providers — a machine-actionable input module that validates the data from them. In Section 3 we discuss the tools and the conversions we perform on the data collected from data providers: getting the full text including math formulae, enhancements of bibliographies, reference and DOI matching modules. Section 4 deals with the interfaces we use to export the enhanced and checked data to the wider public: it is available to EuDML, Google Scholar and review databases.

The schema of the modules described in this paper can be seen in Figure 1. Ellipses refers to external entities. The automatic modules, depicted as rectangles, are integrated in our two main subsystems, Metadata Editor and DSpace, shown here as circles.

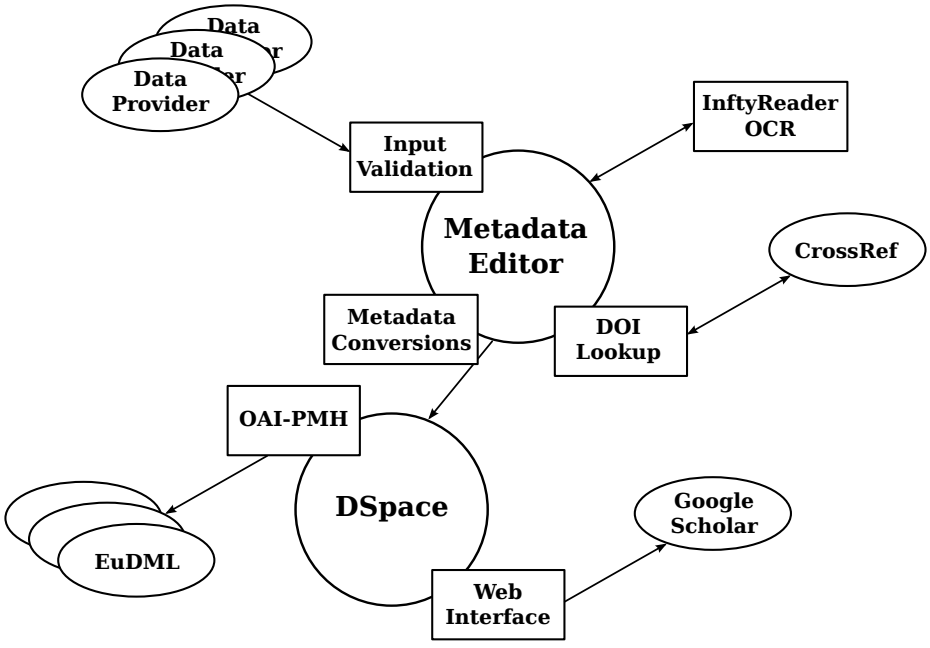

Fig. 1. Overview of modules described in this paper.

2 Inputs

The aim of the DML-CZ project is the digital preservation of the content of the bulk of the mathematical literature that has ever been published in the Czech lands. Since the start of the project several years ago, most of the old publications have been processed. With fewer papers yes to be retro-digitized, it is increasingly important to cooperate with editors of the active Czech mathematical journals on the continuous inclusion of new publications. Journal papers are the core contents that are regularly added to the library. DML-CZ holds ten journals that constitute the content, which amounts to several hundreds new papers per year, on a regular basis:

1. Acta Universitatis Carolinae. Mathematica et Physica
2. Acta Universitatis Palackianae Olomucensis. Facultas Rerum Naturalium. Mathematica
3. Applications of Mathematics
4. Archivum Mathematicum
5. Commentationes Mathematicae Universitatis Carolinae
6. Communications in Mathematics
7. Czechoslovak Mathematical Journal
8. Kybernetika
9. Mathematica Bohemica
10. Pokroky matematiky, fyziky a astronomie

For the long term sustainability of the DML-CZ project it was vital to reduce the costly manual labour in the routine processing of new publications while maintaining the good quality of metadata. To achieve this, we cooperate closely with the publishers who prepare the DML-CZ data as an integral part of their publishing process. Data providers prepare the DML-CZ data according to the DML-CZ input format specification. This simple data format is related to the internal data format of the DML-CZ Metadata Editor tool (see Section 3) and consists of these parts:

1. XML metadata file describing the publication (title, authors, abstract, keywords, ...),
2. XML metadata file containing a semantically marked up list of references of the paper, i.e. each of the references has properly marked author names, title, year of publication etc.,
3. full text of the paper in the PDF format, and,
4. optional but highly recommended bunch of source files[1] suitable for generation of the paper.

An XML metadata file containing a semantically marked-up list of references is important for the proper presentation of metadata via the web interface of the

[1] Being a *mathematical* digital library, the DML-CZ content providers use almost exclusively the TEX typographic system for the preparation of their publications.

digital library and especially for further internal processing (e.g. DOI lookup, see Section 3.2) and for use by third parties (see Section 4.2).

The inclusion of sources in the data package is optional, but we strongly recommend it. The availability of the original source codes enables us to instantly correct some sorts of errors that are occasionally found in the metadata provided. One example of such corrections is the substitution of the authors' custom TEX macros for their LATEX equivalents in the metadata. We support fixed set of macros, including the ones used in LATEX packages developed by the $\mathcal{AMS}$.

There are three basic options for DML-CZ data providers to prepare the data for DML:

1. Develop and use their own tools for DML-CZ data generation,
2. adopt a 'complex' DML-CZ LATEX-based processing system,
3. integrate a 'lightweight' TEX extension for DML-CZ.

These options were documented in our earlier publications. [Růž08, RS10, RS11]

No matter which option the data provider uses, the result is a data archive ready to be delivered to the DML-CZ Metadata Editor for further processing and subsequent publication. To be automatically processable, the data archives have to follow the above-mentioned rules. As there is a variety of different ways our publishers prepare the data the *input validation* module of the Metadata Editor checks compliance of the provided data with the DML-CZ requirements.

The validation process includes

- data integrity tests,
- tests of the completeness of the data set,
- the validation of XML metadata (title of the publication and the list of references),
- validity of the LATEX code included in the metadata.

Mathematical publications collected in the DML-CZ contain a lot of mathematical expressions, and these expressions often appear in the metadata. Thus, DML-CZ allow the use of mathematical expressions in the metadata encoded in the LATEX notation. The use of LATEX markup is allowed only for the mathematical statements. The rest of the metadata is plain text without any special markup. Moreover, the set of allowed LATEX macros is fixed, restricted to the LATEX and subset of $\mathcal{AMS}$ packages. No new macros may be used in titles, abstracts or the bibliography. Compliance with these restrictions is automatically checked, and the metadata validated. These tests save us from further manual corrections as the DML-CZ workflow requires the automatic conversion of the mathematical expressions for various purposes, e.g. conversion of mathematical formulae to MathML [Aus+10] for indexing, exporting as well as further processing like conversion of MathML formulae to text for better accessibility.

Fatal errors detected by the input validation module — such as invalid XML metadata — prohibits data upload to the DML-CZ system. In addition, the validation module produces a variety of warnings. These cover optional parts of the data package such as source codes and possible errors that it cannot be fully to checked automatically, e.g. absence of the list of references can be an error in the case of regular article but might be completely acceptable for an editorial.

3 Processing

At the beginning of the project, the Metadata Editor [BKŠ08] was developed to enable DML-CZ team to organize a large amount of old scanned publications and label them with metadata descriptions manually. In the later stages of the project, the Metadata Editor was used mostly as an entry point for new born-digital publications provided mainly by the editorial staff of the journals included in the DML-CZ. Incoming documents are then checked and assigned to appropriate collections. The web interface of the Metadata Editor allows quick fixes and provides operators with tools for publishing final version of the documents to the public DML-CZ repository which is available to all readers at `http://dml.cz/`.

Now that new data comes in high quality directly from the publishers, the Metadata Editor is now used mainly as a control interface for the processes and as an interface to run enhancement modules on the data and metadata. During the publishing process

- articles and items of their references are checked against review databases — if a match is found, the identifier is attached to the item and presented as a direct link in the DML-CZ public repository,
- PDF full text is equipped with TEX-typeset cover page containing document metadata and providing users with the persistent DML-CZ link to the publication landing page,
- new PDFs are optimized and re-compressed for faster download and viewing in the browser,
- finally, PDF documents are marked with a digital signature and together with metadata are published in the public DML-CZ repository.

The similarity of documents is periodically recomputed over the DML-CZ content enabling the DML-CZ public repository to provide its users with an indication of the similarity of the given documents across the DML-CZ repository.

3.1 Maths Optical Character Recognition

DML-CZ participates in the European Digital Mathematics Library (EuDML) project (see Section 4). With a large proportion of the old publications scanned the necessity to make accessible versions of DML-CZ documents available to the EuDML have lead us among other reasons[2] to reprocess the DML-CZ content with the optical character recognition software (OCR). Previous version of texts extracted from the scanned images by FineReader did not contain mathematical formulae, which often bear the main semantic message in mathematical literature.

The tool used for OCR processing is InftyReader.[3] [Suz+03] This software incorporates the unique ability to recognize mathematical expressions. InftyReader accepts various bitmap image formats on the input (TIFF, BMP, GIF, PNG,

[2] Such as preparation of data for development and testing of improved mathematical documents similarity computations and maths-aware search engines.

[3] `http://www.inftyproject.org/`

PDF) and saves recognized objects in its own XML format. This rich internal representation of the document can be consequently transformed to various formats including LaTeX, XHTML+MathML and other XML formats. This transformation is a challenging task, as conversion ideally goes from presentation to content markup and disambiguation is needed.

Even though the initial Infty internal representation is common for all the different conversion export drivers (to MathML, LaTeX), the drivers seems to be of varying quality. Our main goal was the use of InftyReader generated LaTeX output that could be consequently processed similarly to the LaTeX code contained in the DML-CZ metadata, i.e. converted to the MathML by Tralics [Gri10]. However, InftyReader 2.9.5 generated LaTeX source code proved to contain several types of systematic errors that make its direct use difficult. For example, some math mode commands, such as `\ddagger`, are generated outside the math mode or the math mode itself is occasionally not opened/closed properly (missing a dollar sign). This leads to a substantial amount of subsequent errors during the processing of the rest of the LaTeX code. There is also the use of non-existent commands such as `\napos`, `\uu` instead of `\u{u}`, etc. Thus, use of the InftyReader generated TeX files results in at least one error during their processing by Tralics for more than 60% of the TeX inputs.

On the other hand, the development team of the InftyReader is willing to help us. We managed to correct several kinds of errors tweaking internal configuration tables[4] and other fixes were developed by the InftyReader team. We believe the TeX output will be significantly improved in future releases of the InftyReader.

Luckily, use of InftyReader generated XHTML+MathML seems more reliable with the current version of the transformation module. XHTML+MathML driver outputs less than 5% of invalid output files. Not being directly presented to the users these outputs seem to be good enough for internal use: indexing for similarity search, MathML to text conversion for document similarity computations, document classification and clustering.

Being available for MS Windows only, the InftyReader processing runs on separate server on a virtual machine. Batch processing was further complicated by random crashes of the InftyReader on certain input files. As it required constant monitoring, we used AutoIt software to automate attention handling required for InftyReader; we log all peculiarities of the OCR process to be reviewed only after the whole recognition batch has been processed. Running in four parallel threads on a server with today's standard hardware configuration[5], the content of the DML-CZ with more than 33,000 papers on more than 300,000 pages can be reprocessed by InftyReader-based workflow in approximately two weeks.

3.2 DOI References Parsing

An important part of scientific publications are their lists of references. The usefulness of the digital repository increase if references to other documents

[4] These tweaks were then integrated to newer releases of the InftyReader.
[5] Quad-core CPU at 2.8 GHz, 4 GiB RAM.

contains widely used unambiguous persistent identifiers, such as DOI (Digital Object Identifier), linking directly to the target of the ID.

However, not all authors use DOI or other identifiers as part of the reference and the markup used is not uniform. Moreover, DOI can be assigned to a publication arbitrarily long after the document is published, i.e. author can cite the publication long before the DOI was assigned.

Thus, a DOI look up is often the responsibility of the digital library maintainer and it is necessary to periodically look up for the existence of DOIs for documents that do not have assigned this identifier so far.

CrossRef provides tools for DOI lookup.[6] To achieve the best results, high quality markup for references is necessary, i.e. important elements such as authors, title, year of publication, journal name, publisher, pages etc. have to be properly indicated in the data.

Unfortunately, this is not the usual case. New issues of DML-CZ journals with metadata provided directly by the publishers are of reasonable quality according to the detail of markup references. However, even here we are provided with just basic 'authors — title — the rest' segmentation of the reference string by some publishers as the preparation of richly marked up metadata is a time consuming and costly operation.

A large proportion of the DML-CZ contains old, scanned publications with the only available texts obtained from OCR processing. For these papers, semiautomatic basic 'authors — title — the rest' segmentation required a vast investments in time, money and human resources during the development of the project. Even then, the quality of the metadata cannot be guaranteed. Thus, we have a great interest in the automatic segmentation of unstructured reference strings.

Our first attempt was the use of the Perl module Biblio::Citation::Parser.[7] This tool has proved to be too simple to successfully cope with various citation formats that are in common use in the DML-CZ. A very promising solution to this challenging problem seems to be the ParsCit tool.[8] [CGK08, LNK10]

For example, the plain text reference string

```
[5] Lambe, L., Stasheff, J.: Applications of perturbation theory
to iterated fibrations. Manuscripta Math. 58 (1987), 363-376.
```

is segmented by Biblio::Citation::Parser (version 1.10) as follows:

```
<authors>Lambe, L., Stasheff, J.</authors>: <title>Applications
of perturbation theory to iterated fibrations. Manuscripta Math.
58 (1987), 363-376</title>
```

It should be noted that the citation string was written without a line break. If the line break is part of the reference string, the tool fails completely. In contrast, ParsCit (version 110505) segmented the reference string as shown in Figure 2.

[6] http://www.crossref.org/guestquery/; http://help.crossref.org/#ID5824

[7] http://search.cpan.org/~mjewell/Biblio-Citation-Parser-1.10/lib/Biblio/Citation/Parser.pm

[8] http://wing.comp.nus.edu.sg/parsCit/

```
<?xml version="1.0" encoding="UTF-8"?>
<algorithms version="110505">
  <algorithm name="ParsCit" version="110505">
    <citationList>
      <citation valid="true">
        <authors>
          <author>L Lambe</author>
          <author>J Stasheff</author>
        </authors>
        <title>Applications of perturbation theory to iterated
               fibrations.</title>
        <date>1987</date>
        <journal>Manuscripta Math.</journal>
        <volume>58</volume>
        <pages>363--376</pages>
        <marker>[5]</marker>
        <rawString>Lambe, L., Stasheff, J.: Applications of
                   perturbation theory to iterated fibrations.
                   Manuscripta Math. 58 (1987), 363-376.</rawString>
      </citation>
    </citationList>
  </algorithm>
</algorithms>
```

Fig. 2. Reference segmentation done by ParsCit.

```
<?xml version="1.0" encoding="UTF-8"?>
<references xmlns:str="http://exslt.org/strings">
  ...
  <reference id="5">
    <prefix>[5]</prefix>
    <title>Applications of perturbation theory to iterated
           fibrations</title>
    <authors>
      <author>Lambe, L.</author>
      <author>Stasheff, J.</author>
    </authors>
    <journal>Manuscripta Math.</journal>
    <volume>58</volume>
    <year>1987</year>
    <pages>363-376</pages>
    <suffix>Manuscripta Math. 58 (1987), 363-376.</suffix>
  </reference>
  ...
</references>
```

Fig. 3. Example of the hand made metadata of references from the publisher.

The conversion was equally successful regardless of how many line breaks were included in the reference string. In fact, ParsCit should even be able to recognize individual parts of the full text such as title, abstract, list of references etc. For the purpose of segmenting references we use a plain text file with the string 'References' at the very beginning which is followed by the list of references. Each of the reference strings is one line with no line breaks. One blank line is used as the separator of the references.

We are currently considering using this tool for DOI lookups. Our first tests involving ParsCit as a preprocessor for CrossRef DOI look up by HTTP XML Query[9] suggest reasonable accuracy.

The CrossRef XML query schema defines a large set of elements for the structural description of various parts of the reference string and special element `<unstructured_citation>` that can contain the raw citation string. Structural elements from the ParsCit output XML can be easily transformed into the CrossRef XML query schema. It can be mostly done by renaming ParsCit XML output elements to their CrossRef XML query counterparts. Instead of a full list of authors, it proved better to use just the first author in the 'Lastname, Firstname' notation, i.e. ParsCit output

```
<authors>
  <author>L Lambe</author>
  <author>J Stasheff</author>
</authors>
```

becomes

```
<author search-all-authors="false">Lambe, L</author>
```

in the CrossRef XML query.

To provide a DOI lookup service with as much information as possible the CrossRef query is constructed not only from the structural elements ParsCit has identified in the input, but the raw citation string from the ParsCit `<rawString>` element is added to the query as the `<unstructured_citation>` element.

Hand made metadata record of our sample reference from the publisher is shown in Figure 3. Its transformation to the CrossRef XML query format is similar to the ParsCit output transformation and the resulting XML query varies in small details such as punctuation only.

As a basic test we have used articles of volume 48, issue 5 of the *Archivum Mathematicum* journal (`http://dml.cz/handle/10338.dmlcz/143106`) that was published at the end of 2012. As we have high quality metadata for references from the publisher of the journal we compared the results of the DOI lookup using these hand made metadata and automatically segmented plain text reference strings harvested from the landing pages of the articles.

As can be seen in the summary in Table 1 and Figure 4 both hand made metadata and ParsCit generated metadata led to a surprisingly similar result of

[9] `http://help.crossref.org/#ID5829`

Table 1. Comparison of DOI lookup results using hand made and ParsCit automatically segmented references from the articles of *Archivum Mathematicum*, volume 48, issue 5.

	number of refs.	ParsCit		hand made	
		resolved	not resolved	resolved	not resolved
article #2	10	1 (10.00%)	9 (90.00%)	2 (20.00%)	8 (80.00%)
article #3	6	3 (50.00%)	3 (50.00%)	3 (50.00%)	3 (50.00%)
article #4	19	4 (21.05%)	15 (78.95%)	4 (21.05%)	15 (78.95%)
article #5	12	3 (25.00%)	9 (75.00%)	3 (25.00%)	9 (75.00%)
article #6	17	6 (35.29%)	11 (64.71%)	6 (35.29%)	11 (64.71%)
article #7	16	6 (37.50%)	10 (62.50%)	4 (25.00%)	12 (75.00%)
article #8	9	3 (33.33%)	6 (66.67%)	3 (33.33%)	6 (66.67%)
article #9	10	1 (10.00%)	9 (90.00%)	2 (20.00%)	8 (80.00%)
article #10	13	5 (38.46%)	8 (61.54%)	5 (38.46%)	8 (61.54%)
article #11	13	2 (15.38%)	11 (84.62%)	1 (7.69%)	12 (92.31%)

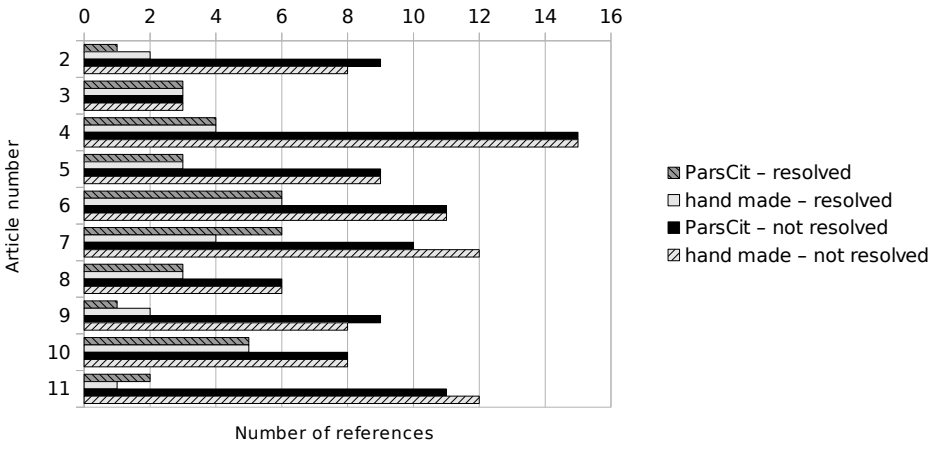

Fig. 4. Visualization of DOI lookup results from Table 1.

CrossRef DOI lookup.[10] ParsCit always correctly identified all the references and segmented them well enough to achieve results comparable to the hand made metadata. Further investigation of the possibilities of deployment of this tool will be of great interest to us in the near future.

4 Output Modules

The content of the DML-CZ digital library is collected to be read and used. To achieve this, it has to be visible and usable. To achieve high visibility, it has to be indexed by search engines, especially by Google, given that it is used for 85% of searches today.

DML-CZ is available to the outside world via the DSpace repository software. This includes the end user interface (classic web based on HTML/JavaScript) and the OAI-PMH server providing the DML-CZ metadata together with links to the DML-CZ data in various XML formats. [Kre08] This section describes the complete workflow from the DML-CZ internal metadata to the EuDML specific NLM formats exported via OAI-PMH. The last subsection discusses the cooperation with Google Scholar.

4.1 Metadata Transformation

The project of the European Digital Mathematics Library EuDML [Syl+10], `http://eudml.org/`, is based on metadata and data of smaller regional DML projects. It was realized that almost every EuDML content provider uses a *different* internal format for their holdings. For example, the DML-CZ internal metadata format was established during the development of DML-CZ several years before the EuDML project was started. To adopt the EuDML format to be used by the DML-CZ internal tools would be quite difficult and time consuming, causing troubles in the well established DML-CZ workflow and would create a lot more work on the publishers' side. Now that the DSpace OAI-PMH is able to provide reliable metadata and their transformations into various formats, we took advantage of it.

Thus, a great deal of efforts went into mapping the metadata into the unique metadata format that is required for central processing and global enhancement methods.

The basic schema stands on the OAI-PMH and assumes that local repositories make their metadata available via an OAI server and are harvested by the EuDML central repository. There are now two ways of exposing the local metadata to the EuDML.

The first possibility is to expose metadata in an internal format which local DML repositories use natively. Metadata is harvested in this format and on the harvester side transformed into the specific EuDML metadata format. This approach puts almost no demands on the local repositories and most of the work is done on the EuDML side.

[10] Articles #1 and #12 are not present as they do not contain any references.

The second option, used in DML-CZ, is to expose metadata in the EuDML specific formats. For this purpose two formats have been set up — one for journals (*eudml-article2*[11]) and one for books (*eudml-book2*[12]). This assumes most of the work has to be done on the local repository side which can however bring some benefits — metadata has to be validated, errors have to be identified and corrected. This process leads to improved local repository metadata. For the DML-CZ this approach has been adopted.

Both *eudml-article2* and *eudml-book2* formats are based on the NLM Journal Archiving and Interchange Tag Suite format (version 3.0) [Dig08]. The NLM format is suitable for describing journal articles and is used without any changes in EuDML. To describe books, the NLM format has been extended and several new tags have been added. In the rest of the text both formats are referred to just NLM. [NIS12]

As the metadata for most of the DML-CZ content is very static and hardly ever changes, the transformations from DML-CZ internal XML metadata [BKŠ08] into the NLM are done in a batch with the help of XSL transformation. The result is a not a fully compatible NLM (pre-NLM) file stored directly next to journal/book metadata file in the internal structures. The file is not in the final NLM format because there are certain kinds of dynamic information (e.g. links to fulltexts) that have to be added on-the-fly by OAI-PMH at the moment the metadata are requested.

The XSL 2.0 transformations are used to obtain the NLM format from the internal format, including EuDML specific XSL functions for handling metadata like language codes. As a transformer the Saxon is used. The transformation process is integrated into the internal tools via set of bash scripts.

4.2 OAI-PMH

The pre-NLM file is then stored (on demand or automatically during import/ /update operations) in the DSpace repository. A DSpace digital object (called *Item*) schema allows various kinds of files to be stored. These are logically separated into so called *Bundles*. The pre-NLM file is stored in such a specific *Bundle* and used later by the DML-CZ OAI-PMH server.

DSpace OAI-PMH server is fully and easily configurable and provides various methods (XSL crosswalks, plugins in Java) how to add a custom format. A special Java plugin for exposing NLM formats has been developed. The plugin is called when metadata for an *Item* (or *Record* in terms of OAI-PMH) are requested on the OAI-PMH server. The plugin loads stored pre-NLM XML and adds links to fulltexts. These links cannot be added during the first transformation phase (pre-NLM) because at that time the information about the publisher's moving wall is not known (articles behind the publisher's moving wall can be used only for indexing and not for exposing fulltexts). The resulting XML is in the final NLM format and is served in an OAI-PMH `<record>` element.

[11] Namespace `http://jats.nlm.nih.gov`.
[12] Namespace `http://eudml.org/schema/2.0/eudml-book`

However, the way DSpace works with OAI-PMH *Sets* in the default configuration is not meaningful in DML-CZ, because the DSpace core structures represented by *Community* and *Collection* objects are handled in a different semantic way in the DML-CZ. The *Sets* are treated as the *Collections* which represent journal issues in the DML-CZ. The DSpace has been patched to change the *Sets* to be top level *Communities*, thus the *Sets* represent whole journals, proceedings series and monographs collections (as can be seen at the DML-CZ homepage). The patch includes a new database index table of articles and chapters for the top communities in DSpace and necessary code changes to work with it.

4.3 Google Scholar

The connection to Google Scholar is made in the way they recommend — the HTML header `<meta>` tags are used to fill up necessary article/chapter metadata. While there is no precise specification of the `citation_` format, the example provided by Google is followed. Every HTML page in DML-CZ is generated on-the-fly via XSL so the `<meta name="citation\_(spec)" content="(value)"/>` tags are processed the same way. Indexing these `<meta>` tags allows Google Scholar to link directly to the fulltexts in DML-CZ without the necessity of parsing paper metadata from landing HTML pages and PDFs. We believe that agreement on this interface might contribute slightly to the Page ranking of papers, as the metadata are contributed from the verified DML-CZ source.

Looking at the Google Analytics statistics over the last five years, the ratio of DML-CZ traffic generated by Google searches continually increases, reaching more than 85% at the time of writing. This shows the importance of this export interface, together with the sitemap updates we regenerate regularly and thus point Google to the newly published items automatically. DML-CZ is now ranked among the top ten repositories in Central and Eastern Europe, and best repository in the Czech Republic, measured by `http://repositories.webometrics.info/`.

5 Conclusions

We have reported on the workflows, interfaces and modules we have developed for the low-cost running of DML-CZ. When agreeing on formal interfaces that could be enforced by validation, considerable savings of manual work have been achieved, while in parallel increasing data quality and services of the library. After introducing the modules described, there is almost no manual intervention necessary. We perform manual checks of uploaded data before sending them to the public library but this is not strictly necessary.

Exporting data via the agreed interfaces to Google and EuDML skyrocketed the visibility of DML-CZ repository content, as measured by `webometrics.info` or similar metrics. We believe that our DML-CZ example demonstrates that by maintaining solid information technologies, Computer Science methodologies and web standards, even a small digital library can be run at a moderate cost.

Our future plans include adding further machine-actionable modules and functionalities into DML-CZ. Having full texts with math formulae by math OCR now, it is natural to add formulae searching with our Math Indexer and Searcher system MIaS [SL11], which already works well in EuDML. One of the most challenging remaining issues is the improvement of the process of the automated OCR of mathematics and its tighter integration into the rest of the system.

Also, new EuDML external APIs will be employed, namely for article similarity — similar articles will be acquired from the EuDML set instead of from a local set of articles in DML-CZ only. We also expect extensive development of the module for automatic parsing of the references. We hope we could significantly improve the quality of references metadata and eliminate the necessity of their costly and inefficient manual corrections.

Acknowledgements This work was partially supported by the European Union through its Competitiveness and Innovation Programme (Information and Communication Technologies Policy Support Programme, 'Open access to scientific information', Grant Agreement No. 250503, a project of the European Digital Mathematics Library, EuDML).

References

[Aus+10] Ron Ausbrooks et al. *Mathematical Markup Language (MathML)*. Ed. by David Carlisle, Patrick Ion, and Robert Miner. Version 3.0. W3C Recommendation 21 October 2010. World Wide Web Consortium (W3C). 2010-10-21. URL: http://www.w3.org/TR/2010/REC-MathML3-20101021/ (visited on 2013-01-06).

[BKŠ08] Miroslav Bartošek, Petr Kovář, and Martin Šárfy. "DML-CZ Metadata Editor: Content Creation System for Digital Libraries". In: *Towards a Digital Mathematics Library*. Ed. by Petr Sojka. Birmingham, UK: Masaryk University, 2008-07, pp. 139–151. ISBN: 978-80-210-4658-0. URL: http://dml.cz/dmlcz/702537 (visited on 2013-01-09).

[CGK08] Isaac G. Councill, C. Lee Giles, and Min-Yen Kan. "ParsCit: An open-source CRF reference string parsing package". In: *Language Resources and Evaluation Conference (LREC 08)*. Marrakesh, Morocco, 2008-05. URL: http://www.comp.nus.edu.sg/~kanmy/papers/lrec08b.pdf (visited on 2013-03-13).

[Dig08] Digital Archive of Journal Articles National Center for Biotechnology Information (NCBI) and National Library of Medicine (NLM). *NCBI Book Tag Library version 3.0*. 2008-11. URL: http://dtd.nlm.nih.gov/book/.

[Gri10] José Grimm. "Producing MathML with Tralics". In: *Towards a Digital Mathematics Library*. Ed. by Petr Sojka. Paris, France: Masaryk University, 2010-07, pp. 105–117. ISBN: 978-80-210-5242-0. URL: http://dml.cz/dmlcz/702579 (visited on 2013-01-09).

[Kre08] Vlastimil Krejčíř. "Building Czech Digital Mathematics Library upon DSpace System". In: *Towards a Digital Mathematics Library*. Ed. by Petr Sojka. Birmingham, UK: Masaryk University, 2008-07, pp. 117–126. ISBN: 978-80-210-4658-0. URL: http://dml.cz/dmlcz/702539 (visited on 2013-01-09).

[LNK10] Minh-Thang Luong, Thuy Dung Nguyen, and Min-Yen Kan. "Logical Structure Recovery in Scholarly Articles with Rich Document Features". In: *International Journal of Digital Library Systems* 4 (1 2010). Forthcoming., pp. 1–23. DOI: 10.4018/jdls.2010100101. URL: http://www.comp.nus.edu.sg/~kanmy/papers/ijdls-SectLabel.pdf (visited on 2013-03-13).

[NIS12] National Information Standards Organization NISO. *JATS: Journal Article Tag Suite, ANSI/NISO Z39.96-2012*. 2012-08. URL: http://jats.niso.org/.

[RS10] Michal Růžička and Petr Sojka. "Data Enhancements in a Digital Mathematics Library". In: *Towards a Digital Mathematics Library*. Ed. by Petr Sojka. Paris, France: Masaryk University, 2010-07, pp. 69–76. ISBN: 978-80-210-5242-0. URL: http://dml.cz/dmlcz/702575 (visited on 2013-01-13).

[RS11] Michal Růžička and Petr Sojka. "Redakční systém odborného časopisu s podporou exportu do digitální knihovny v MathML". In: *Zpravodaj CSTUG* (1 2011), pp. 4–20. DOI: 10.5300/2011-1/4.

[Růž08] Michal Růžička. "Automated Processing of TEX-typeset Articles for a Digital Library". In: *Towards a Digital Mathematics Library*. Ed. by Petr Sojka. Birmingham, UK: Masaryk University, 2008-07, pp. 167–176. ISBN: 978-80-210-4658-0. URL: http://dml.cz/dmlcz/702533 (visited on 2013-01-13).

[SL11] Petr Sojka and Martin Líška. "The Art of Mathematics Retrieval". In: *Proceedings of the ACM Conference on Document Engineering, DocEng 2011*. Mountain View, CA: ACM, 2011-09, pp. 57–60. ISBN: 978-1-4503-0863-2. DOI: 10.1145/2034691.2034703.

[Soj08] Petr Sojka, ed. *Towards a Digital Mathematics Library*. Birmingham, UK: Masaryk University, 2008-07. ISBN: 978-80-210-4658-0. URL: http://dml.cz/dmlcz/702564 (visited on 2013-01-13).

[Soj10] Petr Sojka, ed. *Towards a Digital Mathematics Library*. Paris, France: Masaryk University, 2010-07. ISBN: 978-80-210-5242-0. URL: http://dml.cz/dmlcz/702567 (visited on 2013-01-13).

[Suz+03] Masakazu Suzuki, Fumikazu Tamari, Ryoji Fukuda, Seiichi Uchida, and Toshihiro Kanahori. "INFTY—An integrated OCR system for mathematical documents". In: *Proceedings of ACM Symposium on Document Engineering 2003*. Ed. by C. Vanoirbeek, C. Roisin, and E. Munson. Grenoble, France: ACM, 2003, pp. 95–104.

[Syl+10] Wojtek Sylwestrzak, José Borbinha, Thierry Bouche, Aleksander Nowiński, and Petr Sojka. "EuDML—Towards the European Digital Mathematics Library". In: *Towards a Digital Mathematics Library*. Ed. by Petr Sojka. Paris, France: Masaryk University, 2010-07, pp. 11–24. ISBN: 978-80-210-5242-0. URL: http://dml.cz/dmlcz/702569 (visited on 2013-01-13).