Introducing *Mr. DLib*, a Machine-readable Digital Library

Joeran Beel
Mr. DLib
Magdeburg, Germany
beel@mr-dlib.org

Bela Gipp*
UC Berkeley / OvGU
Mr. DLib / Berkeley
gipp@mr-dlib.org

Stefan Langer
Mr. DLib
Magdeburg, Germany
langer@mr-dlib.org

Marcel Genzmehr
Mr. DLib
Magdeburg, Germany
genzmehr@mr-dlib.org

Erik Wilde
UC Berkeley
School of Information
dret@berkeley.edu

Andreas Nürnberger
OvGU / FIN / DKE
Magdeburg, Germany
andreas.nuernberger@ovgu.de

Jim Pitman*
UC Berkeley
Dpt. of Statistics
pitman@stat.berkeley.edu

ABSTRACT

In this demonstration-paper we present Mr. DLib, a machine-readable digital library. Mr. DLib provides access to several millions of articles in full-text and their metadata in XML and JSON format via a RESTful Web Service. In addition, Mr. DLib provides related documents for given academic articles. The service is intended to serve researchers who need bibliographic data and full-text of scholarly literature for their analyses (e.g. impact and trend analysis); providers of academic services who need additional information to enhance their own services (e.g. literature recommendations); and providers who want to build their own services based on data from Mr. DLib.

Categories and Subject Descriptors

H.3.5 [**Online Information Services**]: Information Storage and Retrieval – Web-based Services.

General Terms

Management, Documentation

Keywords

Digital library, machine-readable, bibliographic data, impact analysis, citation analysis, trend analysis, api, web service, rest

1. INTRODUCTION

Researchers who need bulk access to citation data or the full text of academic articles currently have two options. Either they write parsers for digital libraries, such as *ACM Digital Library*, to collect the required data from the libraries' websites or they use data from some of the few academic services offering their data via an API (for instance, the social bookmarking service *Bibsonomy* offers an API to access their data in XML [1] and so does *Arxiv.org* and *DBLP*). However, existing services have some shortcomings. They either have very little data, offer it only in one format, maintain their API and documentation with low priority (because their focus lies on the web-interface), allow free access only to a limited set of data, or they have no full-text of articles in their database which is important for some analyses.

In this paper, we introduce *Mr. DLib*, a *Machine-readable Digital Library*. Mr. DLib's purpose is to provide machine-readable access, in XML and JSON format via an API, to full-text of academic articles and metadata including impact metrics and information about relatedness of documents. Mr. DLib is not intended to serve a digital library's "end-user" – a researcher looking for a single academic paper or information about it. Instead, Mr. DLib is intended to serve the following users:

1) Researchers requiring bibliographic data and full-text of scholarly literature for their analyses in bulk. For instance, Mr. DLib provides citation graphs and citation counts for scientific articles to perform impact and trend analysis and with full-text of academic articles, document similarities or plagiarism detection could be performed by the users.

2) Providers of academic services requiring additional bibliographic information for enhancing their services. For instance, the upcoming version of the reference manager *JabRef* uses Mr. DLib to obtain metadata of PDF files to make manual data entry superfluous. Also, the academic literature suite *Docear* uses Mr. DLib to provide literature recommendations to its users.

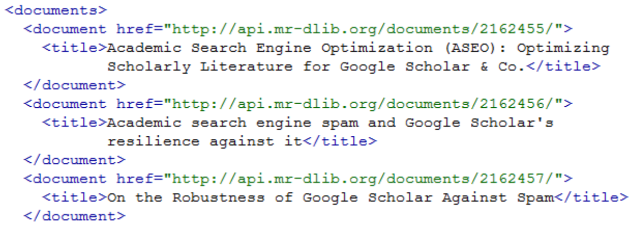
3) Providers who want to build complete services based on data from Mr. DLib. For instance, the *Probability Abstract Service* (PAS)[1] at the *University of California, Berkeley* has installed a web site informing users about the latest articles in the field of Probability. Every time a user visits the PAS website, data is delivered in real-time by Mr. DLib.

2. MR DLIB

2.1 Overview

Mr. DLib's dataset is built by a Web crawler, crawling the Web constantly for full-texts of academic articles, mostly PDFs. Currently, there are more than 2 million full-text articles in Mr. DLib`s database and bibliographic data of several million articles, mostly in the field of computer science.

To access data, Mr. DLib offers an API via a RESTful [2] Web Service. In contrast to a SOAP Web Service, REST is oriented closely on the HTTP standard. This makes using the API easy for web developers because they know the basic concepts already. In addition, REST focuses on resources. With regard to Mr. DLib, there are five types of resources: *documents*[2], *persons*[3], *conferences, journals,* and *organizations*[4]. Whenever possible Mr. DLib provides links between the resources, for instance between authors and articles and between articles and conference proceedings.



```
<documents>
  <document href="http://api.mr-dlib.org/documents/2162455/">
    <title>Academic Search Engine Optimization (ASEO): Optimizing
           Scholarly Literature for Google Scholar & Co.</title>
  </document>
  <document href="http://api.mr-dlib.org/documents/2162456/">
    <title>Academic search engine spam and Google Scholar's
           resilience against it</title>
  </document>
  <document href="http://api.mr-dlib.org/documents/2162457/">
    <title>On the Robustness of Google Scholar Against Spam</title>
  </document>
```

Figure 1: List of *documents* in XML format

To retrieve data, a simple HTTP `GET` command on a URI with an HTTP client (e.g. a Web browser) is sufficient. The URI structure follows the pattern http://api.mr-dlib.org/<resource_type>/<resource_id>/<resource_elements>/<parameters>.

* Work of this author was supported in part by the National Science Foundation Award 0835773 (Bibliographic Knowledge Network)

[1] http://pas.sciplore.org/

[2] Such as journal and conference articles, conference proceedings, books, and theses

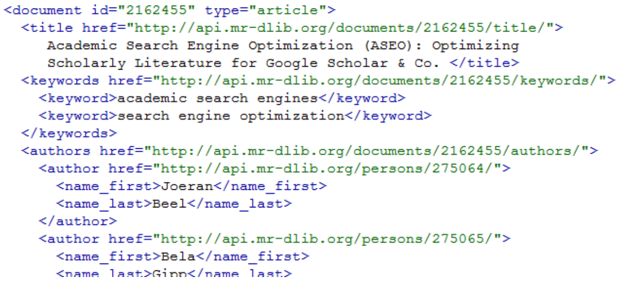
[3] Such as authors, editors, and conference chairs

[4] Such as universities and publishers

For instance, *http://api.mr-dlib.org/documents/* lists all documents in Mr. DLib`s database, including some basic metadata and a URI pointing to the specific document with more data (see Figure 1). Also status messages are adopted from the HTTP 1.1 standard. For instance, when a resource does not exist, a `404 NOT FOUND` error message is returned and when a request was successful, `200 OK` is returned.

2.2 Querying specific resources

Each resource has a unique ID on Mr. DLib and complete metadata can be retrieved by requesting *http://api.mr-dlib.org/<resource_type>/<resource_id>/*. For instance *http://api.mr-dlib.org/documents/2162455/* delivers metadata for the document with ID=2162455 as shown in Figure 2.



```
<document id="2162455" type="article">
  <title href="http://api.mr-dlib.org/documents/2162455/title/">
     Academic Search Engine Optimization (ASEO): Optimizing
     Scholarly Literature for Google Scholar & Co. </title>
  <keywords href="http://api.mr-dlib.org/documents/2162455/keywords/">
    <keyword>academic search engines</keyword>
    <keyword>search engine optimization</keyword>
  </keywords>
  <authors href="http://api.mr-dlib.org/documents/2162455/authors/">
    <author href="http://api.mr-dlib.org/persons/275064/">
      <name_first>Joeran</name_first>
      <name_last>Beel</name_last>
    </author>
    <author href="http://api.mr-dlib.org/persons/275065/">
      <name_first>Bela</name_first>
      <name_last>Gipp</name_last>
```

Figure 2: Metadata for a document (excerpt)

Depending on the resource, different data elements are returned. For instance, for a journal article, elements such as the *title*, *authors*, *keywords*, *publishing date*, *page numbers*, and *URIs to full-text*s are returned.

In addition to the unique ID, a resource can be identified in more ways. A document can be retrieved via specifying its title (e.g. *http://api.mr-dlib.org/documents/Academic_Search_Engine_Optimization_(ASEO):_Optimizing_Scholarly_Literature_for_Google_Scholar_&_Co./*), a third party ID (e.g. *http://api.mr-dlib.org/documents/#acm:187362/*), or a "clean-title"[5] (e.g. *http://api.mr-dlib.org/documents/academicsearchengineoptimizationaseooptimizingscholarlyliteratureforgooglescholarco/*). In addition, a PDF may be send to Mr. DLib via HTTP `POST` command and Mr. DLib then extracts relevant metadata from the PDF to identify it and returns available metadata.

2.3 Querying specific elements of a resource

As shown in Figure 2 each element (e.g. title) is retrievable via a separate URI, for instance http://api.mr-dlib.org/documents/2162455/title/ or *http://api.mr-dlib.org/documents/2162455/keywords/*. This reduces traffic when a user just wants to receive a specific element and not the complete set of metadata for a resource. It also allows retrieving impact metrics and related documents which are not available in the basic view to save computing power and traffic. For impact metrics, such as citation counts, h-index, impact factor, etc. http://api.mr-dlib.org/<resource_type>/<id>/impact/ has to be retrieved. Mr. DLib also offers related articles that can be retrieved via *http://api.mr-dlib.org/ documents/<id>/related/* (in future versions this might be available for other resources, e.g. authors, too).

2.4 Query parameters

Various parameters may be used to specify the request. Valid parameters are:

- '*format'* specifies the data format in which results are returned in. Currently, Mr. DLib supports XML and JSON.
- '*display*' specifies what elements shall be displayed for a resource. For instance, http://api.mr-dlib.org/documents/2162455/*?display=title,id,type,keywords* returns a result similar to Figure 2, but contains only the specified elements and no others.
- '*accuracy*' specifies whether only exact matches shall be returned (accuracy=*exact)* or also results with slight variations (accuracy=*fuzzy*).
- '*sort*' specifies the order results are returned in (alphabetically, relevance, date_created, date_lastmodified, date_insertedinDB).
- '*dlo*' (date low) and '*dhi'* (date high) specify a date range. For instance, http://api.mr-dlib.org/documents/?dlo=2011-01-01&*dhi=2011-02-28* returns documents published in January and February 2011.
- '*order'* specifies whether results are ordered 'ascending' or 'descending'.
- '*page*' specifies which page number of the result list shall be displayed and '*number*' the number of results per page.
- '*filter*' is a special parameter because the user can limit the original query to a subset of data. For instance, *http://api.mr-dlib.org/documents/?filter=arxiv* would display all documents in Mr. DLib`s database originally coming from Arxiv.org.

3. OUTLOOK

Mr. DLib is in Beta stage: it is functional and in use by some third parties (e.g. *Docear*). However, authentication methods, URIs, filters, query parameters, and other details may change in the final version. In addition, effective measures against spam need to be taken [3], performance must be improved, duplicate identification of papers and author name disambiguation is a challenge, and rights management respectively licensing is not yet fully clarified.

Also, a comprehensive search function is planned to allow searching for papers, authors, etc. by specifying keywords, and users shall be enabled to add and modify data. In the long run, faceted search, feeds, canonical URIs, and more standardized query parameters are planned to be implemented.

4. REFERENCES

[1] D Benz, A Hotho, R Jäschke, B Krause, F Mitzlaff, C Schmitz, and G Stumme. The social bookmark and publication management system BibSonomy. *The VLDB Journal*, 19 (6): 849–875, December 2010.

[2] R.T. Fielding. *Architectural Styles and the Design of Network-based Software Architectures*. Phd thesis, University of California, Irvine, 2000.

[3] Joeran Beel and Bela Gipp. Academic search engine spam and Google Scholar's resilience against it. *Journal of Electronic Publishing*, 13 (3), December 2010.

[5] To eliminate errors through different spellings we remove all special characters and spaces from a title