Toolset for image and text processing and metadata enhancement - Value release

Petr Sojka, Krzysztof Wojciechowski, Nicolas Houillon, Michal Ružicka, Radim Hatlapatka, Romeo Anghelache, Josef Baker, Łukasz Bolikowski, Thierry Bouche, Michael Jost, et al.

To cite this version:


HAL Id: hal-03765970
https://hal.univ-grenoble-alpes.fr/hal-03765970
Submitted on 31 Aug 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
### Project Acronym: EuDML

Grant Agreement number: 250503

Project Title: The European Digital Mathematics Library

---

**D7.3: Toolset for Image and Text Processing and Metadata Enhancements — Value release**

Revision: 1.01 as of 8th March 2012

---

**Authors:**

- Petr Sojka, Masaryk University, MU
- Krzysztof Wojciechowski, ICM Warsaw, ICM
- Nicolas Houillon, UJF/CMD Grenoble
- Michal Růžička, Masaryk University, MU
- Radim Hatlapatka, Masaryk University, MU

**Contributors:**

- Romeo Anghelache, FIZ Karlsruhe, FIZ
- Josef Baker, University of Birmingham, UB
- Łukasz Bolikowski, ICM Warsaw, ICM
- Thierry Bouche, UJF/CMD Grenoble
- Michael Jost, FIZ Karlsruhe, FIZ
- Aleksander Nowiński, ICM Warsaw, ICM
- Gilberto Pedrosa, IST Lisbon, IST
- Michal Růžička, Masaryk University, MU
- Alan Sexton, University of Birmingham, UB

---

<table>
<thead>
<tr>
<th>Dissemination Level</th>
<th>P</th>
<th>Public</th>
<th>✓</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>Confidential, only for members of the consortium and the Commission Services</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Author</th>
<th>Organisation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>Jan 11th, 2012</td>
<td>Petr Sojka</td>
<td>MU</td>
<td>First version with structure of the deliverable.</td>
</tr>
<tr>
<td>0.2</td>
<td>Jan 12th, 2012</td>
<td>Petr Sojka</td>
<td>MU</td>
<td>WP7 videoconference agreements added into the text, added WP7 workflow diagram, and <code>PdfToText+MathViaOCR</code>.</td>
</tr>
<tr>
<td>0.3</td>
<td>Jan 18th, 2012</td>
<td>Petr Sojka</td>
<td>MU</td>
<td>Intermediate version with some input from IST, part of text drafted.</td>
</tr>
<tr>
<td>0.31</td>
<td>Jan 20th, 2012</td>
<td>Petr Sojka</td>
<td>MU</td>
<td>Specification of TODOs, workflow described.</td>
</tr>
<tr>
<td>0.32</td>
<td>Jan 23rd, 2012</td>
<td>Petr Sojka</td>
<td>MU</td>
<td>Specification of TODOs (cont.).</td>
</tr>
<tr>
<td>0.33</td>
<td>Jan 24th, 2012</td>
<td>Petr Sojka</td>
<td>MU</td>
<td><code>PDFTester</code> info added.</td>
</tr>
<tr>
<td>0.34</td>
<td>Jan 25th, 2012</td>
<td>Josef Baker</td>
<td>BU</td>
<td>BU tools section updated.</td>
</tr>
<tr>
<td>0.4</td>
<td>Jan 26th, 2012</td>
<td>Petr Sojka</td>
<td>MU</td>
<td>FIZ tools description updated.</td>
</tr>
<tr>
<td>0.5</td>
<td>Jan 27th, 2012</td>
<td>Petr Sojka</td>
<td>MU</td>
<td>Matching tool description updated.</td>
</tr>
<tr>
<td>0.6</td>
<td>Jan 29th, 2012</td>
<td>RH</td>
<td>MU</td>
<td><code>PdfToTextViaOCR</code>, <code>PdfJbIm</code> and <code>Pdfsizeopt</code> updated.</td>
</tr>
<tr>
<td>0.7</td>
<td>Jan 30th, 2012</td>
<td>PS, RH</td>
<td>MU</td>
<td>Further text updates and cuts, section on evaluation added.</td>
</tr>
<tr>
<td>0.8</td>
<td>Jan 31st, 2012</td>
<td>Petr Sojka</td>
<td>MU</td>
<td>Preparation for internal review.</td>
</tr>
<tr>
<td>0.81</td>
<td>Feb 5th, 2012</td>
<td>AS, PS</td>
<td>UB, MU</td>
<td>Preliminary review corrections and additions.</td>
</tr>
<tr>
<td>0.82</td>
<td>Feb 27th, 2012</td>
<td>NH, MR, PS</td>
<td>UJF, MU</td>
<td>Tools and workflow issues update, . . .</td>
</tr>
<tr>
<td>0.9</td>
<td>Feb 29th, 2012</td>
<td>PS</td>
<td>MU</td>
<td>Version for final internal review.</td>
</tr>
<tr>
<td>0.91</td>
<td>Mar 3rd, 2012</td>
<td>PS</td>
<td>MU</td>
<td>Decisions from Grenoble technical meeting incorporated.</td>
</tr>
<tr>
<td>1.0</td>
<td>Mar 5th, 2012</td>
<td>PS, AS</td>
<td>MU</td>
<td>Final release with Alan’s review.</td>
</tr>
<tr>
<td>1.01</td>
<td>Mar 8th, 2012</td>
<td>PS</td>
<td>MU</td>
<td>Minor edit about fixed part of the workflow.</td>
</tr>
</tbody>
</table>

### Statement of originality:

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.
Contents

1 Introduction
   1.1 The Toolset Structure ........................................... 3
   1.2 Structure of the Demo Description ............................... 5

2 Eutools
   2.1 PdfToTextViaOCR .............................................. 5
   2.2 PdfToText+MathViaOCR ......................................... 6
   2.3 PdfTester ...................................................... 6
   2.4 PDF Text Extractor ............................................ 7
   2.5 MathML Extractor (PDF2MML) .................................. 7
   2.6 TeX2NLM ....................................................... 7
   2.7 EnhanceNLMTeXwMML ........................................... 9
   2.8 Plain Text Reference Segmenter ................................. 12
   2.9 Bibliographic Reference Parser ................................ 12
   2.10 ZBMath Metadata Lookup Service .............................. 13
   2.11 ZBMath Reference Matching Service ......................... 16
   2.12 PdfJblm ...................................................... 18
   2.13 Pdfsizeopt .................................................. 18
   2.14 math_metadata_lookup ....................................... 19
   2.15 Metadata Editor with Export to NLM ......................... 19

3 Integration of Eutools into the EuDML Enhancement Workflow .... 20
   3.1 Workflow Evaluation ........................................... 22

4 Summary, Conclusions
   4.1 Roadmap to Working Data Enhancement System ................ 23
   4.2 Future Work .................................................... 26

Index ............................................................... 29
Executive Summary

This demonstration description presents tools and partial workflow results produced by EuDML [partners] and made available for demonstration. They demonstrate enhancement tools, whose functionality should find, check, merge, correct and enhance metadata and full texts collected both from partners, including Zentralblatt MATH, and from the analysis of full text or PDF document versions of items in the EuDML collection. Demonstration web pages allow testing and evaluation of fourteen tools.

http://demo.eudml.eu/demo/
http://www.zentralblatt-math.org/eudml/match?query=524, Inhomogeneous fixed point ensembles, 1811
http://thar.ujf-grenoble.fr/EuDML/demo/NLMTeX2TeX+MML/
http://thar.ujf-grenoble.fr/EuDML/demo/TeX2NLM/
http://www.cs.bham.ac.uk/go/sdag/maxtract.php
http://server.bd2.inesc-id.pt/infyEudml/
http://wysoka.icm.edu.pl:18190/EudmlAnalysisDemo/
1 Introduction

This demonstration deliverable describes current version of a toolset for image and metadata enhancement and editing. The tools needed were identified in D7.1 [24] and initial version of the toolset has been described in D7.2 [26]. This demonstration presents the current status of tools developed and tested not only at the technology provider’s sites, but also partially integrated and used on the real EuDML data from data providers.

The purpose of this demonstration is to verify and discuss functionality, usability, scalability and effectiveness of every tool in a comprehensive workflow integrated in EuDML enhancement subsystem.

Let us recall the top-level enhancement tool structure, with subsystems represented as edges on Figure 1.

![Figure 1: Metadata processing](image)

1.1 The Toolset Structure

The toolset consists of five subsystems: OCR, Extraction, Analysis, Conversion and Refinement, and a set of external tools offered to data providers (External). Subsystems are built based on the smaller bricks of software — eutools — as defined in [24, Section 1.3]. Eutools are being developed and tested mostly at the technology providers’ sites, with well defined interfaces allowing their integration into subsystems on the EuDML core system site. The toolset currently consists of the eutools listed in Table 1 on the following page.

A very important issue is the interface with which the eutools will communicate within WP7 with other eutools and with the EuDML core system. Possible and preferred interfaces to be used within EuDML were listed in [26, Section 2 of D7.2].
Table 1: Eutools overview as of 8th March 2012

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Partner</th>
<th>Eutool</th>
<th>Functionality</th>
</tr>
</thead>
<tbody>
<tr>
<td>OCR</td>
<td>MU</td>
<td>PdfToTextViaOCR</td>
<td>Basic plaintext extraction from bitmap images which are rendered from a PDF document.</td>
</tr>
<tr>
<td>OCR</td>
<td>IST</td>
<td>PdfToText+MathViaOCR</td>
<td>Plaintext and Math extraction from bitmap images which are rendered from a PDF document.</td>
</tr>
<tr>
<td>Extraction</td>
<td>UB</td>
<td>PDFTester</td>
<td>Tests whether a PDF document contains multiple layers, page bitmaps and/or is born-digital, and therefore what further processing is possible and appropriate.</td>
</tr>
<tr>
<td>Extraction</td>
<td>ICM</td>
<td>PDF Text Extractor</td>
<td>Extracts plain text from a PDF document.</td>
</tr>
<tr>
<td>Extraction</td>
<td>UB</td>
<td>PDF2MML</td>
<td>Analyses a PDF document and extracts the mathematical expressions from it as a list of MathML structures.</td>
</tr>
<tr>
<td>Conversion CMD</td>
<td></td>
<td>TeX2NLM</td>
<td>Identifies (\text{\TeX}) formulas in a character string and replaces each with an NLM structure with both (\text{\TeX}) and MathML alternatives.</td>
</tr>
<tr>
<td>Conversion CMD</td>
<td></td>
<td>EnhanceNLMTeXwMML</td>
<td>Takes an NLM document and 1) adds a MathML alternative to each formula represented by the NLM formula structure that has a (\text{\TeX}) version but not MathML, and 2) finds textual (\text{\TeX}) formula and replaces each with a NLM formula structure containing both the original (\text{\TeX}) and its MathML equivalent.</td>
</tr>
<tr>
<td>Analysis</td>
<td>ICM</td>
<td>Plain Text Reference Segmenter</td>
<td>Extracts bibliographic references from plain text.</td>
</tr>
<tr>
<td>Analysis</td>
<td>ICM</td>
<td>Bibliographic Reference Parser</td>
<td>Parses a plain text bibliographic reference (extracts author names, title, publication year, etc.).</td>
</tr>
<tr>
<td>Refinement FIZ</td>
<td></td>
<td>ZBMath Lookup Service</td>
<td>Get [\text{Zbl}] metadata for a publication given its Zentralblatt identifier.</td>
</tr>
<tr>
<td>Refinement FIZ</td>
<td></td>
<td>ZBMath Match Service</td>
<td>Get [\text{ZblID}] and [\text{Zbl}] basic metadata for a publication given its bibliography reference as string.</td>
</tr>
<tr>
<td>Refinement MU</td>
<td></td>
<td>PdfJbIm</td>
<td>Recompress bitmap streams in a PDF document with JBIG2.</td>
</tr>
<tr>
<td>Refinement MU</td>
<td></td>
<td>Pdfsizeopt</td>
<td>Optimize PDF documents for size.</td>
</tr>
<tr>
<td>Refinement MU</td>
<td></td>
<td>math_metadata_lookup</td>
<td>MR/Zbl metadata search and fetch.</td>
</tr>
<tr>
<td>External MU</td>
<td>MU</td>
<td>ME</td>
<td>Metadata Editor—standalone editing for use at providers’ sites.</td>
</tr>
</tbody>
</table>
1.2 Structure of the Demo Description

Section 2 lists eutools that have been prototypically implemented, together with the web pages that demonstrate their functionality. Section 3 describes WP7 workflow—current successes with tools integration and possible further enhancements feasible to be implemented in the D7.4. Section 4 sums up demonstrated tools, achievements and reviews future direction of development of EuDML enhancers and issues to be tackled yet.

2 Eutools

In this section basic information is provided for every tool, including their defined input and output interfaces, license information, programming language and evaluation, as well as the URL of the demonstration web site.

Several tools described in this section are so-called processing nodes, which can be chained together into so-called processes. The initial node in a process typically generates or otherwise obtains chunks of data which are consecutively processed by the following nodes. A node typically enhances the chunks that it receives on its input and sends the enhanced chunks to its output, possibly with side effects such as indexing the contents of the chunk. The final node in a process typically stores the enhanced chunk in storage or discards it. There is a processing framework written in Java which orchestrates the flow of data chunks between nodes. Therefore, an author of an individual tool only needs to implement a processing node with well-defined inputs and outputs.

The tools are integrated one by one by adding them into the enhancement workflow described in the Section 3. Some tools are also used during the metadata ingestion REPOX phase as part of data enhancements to validate.

2.1 PdfToTextViaOCR

PdfToTextViaOCR is a tool written in Java which renders images from PDF and extracts text from them using an OCR engine. As OCR engine is used Tesseract [15], which is licensed under Apache License 2.0 and supports wide variety of languages. For further details about PdfToTextViaOCR see D7.2 [26].

PdfToTextViaOCR is one of a set of tools for extracting text from PDF documents which is further used for indexing and searching inside EuDML, and to provide text to other WP7 or WP10 tools (Braille drivers etc.). It is used as a fallback solution in cases where use of PdfToText+MathViaOCR is not applicable.

Tesseract is called from a processing node via a shell script which handles setting paths to necessary shared libraries and to language data used by Tesseract. This is to allow using Tesseract without requiring its system-wide installation.

Language used by Tesseract is set for each document separately based on metadata found in NLM. The language code found in NLM is transformed to a language code accepted by Tesseract. If such language isn’t supported by Tesseract, the default one is used.

---

1. [http://www.apache.org/licenses/LICENSE-2.0](http://www.apache.org/licenses/LICENSE-2.0)
A demonstration version of this tool, which is modified to be a standalone tool, is made available at [http://nlp.fi.muni.cz/projekty/eudml/eudmldemo.php](http://nlp.fi.muni.cz/projekty/eudml/eudmldemo.php).

### 2.2 PdfToText+MathViaOCR

PdfToText+MathViaOCR is an integrated mathematical document reader system that uses Infty system [29, 17] for OCR (Optical Character Recognition). The system recognizes PDFs with scanned page images of scientific documents, including mathematical expressions, and outputs the results in \( \LaTeX \), HTML, XML etc. It was developed for retro-digitization of mathematical journals and for accessibility support such as automatic transcription of scientific documents into Braille codes.

After harvesting the full-text content of the digital records using REPOX, the system will identify those which contain only bitmap page images via PDFTester, and use the PdfToText+MathViaOCR to convert those images which contain mathematical expressions into plain text and store the result in the EuDML storage system. The interface used for passing the results to the EuDML core is HTTP protocol.

A demonstration version of this tool is available at [http://server.bd2.inesc-id.pt/inftyEudml/](http://server.bd2.inesc-id.pt/inftyEudml/).

### 2.3 PdfTester

PDFTester is a tool written in OCaml that determines how a given PDF file should be processed. The software requires an uncompressed PDF file which is currently created via a call to pdftk, which is licensed under the GNU General Public License Version 2. There are three possible outcomes of running the tester, namely:

- The PDF contains multiple layers which should be extracted automatically.
- The PDF contains sufficient information to automatically extract contents and generate MathML.
- The PDF should be processed via rendering to an image and the use of OCR.

The first check to identify multiple layers within the PDF is completed by reading the content streams within the file and searching for marked content. If marked content is found then the tool will return an integer indicating that the file has multiple layers.

The second check to identify the compatibility of the file with the PDF Text Extractor tool is completed by extracting the fonts and content streams from the file. If these can be parsed successfully, then an integer is returned indicating that the file is compatible.

If both of these checks fail, then the file is unsuitable for enhanced analysis by the current EuDML tool set. Therefore an integer is returned indicating it should be rendered to an image and processed via the PdfToTextViaOCR or PdfToText+MathViaOCR tool.

PDFTester has been run on more than 60,000 EuDML files already and has been integrated into WP7 workflow as a statically linked binary. Efficiency issues and problems with PDFs generated by Pdfsizeopt and other programs were identified and will be fixed in the Deliverable D7.4.
2.4 PDF Text Extractor

**PDF Text Extractor** is written in Java, and uses the Apache PDFBox library [22] to obtain plain text from a PDF document. PDFBox is open source software distributed under Apache License Version 2.0.

For more detailed information we refer to D7.2 [26]. A joint demonstration of text extraction from PDF documents, bibliographic reference extraction from plain text, and bibliographic reference parsing is available at [http://wysoka.icm.edu.pl:18190/EudmlAnalysisDemo/](http://wysoka.icm.edu.pl:18190/EudmlAnalysisDemo/).

2.5 MathML Extractor (PDF2MML)

MathML extraction from PDF documents are handled by a tool written in OCaml that returns the full page text of a PDF document in XHtml format with mathematical expressions embedded as MathML. In addition drivers are also available to produce LATEX, plain text and annotated PDFs. The software requires an uncompressed PDF file which is currently created via a call to the free software tool pdftk.

Given an appropriate PDF file, of which the suitability can be determined by running **PDFTester**, the full page content of the whole document can be extracted. The tool works by extracting the fonts and content streams from a PDF which are then parsed by the PDF2MML, producing a list of symbols and graphics for each page. These, in conjunction with a list of glyphs obtained via image analysis of the page images rendered from the PDF document, are used to split each page into a number of lines. Each line is parsed to create a parse tree, then processed by a driver that separates text from in-line math expressions and produces MathML markup for any formulae that occur on that line. The software is based upon the work described in [4, 3, 5, 6].

A limitation of the tool is that it can only work with PDF files making use of Type 1 fonts and embedded font encodings. This generally means that the file will have been generated from LATEX, Troff, Scientific Word or a number of other document production systems, which does limit the number of potential sources. Also, the segmentation process is still in development and analysis of page layout elements such as headers, footers, section headings etc., may be sub-optimal. These issues, in particular the compatibility with certain PDFs will be addressed in subsequent versions and be available by Deliverable D7.4.


2.6 TeX2NLM

The structure of the tools called TeX2NLM and NLMTeX2TeX+MML in the previous report has been reworked to allow for a wider range of applications, in particular to make the \TeXX to MathML conversion available to other components of the system.

TeX2NLM is now a library that, even if it still requires a working Tralics, is otherwise free of dependencies. Additionally, a linux version of Tralics with a set of configuration files is provided. This tool was made to be used primarily by EnhanceNLM-TeXwMML, that adds a MathML versions of \TeXX formulas in NLM documents (see Section 2.7 for more details). It is also expected that a formula search function will use...
this tool, by converting \( \text{\LaTeX} \) entered by the website user to MathML on the fly, and then using the EuDML tool for Mathematical Indexing and Search (MIsA) \cite{cit1, cit2, cit3} to build a search query from the MathML.

\( \text{\LaTeX2NLM} \) takes, as input, an UTF-8 encoded string of characters that is expected to be valid \( \text{\LaTeX} \) code and returns the same content with formulas identified as such and conforming to EuDML NLM structure (a \textless *-formula} \textgreater element—i.e. a \textless disp-formula} \textgreater or \textless inline-formula} \textgreater element—for each formula, containing an \textless alternatives} \textgreater element with a child \textless tex-math} \textgreater element holding the original \( \text{\LaTeX} \) code, and a \textless mml:mml} \textgreater element with a MathML representation of the formula derived from the provided \( \text{\LaTeX} \).

The tool is written in Java and is provided as a standalone library that embeds a linux version of Tralics, but can be configured to use another version, appropriate to the system the tool is executed on (version 2.14 or higher is required). It also provides a set of configuration files, and more can be added. Once configured properly, this becomes completely transparent for the rest of the system.

This library is free software governed by the CeCILL-C license that can be found at http://www.cecill.info/.

The assumptions for this tool are the following:

- It is applied to an UTF-8 encoded string of characters (typically, the content of a childless XML element holding textual information with \( \text{\LaTeX} \)-encoded mathematical formulas).
- The \( \text{\LaTeX} \) commands switching to math mode must be explicit in the \( \text{\LaTeX} \) code as $ in the supplied example (it could also be \textbackslash[..\textbackslash], \textbackslash begin\{align\}, etc.);
- Once unescaped, the \( \text{\LaTeX} \) code contains no unspecified macros and compiles with allowed and configured \( \text{\LaTeX} \) commands.

The tool identifies each formula in the input string, generates a standard NLM structure for each of them, and returns a Java DOM Element containing the result.

Example: An input string with a \( \text{\LaTeX} \) formula

The formula $J_k(n) := n^k \prod_{p \mid n}(1 - p^{-k})$ clearly defines nothing.

The tool’s output for this example

The formula

\begin{verbatim}
<inline-formula id="d18e3149">
  <alternatives>
    <mml:mml xmlns="http://www.w3.org/1998/Math/MathML">
      <mml:msub>
        <mml:mi>J</mml:mi>
        <mml:mi>k</mml:mi>
      </mml:msub>
      <mml:mtext>_n</mml:mtext>
      <mml:mo>:</mml:mo>
      <mml:mo>=</mml:mo>
      <mml:mrow>
        <mml:mml>
          <mml:msub>
            <mml:mi>n</mml:mi>
            <mml:mo>^</mml:mo>
          </mml:msub>
          \prod_{p \mid n}\{1 - p^{-k}\}
        </mml:mml>
      </mml:mrow>
    </mml:mml>
  </alternatives>
</inline-formula>
\end{verbatim}
The previously considered method using heuristics to detect formulae has been dismissed, mainly because of the way tralics works for input and output. Tralics requires that the TEX to be converted to MathML be written to a file on the file system, which it then reads, converts and writes to another file. This result file has to be read by TeX2NLM and parsed so that the data can finally be used. Giving a complete character string from the original document to Tralics, for example an abstract with several formulae, and letting Tralics determine what is a formula is more efficient and robust than attempting to pre-determine where each formula is and writing each of them to a file for Tralics to convert.

A demonstration prototype of this tool is available at http://thar.ujf-grenoble.fr/EuDML/demo/Tex2NLM/.

2.7 EnhanceNLMTeXwMML

EnhanceNLMTeXwMML takes, as input, a valid XML file that can contain mathematical formulae both as textual TEX formulae within the text data of various element, or formatted as <inline-formula> or <disp-formula> element with its internal EuDML v1.0 DTD structure [16], containing a TEX-encoded version of the formula in the <tex-math> element. It returns the same file with unchanged content except that a <mml:math>
element is added as an alternative within `<*-formula>`, with a MathML representation of the formula derived from the provided \(\text{T\text{}eX}\) version, and a similar structure is created for textual \(\text{T\text{}eX}\) formulæ.

It is a batch tool that will upgrade any existing metadata with (presentation) MathML for any formula written in \(\text{T\text{}eX}\), as long as the \(\text{T\text{}eX}\) functions are known to the compiler.

The tool is written in Java and can either be called directly from another Java program, or can be made available as a REST service.

EnhanceNLM\text{}eXwMML relies on \text{}eX2NLM for the actual conversion from \(\text{T\text{}eX}\) strings to MathML.

This tool is free software governed by the CeCILL-C license that can be found at \url{http://www.cecill.info/}.

The assumptions for this tool are the following:

- It is applied to an XML Java DOM document (it must conform to EuDML specification v1.0, but could also be just a fragment thereof).
- This XML document can contain formulæ in \(\text{T\text{}eX}\) both
  - within the text data of various elements
  - or formatted with the NLM JATS basic structure:
    - a mandatory `<inline-formula>` (or `<disp-formula>`) for displayed math;
    - an optional `<alternatives>` element if the formula already has multiple versions;
    - a mandatory `<tex-math>` element holding valid \(\text{T\text{}eX}\) code (with \& and < escaped using &amp; and &lt; entities);
- Requirements for the \text{}eX2NLM tool apply for the character strings that contains \(\text{T\text{}eX}\) formulæ, whether in a `<tex-math>` element or not.

The result is another similar Java DOM document, where every `<*-formula>` element that had a \(\text{T\text{}eX}\) version now has an `<alternatives>` child, containing both \(\text{T\text{}eX}\) and MathML versions (and other alternatives that were previously present), and every \(\text{T\text{}eX}\) formula found within the text data of other elements is replaced by the appropriate `<*-formula>` element containing both the original \(\text{T\text{}eX}\) and it’s MathML equivalent. If no such formulæ are present in the document, the result is identical to the source.

Example 1: An input formula with \(\text{T\text{}eX}\)-only encoding in NLM structure

```xml
<inline-formula id="d18e3147">
  <tex-math>$J_k(n) := n^k \prod _{p \mid n}(1 - p^{-k})$</tex-math>
</inline-formula>
```

Example 2: An input formula with \(\text{T\text{}eX}\) encoding and alternative image in NLM structure

```xml
<inline-formula id="d18e3148">
  <alternatives>
    <tex-math>$J_k(n) := n^k \prod _{p \mid n}(1 - p^{-k})$</tex-math>
    <graphic xmlns:xlink="http://www.w3.org/1999/xlink" xlink:href="fors2682.f3" object-id="463492"/>
  </alternatives>
</inline-formula>
```
The tool’s output for the first example

\[ J_k(n) := n^k \prod_{p \mid n} (1 - p^{-k}) \]

This tool has allowed us to generate 66,808 NLM formulae from 10,363 article records in EuDML collections. It will now be run extensively on all EuDML content.

An interactive version of the tool is available for demonstration at [http://thar.ujf-grenoble.fr/EuDML/demo/NLMTeX2TeX+MML/](http://thar.ujf-grenoble.fr/EuDML/demo/NLMTeX2TeX+MML/)
2.8 Plain Text Reference Segmenter

The metadata for some documents does not contain bibliographic references, yet such information contributes to improved navigation in user interface, easy linking and, eventually, to further bibliometric analysis.

The Plain Text Reference Segmenter extracts bibliographic references from plain text. From the system integration point of view, it is implemented as a processing node that takes, as input, the plain text of a document (from a cache or from the storage) and an NLM metadata record. The segmenter enhances the NLM record with bibliographic references by adding `<mixed-citation/>` tags and outputs the enhanced NLM record. The tool is written in Java and does not depend on any third-party libraries.

This implementation of bibliographic extraction is based on the observation that certain characters, such as: comma, dot, colon, or parentheses occur frequently in the bibliography, so an abundance of such characters in a line is a hint that the line may be in the references section of an article. Therefore, the algorithm begins with “marking” the lines that might reside in the bibliographic references section of the document. Next, the region of the text with the largest concentration of such lines is assumed to be the references section. Finally, the section is split into individual references. To meet this end, several patterns are tested and line lengths are examined in order to deduce where one reference ends and another one begins. The current implementation is a proof-of-concept prototype that needs additional tuning and evaluation.

The tool is demonstrated at [http://wysoka.icm.edu.pl:18190/EuDmlAnalysisDemo/](http://wysoka.icm.edu.pl:18190/EuDmlAnalysisDemo/).

2.9 Bibliographic Reference Parser

Metadata sources often provide bibliographic references in the form of raw, untagged text. In order to navigate the references in a user interface or to analyze the citation network, it is necessary to parse the raw texts of references into fragments such as: author, title, journal, volume, year, etc. For example, the following input text:


should be parsed as follows (here in the RIS exchange format):

```
TY - JOUR
AU - Višňovský, Š.
VL - 36
SP - 625
PY - 1986
```

However, reference parsing is not a trivial task, for several reasons:

- There are dozens of established reference formats, and a great variety of formats “invented” by authors.
- Reference texts are “noisy” due to: misspellings, OCR errors and imperfect transformations from one format to another (the latter especially affects characters with diacritical marks and mathematical formulas).
- Interpretation of a reference is sensitive to punctuation: a single comma changed to a colon may alter the meaning of a whole citation.
In EuDML, the current implementation of reference parser is based on regular expressions (a future version may be based on Conditional Random Fields). It is implemented in Java, as a processing node which takes, as input, an NLM metadata record and returns a similar NLM record in which all the unparsed references are now parsed. The tool does not use any third-party libraries.

A joint demonstration of text extraction from PDF, bibliographic reference extraction from plain text, and bibliographic reference parsing is available at [http://wysoka.icm.edu.pl:18190/EuDmlAnalysisDemo/](http://wysoka.icm.edu.pl:18190/EuDmlAnalysisDemo/).

2.10 ZBMath Metadata Lookup Service

The ZBMath Lookup Service is a special online interface to the Zentralblatt MATH (ZBMath) database [10] which, given a Zentralblatt MATH item identification number (ZblId, an ZBMath 'AN' field value), returns the metadata that are stored in the ZBMath database for this publication. Metadata are provided in an XML formatted answer of the corresponding ZBMath record (content-type 'text/xml', encoding 'utf8', mathematical formulas are encoded in presentation MathML generated by Tralics. The DTD of the ZBMath Lookup XML answer is quoted below.

Access to this interface is free for EuDML project partners. EuDML project partners may use metadata from ZBMath to enhance or refine metadata for publications provided by them.

The tool provides a HTTP query interface that takes as input the ZblId of the item in question (URL encoded, see example below). The interface returns XML-encoded ZBMath metadata for that item, MathML encoded where possible, and Tex otherwise. The question of whether the data exposed should be converted to the EuDML metadata format based on NLM JATS DTD [16], or better delivered in a structure that closely reflects the ZBMath internal data format has been discussed, and it was resolved that during the current stage of the project the latter option is more suitable. A processing node that fetches a Zentralblatt record whenever a ZblId is encountered in the EuDML metadata was written by ICM, as part of the EuDML workflow. ICM has also written a processing node that calls a Zbl-to-EuDML metadata converter (to be implemented by IST) and stores the resulting EuDML metadata for further processing.

The tool is written in Python as part of the Zentralblatt MATH proprietary search engine. Access is possible via a HTTP interface. For example, [http://www.zentralblatt-math.org/eudml/lookup?zblid=1163.57016](http://www.zentralblatt-math.org/eudml/lookup?zblid=1163.57016) returns the following piece:

```xml
<?xml version="1.0" encoding="utf-8" ?>
<ZBMath-result schema-version="1.2"><item id="1163.57016">
  <title>
    <encoding type="tex"><![CDATA[$L^2$-invariants of finite aspherical CW-complexes.]]></encoding>
    <encoding type="tralics-v.2.14.4"><![CDATA[<math xmlns='http://www.w3.org/1998/Math/MathML'><msup><mi>L</mi><mn>2</mn></msup>-invariants of finite aspherical CW-complexes.]]></math>]]></encoding>
  </title>
  <authors>
    <author><name><encoding type="tralics-v.2.14.4"><![CDATA[Wegner, Christian]]></encoding></name></author>
  </authors>
</item></ZBMath-result>
```
The author calculates the $L^2$-invariants of finite aspherical CW-complex whose fundamental group $\Gamma$ has a finite subnormal series which terminates in a non-trivial elementary amenable group.

$L^2$-invariants have been intensively studied in recent years by many mathematicians, and have quite a few applications ranging from ring theory to differential geometry.

Among the more mysterious ones is the $L^2$-torsion. The main contribution of Wegner’s paper concerns the conjecture that this invariant is zero for finite classifying spaces of amenable groups. He establishes the conjecture when $\Gamma$ has a finite subnormal series which starts with an infinite elementary amenable group. To do this, he establishes a very nice technique of localization which preserves determinants. More generally, the result holds when $\Gamma$ is not aspherical, but has semi-integral determinant (introduced in [T. Schick, Trans. Am. Math. Soc. 353, No. 8, 3247–3265 (2001; Zbl 0979.55004)]).

The paper also establishes that under the same conditions the Novikov-Shubin invariants of the group are positive.

The author calculates the $L^2$-invariants of finite aspherical CW-complex whose fundamental group $\Gamma$ has a finite subnormal series which terminates in a non-trivial elementary amenable group.

$L^2$-invariants have been intensively studied in recent years by many mathematicians, and have quite a few applications ranging from ring theory to differential geometry.

Among the more mysterious ones is the $L^2$-torsion. The main contribution of Wegner’s paper concerns the conjecture that this invariant is zero for finite classifying spaces of amenable groups. He establishes the conjecture when $\Gamma$ has a finite subnormal series which starts with an infinite elementary amenable group. To do this, he establishes a very nice technique of localization which preserves determinants. More generally, the result holds when $\Gamma$ is not aspherical, but has semi-integral determinant (introduced in [T. Schick, Trans. Am. Math. Soc. 353, No. 8, 3247–3265 (2001; Zbl 0979.55004)]).

The paper also establishes that under the same conditions the Novikov-Shubin invariants of the group are positive.
DTD for the service output is here:

<!DOCTYPE ZBMath-result [ 
<!ELEMENT ZBMath-result (item | error)> 
<!ATTLIST ZBMath-result schema-version CDATA #FIXED "1.2"> 
<!ELEMENT item (title, authors, abstract*, source, classification?, keywords*, links*, language)> 
<!ATTLIST item id CDATA #REQUIRED > 
<!ELEMENT title (encoding)+> 
<!ELEMENT encoding (#PCDATA)> 
<!ATTLIST encoding type (tex|tralics-v.2.14.4) "tex"> 
<!ELEMENT authors (author+)> 
<!ELEMENT author (name)> 
<!ELEMENT name (encoding)+ > <!-- Author full name --> 
<!ELEMENT abstract (encoding)+ > <!-- Abstract of the ZBL Item --> 
<!ATTLIST abstract lang CDATA "en"> 
<!ELEMENT source (citation,serial|book)?,publisher*,year)> <!-- Bibliographic Source --> 
<!ELEMENT citation (#PCDATA)> 
<!ATTLIST citation encoding CDATA #FIXED "tralics-v.2.14.4"> 
<!ELEMENT serial (title,short-title,volume*,issue*,pages,issn*)> <!-- Bibliographic Source: serial type-- > 
<!ELEMENT short-title (#PCDATA)> 
<!ELEMENT volume (#PCDATA)> 
<!ELEMENT issue (#PCDATA)> 
<!ELEMENT pages (#PCDATA)> 
<!ELEMENT issn (#PCDATA)> ]>

The Zentralblatt identifier in the URL can be replaced by any valid Zentralblatt identifier.

### 2.11 ZBMath Reference Matching Service

This reference matching online **ZBMath Match Service** (’Match’) receives a string and searches for a matching record in the ZBMath database. If it finds a best-matching record it returns it as an XML formatted answer. The XML answer is validated by the same DTD as the ZBMath Lookup’s answer. If it does not find such a record the answer will be an XML formatted error:

```xml
<?xml version="1.0" encoding="utf-8" ?>
<ZBMath-result schema-version="1.2">
<error>ZBMath found no item with ID="-1"</error>
</ZBMath-result>
```

Internally, after this service identified the best-matching record, it asks the **ZBMath Lookup Service** with a **ZblId** to provide the corresponding XML formatted record. Then the Match service returns a copy of this answer to the query sender.

Query looks like this: [http://www.zentralblatt-math.org/eudml/match?query=B24%20,%20Inhomogeneous%20fixed%20point%20ensembles%20%20%201811](http://www.zentralblatt-math.org/eudml/match?query=B24%20,%20Inhomogeneous%20fixed%20point%20ensembles%20%20%201811). If the query string contains semicolons, they have to be URL-encoded, otherwise URL-encoding is optional. XML formatted answer to the above query is here:

```xml
<?xml version="1.0" encoding="utf-8" ?>
<ZBMath-result schema-version="1.2">
<item id="1195.82046">
<title>
<encoding type="tex"><![CDATA[Inhomogeneous fixed point ensembles revisited.]]></encoding>
</title>
</item>
</ZBMath-result>
```
Summary: The density of states of disordered systems in the Wigner-Dyson classes approaches some finite non-zero value at the mobility edge, whereas the density of states in systems of the chiral and Bogolubov-de Gennes classes shows a divergent or vanishing behavior in the band centre. Such types of behavior were classified as homogeneous and inhomogeneous fixed point ensembles within a real-space renormalization group approach. For the latter ensembles, the scaling law $\mu = d\nu -1$ was derived for the power laws of the density of states $\rho \propto |E|^\mu$ and of the localization length $\xi \propto |E|^{-\nu}$. This prediction from 1976 is checked against explicit results obtained meanwhile.
The Match service will not find an appropriate ZblId if there is insufficient information in the query, as its goal is to find exactly one result.

The tool is demonstrated at http://www.zentralblatt-math.org/eudml/match?query=B24, Inhomogeneous fixed point ensembles, 1811. The reference string in the URL can be replaced by other reference strings.

2.12 PdfJbIm

PdfJbIm is a PDF enhancer written in Java which reduces the size of PDF documents containing bitonal images [14]. It takes advantage of the extremely high compression ratio of visually lossless JBIG2 compression [9].

PdfJbIm uses the external open-source encoder jbig2enc [19], developed by Adam Langley. The compression ratio of jbig2enc has been improved by about ten percent by creating an additional comparison process for distinguishing representatives of symbols. For more information about PdfJbIm and the modification of jbig2enc see [13, 25].

Jbig2enc is being improved by integrating it with an OCR engine. OCR results are used to improve the compression ratio further and possibly even to improve the quality of the optimized PDF document. A Jbig2enc version with integrated OCR engine is now under development and shall be available for the next release in Deliverable D7.4.

It has been tested on journals stored at DML-CZ where the size of PDF documents originally compressed using Fax G4 was reduced on average by thirty percent. With the usage of Pdfsizeopt as the next optimization step, the size of PDF documents was reduced even further. For more details see Deliverable D7.2 and [25].

PdfJbIm uses library iText [8], which is provided under the AGPL license, which prevents direct integration in EuDML. Therefore it shall be used as separate tool or service. Thus PdfJbIm shall be used either as a service for the EuDML Refinement subsystem or as an external service for content providers. For further details see D7.2 [26].

It is currently integrated to EuDML as a processing node which uses shell script for engaging jbig2enc. Shell script sets appropriate parameters and shared libraries needed by jbig2enc. It’s used to prevent necessity of installing all necessary libraries system-wide.


2.13 Pdfsizeopt

Pdfsizeopt [23] is not a tool in the classic meaning of the word. It is a collection of best practices and Unix scripts to optimize the size of PDF documents. It is being developed in Python by Peter Szabó with support of Google.

While using Pdfsizeopt, a problem was encountered with removing trailers, xrefs and some other data (which has been allowed in PDF documents since PDF 1.5). Certain
tools have difficulties processing this kind of size optimized document—mainly tools based on the iText library [8]). This behaviour is possible to disable in Pdfsizeopt but the resulting size after optimization is not so greatly reduced.


2.14 math_metadata_lookup

The math_metadata_lookup program [18] is a small open-source (GNU LGPL [12]) command-line utility implemented in Ruby. It searches through mathematical reviews databases and fetches metadata. Supported reviews databases are Mathematical Reviews [2] (MR licensed access is necessary) and Zentralblatt MATH [10] (full metadata set is available only if Zbl licensed access is used, otherwise results are suboptimal) but it should be easy to add another database when necessary. The current version of the tool is in addition able to search through some other repositories, specifically DML-CZ, CEDRAM, NUMDAM and BulDML.

The math_metadata_lookup utility provides their users with four different search options:

- **article search** Search and fetch article metadata from reviews databases according to title, authors, year, or ID parameters given by the user.
- **author search** Search reviews databases and fetch all the different name forms for given author name. The database author ID and preferred name form is also given.
- **heuristic search** This is similar to the article search but only one (the best) match from each database is returned. Moreover, similarity of given parameters (title, authors, year) and found records must be higher than a threshold given by the user. Parameters given by the user are not used directly but are preprocessed to increase the probability of a search hit. Similarity of found records and original user given search parameters is computed using a generalized Levenshtein edit distance [20].
- **reference search** In this mode, the math_metadata_lookup utility tries to parse a user given reference string and identify title, author and year fields. These parameters are then used for a heuristic search. A user specified threshold is used to filter out records with insufficient similarity to the original search parameters.

Records found by the math_metadata_lookup utility are printed to the standard output in the desired format: plain text, HTML, XML, JSON, Ruby, or YAML.


2.15 Metadata Editor with Export to NLM

The Metadata Editor, ME, [1, 7] is an open-source (GNU GPL [11]) client–server web application implemented mainly in Ruby and Perl programming languages. It is designed to manage, edit and validate metadata and full texts of digital publications prior to their integration into a digital library.

Description of the tool can be found in Section 3.13 of [26]. However, recent versions of ME were enhanced with the ability to export its internal XML-based metadata
format in EuDML-compatible NLM files. Thus, ME can be directly used for EuDML metadata preparation.

The conversion process is implemented as a collection of XSL transformations of ME internal XML metadata files. The code was based on prototype transformation tools of metadata to EuDML NLM format implemented by Nicolas Houillon and Claude Goutorbe (Cellule MathDoc—Université Joseph Fourier). However, the functionalities of both tools were merged, the auxiliary metadata parser and processing tools implemented in the Java language were removed and their functionality was reimplemented in ME integrated processing scripts. The core of the NLM conversion process—a set of XSLT stylesheets—was expanded in functionality to cover all publication types and metadata elements used by ME. Several bugs were also fixed.

NLM metadata files are automatically generated by ME in the case of publishing of a prepared item (a journal article, for example) to the public digital library interface together with other data files (full-text in PDF format, for example) which can be made available in various ways (OAI-PMH, for example). Manual and collective generation of NLM metadata is also possible.

3 Integration of Eutools into the EuDML Enhancement Workflow

After the individual testing of tools at the technology provider sites, most tools mentioned in the previous section were integrated at EuDML central site at ICM in Warsaw, with the exception of PdfToText+MathViaOCR, math_metadata_lookup and ME, which provide enhancements at the providers’ sites.

The WP7 enhancement tools are used in two places in the system. One workflow takes place during the ingestion phase as services invoked by REPOX — called the ingestion workflow. The second core enhancement workflow takes place iteratively over validated ingested metadata and PDFs for every EuDML publication.

The ingestion workflow takes part in the REPOX 2.0 part of EuDML. The result of this workflow is validated metadata in EuDML NLM format. At the time of writing this proposal full details of this workflow has not yet been finalised. It currently contains EnhanceNLMTcXwMML and PdfToText+MathViaOCR, but will be enhanced as drawbacks in data from providers are incorporated.

The enhancement workflow is shown on Figure 2, showing the main procedural steps taken on the latest versions of the data ingested, taking into account time stamps from the ingestion process and previous enhancement loops.

1. Starting the enhancement process, ZBMath lookup is carried out. In the case that the Zentrallblat identifier is known, metadata from Zbl are fetched and added to those already collected from elsewhere.
2. Deduplication is performed, which identifies all duplicates not yet detected at Id assignment level.
3. Merging: records with multiple source records (including ZB math) are combined into single EuDML NLM records.
At this point records are in EuDML NLM format and are ready for enhancement. For some documents there is PDF with fulltext for the plain text content already, but not fulltext with mathematical formulae for indexing.

4. Next pdfTester is used to determine which type of PDF full text is supplied with the record. We have three possible PDF types:
   - retro-digitised — handled by PdfToTextViaOCR;
   - retro-born digital — handled by PDF Text Extractor and/or PDF2MML;
   - born digital — handled by PDF2MML.

Note: This step will be omitted if plain text in \texttt{\LaTeX} format is present (as in this case there is nothing better that could be obtained from our tools) or if we have plain text in any format for retro-digitised and retro-born digital files (here also we probably won’t get any new information).

5. When we have born digital PDF and only plain text as *.txt file it makes sense to treat this file with PDF2MML to obtain math formulae.

6. Then reference extraction will be performed if references are missing in the NLM version.

7. Finally all interesting fields (abstract, title, plain text, references, etc.) will be processed by tool that converts \texttt{\LaTeX} formulas to MathML format.

The result of this workflow is that fully enhanced metadata is present.
3.1 Workflow Evaluation

We have run a (partial) WP7 workflow on data from data providers and evaluated the results and part of the workflow integration and testing. More than 60,000 PDFs were processed as part of PDF workflow.

<table>
<thead>
<tr>
<th>tools</th>
<th># of runs during testing</th>
<th>Average speed (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>pdfTester</td>
<td>20,401</td>
<td>1.617879</td>
</tr>
<tr>
<td>PdfToTextViaOCR</td>
<td>19,482</td>
<td>22.420955</td>
</tr>
<tr>
<td>pdf2mml</td>
<td>448</td>
<td>5.376556</td>
</tr>
</tbody>
</table>

It is clear from Table 2 that some work on increasing efficiency has to be considered, as targeting for hundreds of thousands fulltexts in the future would otherwise require very considerable computing power.

During integration of eutools and their testing we were faced with several problems (and more are probably yet to come):

OCR Granularity

PDFs we have mined for fulltext (and math) indexing are of very different origin and were created by many different methods and tools. Only after processing them did we arrive at the problems at hand. One example to mention is file http://www.emis.de/journals/EM/expmath/volumes/12/12.3/Lewiner_et_al.pdf which consists of no fonts, but 800,000 images. This resulted in calling Tesseract 800,000 times on those images to get the fulltext of a single paper, which is quite inefficient. A different workflow strategy and fulltext algorithm had been designed to protect against this kind of inefficiency. It was solved by heuristics: if there are more than six images on one page, instead of calling OCR program on each of them, the whole page is rendered in high resolution first (using PDBBbox library) and the whole page is then passed to the OCR machinery to get fulltext.

Matching

EuDML records not containing Zbl and MR IDs should be enhanced by them as the principle input for the de-duplication algorithm. Currently, there are two tools for this purpose — ZBMath Reference Matching Service (see 2.11) and math_metadata_lookup (see 2.14).

The current version of the ZBMath Reference Matching Service available at http://www.zentralblatt-math.org/eudml/match?query= turned out have occasional long response times or service interruptions. However, these appear to be minor problems of the development version of the system and will be fixed in the near future.

ZBMath Reference Matching Service supposes use of unstructured ‘real-life’ reference strings. But in the EuDML system, there is the possibility that we have articles described in NLM format but no ‘real-life’ citation of it. In order to use the ZBMath Reference Matching Service, we have to create the reference string in some way.

The ZBMath Reference Matching Service algorithm considers the following fields from an NLM record the most relevant/helpful:
The algorithm considers two fields rather as sources of ambiguity: a) ISSN/ISBN; b) publisher. Thus, the EuDML system should prepare reference strings for look up with this in mind.

The math_metadata_lookup differs and is able to directly use known elements for search. In the ‘heuristic’ mode just one article is returned — according to adjustable threshold parameter the best matching. But ‘article’ mode can be used too: in this mode we can obtain more than one result for each repository. This could be useful as proper further processing of found results could finally find correct article IDs and even correct EuDML metadata.

math_metadata_lookup tool can also be used in ‘reference’ mode. This uses the same reference string as the ZBMath Reference Matching Service. Compiling results found from all the repositories math_metadata_lookup search through it provides persistent identifiers of articles in the other repositories (such as Mathematical Reviews IDs) than Zentralblatt MATH and possibly finds more Zbl IDs than the ZBMath Reference Matching Service.

### 4 Summary, Conclusions

A summary of the eutools prepared is shown in Tables 3 and 4. Most enhancers are to be used internally in the main EuDML architecture, with the exception of ME, which is being offered to potential partners (data providers) to edit the metadata to comply with EuDML data specification.

The tools were tested by the partner providing the tool, unit tested, and with the exception of ME and math_metadata_lookup they were integrated into the EuDML toolset. Eutools now will be merged into larger components and workflows and run on the central EuDML site on real data from providers.

Enhanced PDFs might be offered to partners via agreed interfaces. Otherwise they will be used for EuDML internal purposes, mainly by WP7–WP10 toolsets.

#### 4.1 Roadmap to Working Data Enhancement System

Analyzing subsystem components in Tables 3 and 4 it seems inevitable, because of reasons of efficiency, overlapping functionality, etc., that current eutools will have to be integrated into bigger programs—one per subsystem.

As a next step, eutools will be merged into bigger components (one enhancer per subsystem) and tested on the central EuDML site on real data from providers. The schedule to realize this will depend on the data collected and integration strategies developed by ICM. As a dozen programming languages, third-party tools and libraries must be collected and integrated, the scenario for integration is rather complicated—a set of rules for it has been written at https://wiki.eudml.eu/eudml/EuDML_System_Documentation for internal communication of developers.
Table 3: WP7 Eutools integration summary – OCR, Extraction, Conversion, and Analysis

<table>
<thead>
<tr>
<th>Tool name</th>
<th>Input</th>
<th>Output</th>
<th>Main benefit for EuDML</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OCR subsystem</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>PdfToTextViaOCR</strong></td>
<td>PDF with images as input stream or file (content/&lt;file name&gt;)</td>
<td>Set of plaintext/text files</td>
<td>OCR-ed text suitable for indexing</td>
</tr>
<tr>
<td><strong>PdfToText+MathViaOCR</strong></td>
<td>PDF with images as input stream or file (content/&lt;file name&gt;)</td>
<td>Set of XHTML files with mathematics in MathML</td>
<td>OCR-ed text with MathML suitable for indexing</td>
</tr>
<tr>
<td><strong>Extraction subsystem</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>PDFTester</strong></td>
<td>Any PDF file</td>
<td>Integer recording whether the file (a) has an additional layer for extraction, (b) can be used with the MathML extractor or (c) should be processed via OCR</td>
<td>Indicates most suitable tool for PDF analysis</td>
</tr>
<tr>
<td><strong>PDFBox</strong></td>
<td>Article PDF</td>
<td>Article fulltext</td>
<td>Extracts text from born-digital PDF (without using OCR)</td>
</tr>
<tr>
<td><strong>PDF2MML</strong></td>
<td>Suitable PDF, as indicated by PDFTester</td>
<td>List of MathML fragments for each formula within the file</td>
<td>Indexable, accessible MathML from a standard PDF</td>
</tr>
<tr>
<td><strong>Conversion</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>TeX2NLM</strong></td>
<td>UTF-8 encoded character string</td>
<td>Java DOM Element with (\text{T!!!!e}X) formulas converted to NLM structure with (\text{T!!!!e}X) and MathML alternatives</td>
<td>Upgrades untagged (\text{T!!!!e}X) formulas in a character string to full (\text{T!!!!e}X)+MathML NLM structure. Enables MathML based knowledge management and information retrieval</td>
</tr>
<tr>
<td><strong>EnhanceNLM\text{T!!!!e}X\text{w}\text{MML}</strong></td>
<td>Java DOM Document</td>
<td>Java DOM Document with all recognized (\text{T!!!!e}X) formulas represented by the NLM formula structure, containing both (\text{T!!!!e}X) and MathML version.</td>
<td>Enhances (\text{T!!!!e}X) formulas in NLM documents, either already in a NLM formula structure or not, to all be in such a structure and have a MathML alternative.</td>
</tr>
<tr>
<td><strong>Analysis subsystem</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Plain Text Reference Segmenter</strong></td>
<td>article plain text part with references</td>
<td>segmented set of references</td>
<td>Provides bibliographic references from plain text</td>
</tr>
<tr>
<td><strong>Bibliographic Reference Parser</strong></td>
<td>CDATA string of a reference</td>
<td>parsed reference in NLM XML</td>
<td>Identifies author names, title, publication year in a reference string</td>
</tr>
<tr>
<td>External/standalone tools</td>
<td>Description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------------------</td>
<td>-------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ME</td>
<td>Unorganized collection of scanned pages of documents</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Refinement subsystem**

<table>
<thead>
<tr>
<th>Service</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZBMath Lookup Service</td>
<td>Zbl identifier, Provides checked and rich article metadata given ZblId</td>
</tr>
<tr>
<td>ZBMath Match Service</td>
<td>bibliography reference string, Provides checked and rich article metadata given reference string</td>
</tr>
<tr>
<td>PdfJbIm</td>
<td>PDF as file or input stream, Re-compressed PDF as file or in output stream, Reduces size of PDF files containing images by about 30%</td>
</tr>
<tr>
<td>Pdfsizeopt</td>
<td>PDF as file (content/ &lt;file name&gt;), Optimized PDF as file, Reduction of PDF size by further thirty percent if used after running PdfJbIm</td>
</tr>
<tr>
<td>math_metadata_lookup</td>
<td>Search options (title/author/year plain text strings), Metadata found in mathematical reviews databases in desired format (plain text/ HTML/XML/JSON/Ruby/YAML), Stand-alone tool able to search through mathematical reviews databases and fetch metadata according to given search options</td>
</tr>
</tbody>
</table>

**Table 4: WP7 Eutools integration summary – Refinement and External**
4.2 Future Work

Before and after enhancement, duplicate document items should be detected and/or merged. A strategy for metadata conflict resolution remains to be developed, e.g. when a paper is available from different sources and the same item has different/conflicting metadata, usually compared to those from Zentralblatt.

A policy might be as follows: when a metadata field is absent or empty in the provider’s supplied metadata, and exists in the corresponding harvested Zbl record, then we should use the Zbl values. The paper might have keywords in various languages and lack English keywords: this is a typical added value/use case of enhancement expected from an enhancer, so the rule must take this kind of thing into consideration.

We could refine this policy on the basis of particular elements:

- title, source, author, main language, English title: use original if present
- MSC, English keywords: adding those provided by Zbl if they do not match the original ones could add value, even if the original ones exist. This would help to make more links/associations between our articles.

These and other issues have to be decided before EuDML workflows will deliver fully-fledged enhanced content in the EuDML system.

References


Index

Apache PDFBox library, 7

Bibliographic Reference Parser, 4, 24
BulDML, 19

CEDRAM, 19

EnhanceNLMTeXwMML, 4, 7, 9, 10, 20, 24

Eutool

  Bibliographic Reference Parser, 4, 24
  EnhanceNLMTeXwMML, 4, 7, 9, 10, 20, 24
  math_metadata_lookup, 4, 19, 20, 22, 23, 25
  ME, 4, 19, 20, 23, 25
  NLMTeX2TeX+MML, 7
  PDF Text Extractor, 4, 6, 7, 21
  PDF2MML, 4, 7, 21, 24
  PDFBox, 24
  PdfJbIm, ii, 4, 18, 25
  Pdfsizeopt, ii, 4, 6, 18, 19, 24
  PDFTester, ii, 4, 6, 7, 24
  PdfToText+MathViaOCR, ii, 4, 6, 20, 24
  PdfToTextViaOCR, ii, 4–6, 21, 24
  Plain Text Reference Segmenter, 4, 12, 24
  TeX2NLM, 4, 7, 10, 24
  ZBMath Lookup Service, 4, 13, 16, 25
  ZBMath Match Service, 4, 16, 25
  ZBMath Reference Matching Service, 22, 23

Infty, 6

Language

  Java, 8, 12, 13
  JSON, 25
  OCaml, 6, 7
  Perl, 19
  Python, 13, 18
  Ruby, 19, 25
  YAML, 25

math_metadata_lookup, 4, 19, 20, 22, 23, 25
ME, 4, 19, 20, 23, 25
MIaS, 8

NLMTeX2TeX+MML, 7
NUMDAM, 19

OAI-PMH, 20
OCR, 6
Optical Character Recognition, 6

PDBBox, 22
PDF Text Extractor, 4, 6, 7, 21
PDF2MML, 4, 7, 21, 24
PDFBox, 24
PdfJbIm, ii, 4, 18, 25
Pdfsizeopt, ii, 4, 6, 18, 19, 25
PDFTester, ii, 4, 6, 7, 24
PdfToText+MathViaOCR, ii, 4–6, 20, 24
PdfToTextViaOCR, ii, 4–6, 21, 24
Plain Text Reference Segmenter, 4, 12, 24
process, 5
processing node, 5
REPOX, 6

Subsystem

  Analysis, 3, 24
  Conversion, 3
  External, 3, 25
  Extraction, 3, 24
  OCR, 3, 24
  Refinement, 3, 25

Tesseract, 22
TeX2NLM, 4, 7, 10, 24
Tool

  jbig2enc, 18
  pdftk, 6, 7
  Tesseract, 5
Zbl-to-EuDML metadata converter, 13
ZblId, 4, 13, 16, 18, 25
ZBMath Lookup Service, 4, 13, 16, 25
ZBMath Match Service, 4, 16, 25
ZBMath Reference Matching Service, 22, 23
Zentralblatt MATH, 4