© 2014 by the authors; licensee RonPub, Lübeck, Germany. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (http://creativecommons.org/licenses/by/3.0/).

Open Access

Open Journal of Web Technologies (OJWT)
Volume 1, Issue 2, 2014

http://www.ronpub.com/journals/ojwt
ISSN 2199-188X

Getting Indexed by Bibliographic Databases in the Area of Computer Science

Arne Kusserow, Sven Groppe

Institute of Information Systems (IFIS), University of Lübeck, Ratzeburger Allee 160, D-23562 Lübeck, Germany,
kusserow@informatik.uni-luebeck.de, groppe@ifis.uni-luebeck.de

Abstract

Every author and publisher is interested in adding their publications to the widely used bibliographic databases freely accessible in the world wide web: This ensures the visibility of their publications and hence of the published research. However, the inclusion requirements of publications in the bibliographic databases are heterogeneous even on the technical side. This survey paper aims in shedding light on the various data formats, protocols and technical requirements of getting indexed by widely used bibliographic databases in the area of computer science and provides hints for maximal database inclusion. Furthermore, we point out the possibilities to utilize the data of bibliographic databases, and describes some personal and institutional research repository systems with special regard to the support of inclusion in bibliographic databases.

Type of Paper and Keywords

Research review: *Bibliographic Database, Repository, Digital Library, Publishing*

1 Motivation

Scholarly bibliographic databases collect information about scientific publications. Scholarly bibliographic databases for computer scientists of databases include commercial (but free to use) search engines like Google Scholar [32] and Microsoft Academic Search [59]. Search engines of associations like the Association for Computing Machinery (ACM) [7] and the Institute of Electrical and Electronics Engineers (IEEE) [39] contain mainly (but not only) the publications of their workshops, conferences and journals. Some scholarly institutions in particular universities maintain also search engines like CiteSeerX [48, 18], DBLP Computer Science Bibliography [49], Bielefeld Academic Search Engine (BASE) [8], Arnetminer [74] and arXiv [4]. There are even some special kind of social networks with statistics and recommendation (e.g. Research Gate [68] and Mendeley [57]). Readers can browse and query these collections of bibliographic databases, which provide links to full text pdfs of articles and/or to articles' pages of the publishers. Hence bibliographic databases are the starting point for literature review done by scientists.

Scientists wish their work to be *visible* (at least) inside (but also outside) their scientific community. For this purpose they meet with each other on workshops and conferences in order to present their work, discuss their ideas and build personal networks. After the event (and also for journal articles) it is very important for a scientist that his or her articles can be found in bibliographic databases such that scientists, which have not participated in the event, notice and consider his or her contributions. This is the precondition that his or her articles are cited and that the scientist reaches higher impact factors: The impact factor of a scientist is often measured by the Hirsch index [36] or one of its variants [2]. The Hirsch index is directly based on the citation count of articles and according to [36] defined as: A scientist

has index h if h of his or her N_p papers have at least h citations each and the other $(N_p - h)$ papers have $\leq h$ citations each.

Not surprisingly the inclusion in bibliographic databases is one of the most important key factors for the success of young publishers. On the other side bibliographic databases aim to include as many articles as possible, but some focus on already successful publishers due to their limited capacities and resources.

Some institutional research repositories like PUB[1] at Bielefeld University also prepare metadata of their contained publications for bibliographic databases. Authors using these institutional research repositories are independent from the publisher for the inclusion in bibliographic databases. Once they have added their publication information to their institutional research repository, their publications appear in the supported bibliographic databases after the next crawling turn. This ensures a timely inclusion as well as a higher number of bibliographic databases aware of their publications. However, many bibliographic databases only collect data from publishers and not from institutional or even personal research repositories.

Related questions immediately arise:

- What are widely used and hence important scholarly bibliographic databases for a specific research area like computer science?
- Who can add publication information to these bibliographic databases?
- How can publication information be added to bibliographic databases?
- What data formats and protocols are used for im- and export?
- How to utilize and automatically process the information provided by bibliographic databases?

This review aims to provide the answers to these questions by analyzing

- the basic data formats and protocols for data exchange to and from bibliographic databases in Section 2,
- the most important scholarly bibliographic databases for computer scientists in Section 3,
- personal and institutional research repository systems in Section 4, and
- further related work in Section 5.

[1] http://pub.uni-bielefeld.de/en

2 Data Formats and Protocols

This section deals with several bibliographic data formats and protocols. These are used to exchange or embed the metadata of publications, to manage bibliographies or even for describing the formatting and styling of citations.

2.1 BibTeX

BibTeX was developed by Oren Patashnik and Leslie Lamport for managing and formatting lists of references [65]. It is typically used together with LaTeX documents but can be used with other software and became a de facto standard to define references. The first release was in 1985 and the latest version is from 2010.

A BibTeX file consists of a set of entries each of which represent a reference. There are currently 14 common entry types in BibTeX, for example `book`, `article` or `incollection`. Each of this entry type has required fields and some optional fields, where for example `title` and `author` are almost always required. Multiple authors should be separated with an `and` instead of commas. Further information can be found in the format description [29, 30].

Listing 1 presents an exemplary BibTeX entry and Figure 1 the generated output in the *References* section for this BibTeX entry. The generated output depends on the used citation style. The one shown here corresponds to the style ieeetrans.

```
@article {kumu14:sample-pub,
  author = {Arne Kusserow and
            Max Mustermann},
  title = {Researching Something Special},
  journal = {Special Research Journal},
  number = {1},
  volume = {4},
  pages = {15--21},
  year = {2014},
  month = {November}
}
```

Listing 1: BibTeX example

[1] A. Kusserow and M. Mustermann, "Researching something special," *Special Research Journal*, vol. 4, no. 1, pp. 15-21, November 2014.

Figure 1: Generated output in the *References* section for the BibTeX entry of Listing 1

Each entry must have a citation key, which is later used as a reference to this entry within the LaTeX doc-

ument. This key must be unique within a BibTeX file and is ideally easy recognizable. When the LaTeX document is compiled, BibTeX generates the list of references from the given BibTeX entries using the template for a selected citation style. A BibTeX file is a plain text file that can be edited with any text editor, but there is also software (for example BibDesk[2], JabRef[3] or Bibfilex[4]) that can manage whole bibliographies in one big file, offering a graphical user interface, search and sort features and auto-generation of the citation keys following a predefined or user-defined pattern as well as managing associated PDF files or URLs.

2.2 Dublin Core

The Dublin Core Metadata Initiative (DCMI) [24] has developed a common system for using metadata to describe web resources at a conference at Dublin/Ohio in the mid 1990s. This system is known as Dublin Core Metadata and was meant to allow the authors of websites to describe their content in a way it could be discovered by keyword-based search engines.

```
<dc:title>
  Researching Something Special
</dc:title>
<dc:type>Text</dc:type>
<dc:creator>Kusserow, A</dc:creator>
<dc:creator>Mustermann, M</dc:creator>
<dcterms:issued>
  2014-11-15
</dcterms:issued>
<dcterms:bibliographicCitation>
  Special Research Journal 4(1), 15-21.
  (2014)
</dcterms:bibliographicCitation>
```

Listing 2: Dublin Core example

Nowadays though most of the leading search engines ignore most of the metadata of websites because there was an extensive misuse and spamming with metadata in the past. Practical benefits of the use of metadata remain only in fields where the search engines can trust the data and thus consider it. That is why the Dublin Core Metadata, although originally conceived for all kinds of content, is now primarily used in the library world [72]. But even there Google claims that "they work poorly for journal papers because Dublin Core doesn't have unambiguous fields for journal title, volume, issue, and page numbers".[5] Instead, a complete bibliographic citation including for example the journal, volume, issue and page numbers can and is recommended to be given in the `bibliographicCitation` field [3].

Dublin Core Metadata can be presented in RDF/XML or embedded in the head section of HTML documents in `<meta>` tags. There are 15 core elements and some more additional elements in Dublin Core. All elements are optional, can be repeated and can be in any order. We refer the interested reader to the format description [23, 24] for more details. Listing 2 shows an exemplary Dublin Core entry.

2.3 Highwire Press Tags

Highwire Press Tags is a meta tag format developed by HighWire. To the best of our knowledge there is no official documentation or format description about these tags available online. However, the Google Scholar Inclusion Guidelines for Webmasters (see Section 3.1) [33] describe the usage of these tags. Highwire Press Tags can be embedded in the head section of HTML documents in `<meta>` tags.

```
<meta name="citation_title"
  content="Researching Something Special"/>
<meta name="citation_author"
  content="Kusserow, A."/>
<meta name="citation_author"
  content="Mustermann, M."/>
<meta name="citation_publication_date"
  content="2014"/>
<meta name="citation_journal_title"
  content="Special Research Journal"/>
<meta name="citation_volume"
  content="4"/>
<meta name="citation_issue"
  content="1"/>
<meta name="citation_firstpage"
  content="15"/>
<meta name="citation_lastpage"
  content="21"/>
<meta name="citation_pdf_url" content="
    http://www.example.com/full.pdf"/>
```

Listing 3: Highwire Press Tags example

Listing 3 presents an exemplary entry using Highwire Press Tags. Google Scholar requires the minimum of `citation_title`, one `citation_author` tag for every author and the `citation_publication_date` in order to properly index the content. A link to the PDF file

[2] http://bibdesk.sourceforge.net/
[3] http://jabref.sourceforge.net/
[4] https://sites.google.com/site/bibfilex/

[5] http://scholar.google.com/intl/en/scholar/inclusion.html#indexing

with the full text should ideally be supplied, too [33]. Presumably it is the best to include as many meta tags as reasonably possible.

2.4 Publishing Requirements for Industry Standard Metadata (PRISM)

PRISM is developed by the PRISM Metadata Initiative and was first released in 2001. The current version is 3.0 from 2012. It specifies an XML/RDF element set for storing and interchanging metadata that describes resources or their relationship to other resources and incorporates Dublin Core elements (see Section 2.2) [37]. Listing 4 contains an example of PRISM meta tags. Although initially developed with the needs of publishers in mind and focused on handling print content like magazines or journals, its newer version is not limited to bibliographic usage but can describe many types of content. PRISM can also define property rights or usage permissions. The official PRISM introduction document [38] contains further information.

```
<meta name="prism.title"
  content="Researching Something Special"/>
<meta name="prism.publicationYear"
  content="2014"/>
<meta name="prism.publicationDate"
  content="2014-11-15"/>
<meta name="prism.publicationName"
  content="Special Research Journal"/>
<meta name="prism.volume"
  content="4"/>
<meta name="prism.number"
  content="1"/>
<meta name="prism.startingPage"
  content="15"/>
<meta name="prism.endingPage"
  content="21"/>
<meta name="prism.url" content="http://www
    .example.com/full.pdf"/>
```

Listing 4: PRISM Tags example

2.5 Metadata Object Description Schema (MODS)

The Metadata Object Description Schema (MODS) was developed in 2002 by the Network Development and MARC Standards Office of The United States Library of Congress [83]. The current version is 3.5 from 2013. It is an XML format schema originally intended for use in library applications.

MODS is a derivative of the MARC21 format (see Section 2.12), designed to be less detailed, more simple and more human readable with less abbreviations or coded values [54]. MODS offers 19 top-level-elements and several sub-elements. All elements except `titleInfo` are optional and all but `recordInfo` are repeatable. According to [34], MODS is intended to offer an alternative or complement between too simple metadata formats and too complex, feature-rich ones.

2.6 Journal Article Tag Suite (JATS)

JATS is an XML format schema developed by the National Information Standards Organization (NISO) for exchanging the metadata and content of scholarly journals [62]. It is not intended to be used for books, magazines, reviews or any other type of publication.

JATS comes with three different tag sets for the purposes of archiving, publishing and authoring, each defining numerous XML elements and attributes. We refer the interested reader to the official document from NISO [62] for more information.

2.7 Open Archives Initiative Protocol for Metadata Harvesting

The Open Archives Initiative (OAI) developed an interoperable standard for gathering, exchanging and processing metadata of publications scattered across many services or repositories [47, 60]. This standard is referred to as the Open Archives Initiative Protocol for Metadata Harvesting (or abbreviated as OAI-PMH).

The protocol differentiates between Data Providers and Service Providers [47]. Data Providers provide the metadata and Service Providers use and process that data, for example for offering a search engine. It is possible to register as an official Data Provider [85] but there is no need to register in order to access data as a Service Provider. There is a list with all registered Data Providers available.[6]

The protocol is based on HTTP Requests and XML. The request for data is send via a HTTP `GET` or `POST` request to the repository's web server, which then processes this request and responds with an XML output. The metadata format of the response can be chosen from a list of formats that the current repository has implemented. All repositories must at least implement the Dublin Core format (see Section 2.2). Further information can be found in the official protocol description [47] and the appended implementation guidelines. There is also a list available with existing tools for implementing an OAI-PMH interface.[7]

[6] `http://www.openarchives.org/Register/BrowseSites`
[7] `http://www.openarchives.org/pmh/tools/tools.php`

Listing 5 presents an exemplary OAI-PMH request that requests the metadata (in Dublin Core format) of a record with the given identifier and Listing 6 shows the corresponding response.

```
http://oai.sample-repository.org?verb=
    GetRecord&metadataPrefix=oai_dc&
    identifier=oai.sample-repository.org:
    id5
```

Listing 5: OAI-PMH request example

```
<OAI-PMH xsi:schemaLocation="http://www.
    openarchives.org/OAI/2.0/ http://www.
    openarchives.org/OAI/2.0/OAI-PMH.xsd">
 <responseDate>2015-02-17</responseDate>
 <request verb="GetRecord" identifier="oai.
     sample-repository.org:id5"
     metadataPrefix="oai_dc">http://oai.
     sample-repository.org</request>
 <GetRecord>
  <record>
   <header>
    <identifier>oai.sample-repository.org:
        id5</identifier>
    <datestamp>2014-11-21</datestamp>
    <setSpec>gmd</setSpec>
   </header>
   <metadata>
    <oai_dc:dc xsi:schemaLocation="http://
        www.openarchives.org/OAI/2.0/oai_dc
        / http://www.openarchives.org/OAI
        /2.0/oai_dc.xsd">
     <dc:title>Researching Something
         Special</dc:title>
     ...
    </oai_dc:dc>
   </metadata>
  </record>
 </GetRecord>
</OAI-PMH>
```

Listing 6: OAI-PMH response example

2.8 DBLP XML/Submission Format

The DBLP XML format is a special XML format developed by the DBLP Computer Science Bibliography. It can be concisely described as *BibTeX written in XML*.

Like BibTeX, a DBLP XML file consists of one or multiple entries. Each entry has an entry type and some tags associated. The entry types are the same as specified in BibTeX but limited to `article`, `inproceedings`, `proceedings`, `book`, `incollection`, `phdthesis` and `masterthesis`. As an addition, there is the entry type `www` which is not present in BibTeX and is used for author homepages.

The tags are also the same but the only tag required is the title tag, all others are optional. Another difference is that in DBLP XML each author gets his own tag, sorted by the order of their appearance on the publication. Additionally there are tags for the last modification date, for the URL to the publisher's website and for the URL to the table of contents of the book or conference the entry is a part of. Finally, all entries have a unique key generated by a specific pattern, similar to the cite key known from BibTeX. Further information can be found in [49]. Listing 7 shows an exemplary DBLP XML entry.

```
<?xml version="1.0"?>
<article key="journals/sjans/kusserow"
    mdate="2014-11-19">
  <author>Arne Kusserow</author>
  <author>Max Mustermann</author>
  <title>
    Researching Something Special
  </title>
  <pages>15-21</pages>
  <year>2014</year>
  <volume>4</volume>
  <journal>
    Special Research Journal
  </journal>
  <number>1</number>
  <ee>http://doi.acm.org/...</ee>
  <url>http://www.example.com/full.pdf
  </url>
</article>
```

Listing 7: DBLP XML example

The DBLP Submission Format has high similarities with DBLP XML but is intended only for letting publishers submit metadata to the DBLP Computer Science Bibliography. Detailed information about the submission format can be found in [82].

2.9 Research Information Systems (RIS)

The Research Information Systems (RIS) format was developed by Thomson Reuters mainly for allowing users of their applications EndNote[8] and ReferenceManager[9] to import bibliographic metadata from websites or export data [88].

[8] http://endnote.com/
[9] http://www.refman.com/

```
TY - JOUR
AU - Kusserow, Arne
AU - Mustermann, Max
TI - Researching Something Special
AB - Abstract of this publication...
T2 - Special Research Journal
PY - 2014/11/15
IS - 1
VL - 4
SP - 15
EP - 21
UR - http://www.example.com/pub_id5
L1 - http://www.example.com/full.pdf
ER -
```

Listing 8: RIS example

A RIS file is a plain text file that contains one or multiple entries. Each entry consists of several fields, which are composed of a tag indicating the field type followed by two whitespaces, a single dash, another whitespace and finally the content of that field. Each field must be on a separate line. They can be in any order with the exception of the first field `TY` (the reference type, similar to BibTeX but with many more types) and the last field `ER` (end of reference). Authors' names must be in the format `Surname, Forename` and every author gets his own `AU` field. Authors can also be represented as primary, secondary, etc., using `A1`, `A2`, `A3`, `A4`. A full list of all supported tags, reference types and other format specifications can be found in the format description [88]. Listing 8 shows an exemplary RIS entry.

When a website offers export to the RIS format, sending the MIME type `application/x-research-info-systems` along with the exported data will automatically open up EndNote or ReferenceManager on the user's system and import it to the user's library [88].

2.10 Citation Style Language (CSL)

The Citation Style Language (CSL) is an open standard for describing the formatting and styling of citations in an XML-based format. It was initially created by Bruce D'Arcus and is maintained by many contributors [96]. It offers about 7000 free crowdsources citation styles and is used by many commercial or open source projects.

There are CSL processors (known as `citeproc`'s) available for several programming languages. They can create correctly formatted citations from given publication metadata using the CSL styles. `Citeproc-js`, a well-known CSL processor for JavaScript, introduced a JSON[10] schema called CSL-JSON to handle the bibliographic metadata. It is not an official standard but it's used by other CSL tools, too. The schema is described on the CSL project's GitHub page.[11] The project's website [96] describes more details of CSL. Listing 9 presents an exemplary CSL-JSON entry.

```
[{
    "author": [
        {
            "family": "Kusserow",
            "given": "Arne",
        },
        {
            "family": "Mustermann",
            "given": "Max",
        }
    ],
    "container-title": "Special
        Research Journal",
    "id": "5",
    "issued": {
        "date-parts": [
            ["2014", "11", "15"]
        ]
    },
    "page": "15-21",
    "title": "Researching Something
        Special",
    "type": "article-journal",
    "volume": "4"
}]
```

Listing 9: CSL-JSON example

2.11 Digital Object Identifier (DOI)

A digital object identifier (DOI) is used to identify an object unrelated to its current location and thus providing persistence when referencing to it [40]. Metadata and location can be stored together with the DOI but the identifier will remain fixed even when the data changes.

That is why a DOI provides a great benefit when linking to publications. When the location of the document has changed and the location data of the identifier was updated accordingly, the linking via the DOI will stay valid. If a DOI was not used, all links to the publication would become invalid and would need to get updated with the new location of the document.

The standardized DOI system is developed and controlled by the International DOI Foundation (IDF) [42]. Several Registration Agencies[12] provide the necessary

[10] `http://json.org/`

[11] `https://github.com/citation-style-language/schema`

[12] `http://www.doi.org/registration_agencies.html`

infrastructure for registering DOI names and maintaining the data. Each Registration Agency is allowed to offer its own business model, so the cost of registering new DOI's can vary. An example for prices can be seen in [22]. Resolving an existing DOI is free.

Exemplary DOI's would be `10.1.1.102.0815` or `10.3390/jsan2020172`. More information about the DOI Format can be found in [41].

2.12 Other Data Formats

The **EPrints** repository software (see Section 4.2) has its own meta tag format and the Berkeley Electronic Press has developed the meta tag format **BE Press Tags** for its repository software (see Section 4.3).

MARC21 is a meta data format designed for carrying bibliographic information [84]. Due to its numeric field names and coded values it is rather unintuitive and hard to read (see some examples[13]). More information about the format can be found in [84].

CSV is the abbrevation for *Comma-seperated values* which is not really a bibliographic data format, but a widely used plain text format that can store tabular data. It can have any number of records separated by line breaks. Each line shares an often identical series of fields which are separated by a character, commonly the comma.

YAML is a markup and data serialization language with high similarities to JSON (see CSL-JSON in Section 2.10) [11]. In fact JSON is a subset of YAML. The author of [31] suggests the usage of YAML together with CSL (see Section 2.10) because of the balance between human and machine readability.

2.13 Overview of Data Formats

After introducing these formats and protocols it is now important to find out how widely each of them is used. As a reference, the usage of these formats by the bibliographic databases introduced in Section 3 has been considered.

Table 1 gives an overview of the formats that the bibliographic databases offer their users to export the metadata of publications. The user can import this exported data in his or her favorite software to create and manage citation lists or bibliographies. In this case BibTeX is by far the most frequently offered format, which perfectly makes sense since it is the most widely accepted bibliographic metadata format in general. Following at a distance is RIS and then some other formats.

DOIs are also widely used among the examined databases. Almost all of them show and use the DOI to provide a permanent link to the full text document.

It is also interesting to take a look at the import and export formats used on the backend of the databases. This will be further analyzed in Section 3.12 and Section 3.13 after the databases have been introduced.

3 Scholarly Bibliographic Databases for Computer Scientists

This section deals with several established bibliographic databases and search engines for scholarly publications which are most important for computer scientists. These services provide information to the users about publications, authors, journals or conferences and much more. Some also offer detailed statistics and visualizations about the data. Each of these services has been examined on how to add data to their index or collect data from them.

3.1 Google Scholar

In 2004, Google released a search engine designed for scholarly publications called Google Scholar [32]. Beside the information on publications it provides author profiles and citation statistics. It also offers full-text-indexing of the publications.

Google Scholar uses a crawler to autonomously search the web for new data. The indexed data gets evaluated and ranked by computer algorithms. Google neither gives details about these algorithms nor any information on when this crawling happens or why it might have failed. But they have published detailed guidelines how the data should be presented to get added to the index successfully [33].

Google Scholar can index and extract information from PDF files and HTML pages [33]. Each page must contain the meta information of no more than one publication. They recommend to use HTML meta tags and support Highwire Press (see Section 2.3), Eprints, BE Press, PRISM and Dublin Core tags (see Section 2.2). The usage of Dublin Core tags is not recommended because they "work poorly for journal papers".[14] Google uses Highwire Press tags in all its given examples. In addition to that or if there are no meta tags present, Google Scholar can also consider the structure of the HTML page or PDF file and its elements and thus try to identify the metadata. Every page or file of a website must be reachable from within a maximum of ten links down from the starting page. That is why larger databases are recommended to provide listings of all entries grouped

[13] `http://www.loc.gov/marc/bibliographic/examples.html`

[14] `http://scholar.google.com/intl/en/scholar/inclusion.html#indexing`

Database	Front End Export Formats
Google Scholar	BibTeX, RIS, EndNote, CSV
CiteSeerX	BibTeX
BASE	BibTeX, RIS, RDF (Dublin Core), EndNote, JSON, MARC, YAML
MAS	BibTeX, RIS
Arnetminer	BibTeX
DBLP	BibTeX, RIS, DBLP XML
CCSB	BibTeX
arXiv	none
ResearchGate	none
Mendeley	BibTeX, RIS, EndNote XML

Table 1: Front end export formats offered by the databases

by for example the year of publication. Also, offering a list of recently added entries is highly recommended because this smaller list can be crawled more frequently.

There seems to be no way of getting or accessing data from Google Scholar except just linking to it. Google does not offer an API nor interface to access its data.

3.2 CiteSeerX

CiteSeerX is a scientific literature search engine and database with focus on computer sciences. Its former version, CiteSeer, was first developed in 1997 but couldn't stand the extended tasks any longer after about ten years. It was then redeveloped and became CiteSeerX [18]. CiteSeer was the first to provide autonomous citation indexing, what means that it automatically extracts and indexes citations from publications for better search results and for evaluational or statistical usage [48]. It also offers full-text-indexing of the publications.

CiteSeerX uses a focused crawler to get new data. This crawler doesn't search the whole web but only sites from a crawl list [16]. There is a form that allows to submit URL's for crawling, but it states it only searches for PDF files (and some other) to a depth of one [19]. There is no validation taking place to verify the content or the author of the found publications.

The data from CiteSeerX is freely available for access via an OAI-PMH interface. The full data can also be downloaded, but this requires getting in further contact with the team of CiteSeerX [17].

They also offer an extractor which is open source and can extract metadata and citations from documents and return it in several different formats, for example BibTeX or XML [15].

3.3 DBLP Computer Science Bibliography

The DBLP Computer Science Bibliography is a web service of the Schloss Dagstuhl - Leibniz-Zentrum für Informatik GmbH and the University of Trier [49]. In 2009 it contained more than 1.2 million indexed publications and about 700,000 authors. Today there are even more than 2.6 million publications listed [80].

A direct input of data into the DBLP Computer Science Bibliography is not possible for everyone [81]. The DBLP automatically indexes the table of contents of complete proceedings or journals provided by the major publishers. If there is no metadata provided, conference chairmen or journal editors can provide the metadata using the DBLP Submission format [82].

All indexed data of the DBLP is offered for download as a single XML file using the DBLP XML format (see Section 2.8) [49]. Additionally there is the possibility to access specific data via an API that returns data in the DBLP-XML format.

Using crawlers to get data from the DBLP website is currently allowed, but there is a notice that "this feature will change in the near future" [79]. It is recommended to use the offered API. Whether crawling or API calling, the scripts should be able to react correctly to timeouts and must not perform a more than reasonable amount of data requests per minute. If a lot of data is needed, one is advised to use the complete XML file instead [79].

3.4 The Collection of Computer Science Bibliographies

Founded in 1993, the Collection of Computer Science Bibliographies gets and merges data from now more than 1500 individual bibliographic databases. It covers most aspects of computer sciences and has currently more than 3 million references to scientific publications. Some major sources are for example CiteSeer, DBLP and arXiv. But there are also much smaller author-, subject- or

conference-specific databases that contribute. The data from these sources gets updated weekly [75, 78].

The Collection of Computer Science Bibliographies offers the possibility to add own data via a direct input of single BibTeX entries which will then take a week to appear on the site [76]. Or one could add a whole new source collection if certain criteria are fulfilled. In order to do this, the bibliography needs to have a single compressed plain text file with a fixed name containing all data as BibTeX entries, accessible via HTTP or FTP. This file will then be mirrored and automatically kept up-to-date. A submitted bibliography should cover either a specific subject area, a complete journal or conference or all publications of a specific institution or publisher [77].

There seems to be no way of getting or accessing data except just linking to it or manually downloading single BibTeX files. It is possible to become a mirror site that copies the whole collection, but this won't serve the purpose of accessing particular data.

3.5 Microsoft Academic Search

Microsoft Academic Research is an experimental search engine and research service by Microsoft Research [59]. It is not intended for production use and can be taken offline once its project goals are fulfilled. In addition to the indexing of publications it offers detailed statistics about the publications, citations and the relationship between subjects and authors. It also offers extensive author profile pages and many visualizations.

There is a full list of data providers available online.[15] It includes for example arXiv, BASE, CiteSeer and DBLP. If one has another content source that could be added, he or she is encouraged to contact Microsoft Research. Registered users can also submit a PDF file or a BibTeX file to add a single publication to an author profile [59].

Microsoft Academic Research offers an API to access its data, but in order to use this one has to be a registered user and request an AppID, wherefore the desired application must met certain criteria [59]. Anyway it is possible for everyone to embed the author profiles using JavaScript.

3.6 Bielefeld Academic Search Engine (BASE)

The Bielefeld Academic Search Engine (BASE) is a search engine for academic open access documents by the Bielefeld University Library [8]. It contains data about more than 60 million documents from more than 3000 indexed sources using the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH). BASE claims that it intellectually selects and evaluates these sources before indexing them.

BASE is a registered Open Archives Initiative Service Provider (see Section 2.7) [8]. It is possible to suggest a new data source for inclusion if this source has a correctly implemented OAI-PMH interface providing the data [10], but if the source is a registered OAI Data Provider it should already be included [9]. BASE offers an OAI-PMH Validity Checker to verify if the OIA-PMH interface is working correctly. Besides that, the intellectual criteria used to select the sources are not really specified. The data from the selected sources gets updated every weekend [9].

BASE also provides access to its own data via a HTTP interface or an OAI-PMH interface that returns data in the Dublin Core format or an extended Dublin Core format with extra elements by BASE [8]. In order to get access to these interfaces, one needs to register its IP-address.

3.7 Mendeley

Mendeley is a web service with associated desktop and mobile applications for managing, sharing and citing scientific publications. It offers author profiles, a kind-of social network, statistics and recommendations [57]. Mendeley is accessible for registered users only. The registration is free.

Mendeley gets its data from external sources, among others from Google Scholar and arXiv [56]. Also, webmasters can add an import button to their websites with publications. If a users clicks that button, Mendeley will import the publication to this user's collection. It recognizes many HTML meta tag formats and recommends to stick to the Google Scholar Inclusion Guidelines for Webmasters (see Section 3.1). In addition to that, registered users can directly submit single publications.

There is an API that offers developers the possibility to develop own applications extending or using Mendeley's features [58]. But this does not seem to serve the purpose of accessing or embedding particular data from Mendeley for use in an external service.

3.8 Arnetminer

Arnetminer is a search engine for scientific publications and authors which was developed in 2006 by Professor Jie Tang from the Tsinghua University, China [74]. It aims to create extensive author profiles by extracting information and integrating academic data from various sources and analyzing the created author's social network. It offers a large amount of statistics and visualizations about authors and had 3.2 million publications and 700,000 authors listed in the year 2010.

[15] http://academic.research.microsoft.com/About/Help.htm#5

According to the authors of [50], Arnetminer fetches data on researchers and authors by using a search engine (Google API) to find relevant websites and then extracts information from these. Additionally, it integrates the publication data from existing online sources, in particular from the DBLP Computer Science Bibliography (see Section 3.3). There seems to be no way to input data directly.

Arnetminer offers an API for searching or accessing data, even the author profiles can be accessed. When sending a request using a HTTP request, the server will answer with a formatted output of the data. It is possible to search for publications, conferences and experts on a subject by a keyword and for author profiles by name or id. Further information can be found in [35].

3.9 arXiv

In 1991, arXiv started as xxx.lanl.gov, a server for exchanging and archiving articles about physics before they got printed [4]. Today it is maintained by the Cornell University Library and contains about 989,000 open access documents about physics, mathematics, computer sciences and other areas.

Submitting single articles is only possible for registered authors which therefore have to grant arXiv the license to distribute their work [6]. The submitted work should ideally be a LaTeX file or a PDF file instead if the author does not use LaTeX.

Accessing data by crawling the sites is not allowed and will be blocked by the server if noticed [5]. One is advised to use the offered API or OAI-PMH interface instead.

3.10 Research Gate

Founded in 2008 in Hannover and Boston, ResearchGate rather is a kind of social network for scientists than a pure bibliographic database [68]. It currently has about 67 million publications and more than 5 million members, which can collaborate together, share publications and get statistics about their work. The most important part of ResearchGate is its author profiles. ResearchGate is accessible for registered users only and the registration is free.

According to [67], ResearchGate gets its publication and author data from other bibliographic databases, such as, in terms of computer sciences, CiteSeer (see Section 3.2). Additionally, registered users can upload publications to their profiles. There seems to be no way to access the data from ResearchGate.

3.11 Other Databases or Services

The **ACM Digital Library** is a bibliographic database that provides subscription based full text document access to all publications of the Association for Computing Machinery (ACM) and access to bibliographic citations from other major publishers [7]. It offers analyzation of references and citations and exports metadata to BibTeX and EndNote.

IEEE *Xplore* is a bibliographic database that provides access to more than 3 million publications on electrical engineering, computer science and electronics [39]. It mainly covers documents published by the Institute of Electrical and Electronics Engineers (IEEE) and its partners and offers a subscription model for accessing full text documents. IEEE *Xplore* also offers analyzation of references and citations and exports meta data to BibTeX and RIS.

Web of Science (WoS) (former known as Web of Knowledge) is a subscription-based interdisciplinary bibliographic database from Thomson Reuters stating that it is "the world's largest collection of research data" and the "only true citation index" [86]. It also claims that it objectively evaluates and selects its data sources. It is possible to suggest a repository for indexing. More information on the selection process can be found in [87].

The **Directory of Open Access Journals (DOAJ)** is a database that indexes open access, peer-reviewed journals [28]. It states that it uses a quality control system for its content and offers a quite sophisticated application form for journal publishers.

CiteULike is a bookmarking web-service for creating and sharing bibliographies [20]. It can import data from various sources (see [20] for a full list of currently supported sources) and allows the user to organize or share its bibliography with other users. CiteULike can also import BibTeX files and export the user's bibliography to BibTeX or RIS.

BibSonomy is a bookmarking web-service for creating and sharing bibliographies [13]. It can import data from various sites.[16] It offers a great amount of export formats, including but not limited to BibTeX, CSL-JSON, EndNote, DBLP XML, RIS and several plain text citations.

3.12 Overview of Indexing Requirements

After introducing these bibliographic databases, it is now important to summarize what to do in order to get publications indexed by the examined databases.

Table 2 shows that there is a wide variety of ways to get publications indexed by the databases. Some offer

[16] http://www.bibsonomy.org/scraperinfo

Database	How to get indexed	Supported formats
Google Scholar	by getting crawled	HTML Meta tags (Highwire, Dublin Core and others), HTML structure, PDF
CiteSeerX	by submitting a URL for crawling	PDF
BASE	by being a OAI Data Provider or registering as a source	OAI-PMH
MAS	by submitting single files (only registered users) or being indexed by one of the data sources	PDF, BibTeX
Arnetminer	by being indexed by DBLP or getting crawled	unknown
DBLP	only publishers or conference chairmen can submit data	DBLP Submission Format
CCSB	by submitting single files, registering as a source or being indexed by one of the data sources	BibTeX
arXiv	by submitting single files (only registered users)	PDF, LaTeX
ResearchGate	by submitting single files (only registered users) or being indexed by one of the data sources	BibTeX, RIS, MODS, DBLP-XML, PDF and others
Mendeley	by being indexed by one of the data sources or import button on website	HTML Meta tags, PDF

Table 2: How to get indexed by the databases

automatic ways like getting crawled or offering an OAI-PMH interface, but most of them want a manually uploaded file or a submitted URL. If you are an individual author submitting single files may be sufficient for you, but if you own a repository with many publications the automatic ways are the most interesting ones.

Another important factor to consider is the exchange of information that takes place between the databases. Almost all of the examined databases harvest data from other sources or distribute their own information to others. Figure 2 gives an overview of known relationships between the databases.

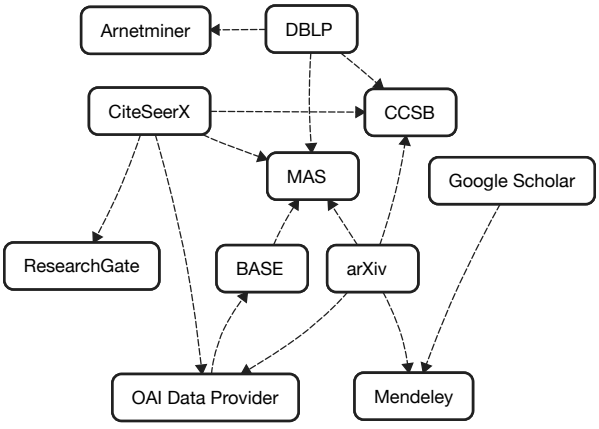

Figure 2: Known relationships between the databases

Please note that although Figure 2 may create the impression that these are "push" operations, they really are "pull" operations instead. The databases do not actively distribute their data to others, their data is accessed by others.

The three databases whose data is accessed most are CiteSeerX, arXiv and the DBLP Computer Science Bibliography. Unfortunately, getting CiteSeerX to index a lot of publications is a little bit difficult, because one has to manually submit a URL to every single PDF file to get crawled. And submitting publications to arXiv is still more difficult, as arXiv only accepts manual uploads of single PDF files. At least there is no need to take care of the DBLP Computer Science Bibliography if you are an author, because if your publication was published by a major publisher or is part of a big conference it should be already indexed. If not, you will most certainly not be able to change this.

Getting indexed by these three databases should cover all other examined databases with the exception of Google Scholar. But it can be assumed that Google's crawler will catch the information from one of the other covered databases anyway. Also, it should be easy to get crawled by Google Scholar with adding the required meta tags to the publication pages and following their guidelines (see Section 3.1).

As CiteSeerX and arXiv are registered OAI Data Providers, their data will be distributed among other OAI Service Providers. Unfortunately, BASE is the only officially registered OAI Service Provider of the examined databases, but there are others outside the scope of this paper. Providing an OAI-PMH interface may be a wise idea anyway, as it is a good way for others to get access to your data.

So if you want a lot of publications to get indexed, a reasonable approach requiring the least manual effort

would be to follow the guidelines of Google Scholar, provide an OAI-PMH interface, register as an official OAI Data Provider and maybe register as a data source for BASE and the Collection of Computer Science Bibliographies. To cover everything you would need to find a way to automatically submit single URL's or files to the other databases.

3.13 Overview of Crawling Possibilities

Another interesting aspect to summarize is how the data of the bibliographic databases can be accessed or harvested in order to embed their data into your repository.

Table 3 gives an overview of how to access data. It shows that popular ways to offer access are to provide an API or an OAI-PMH interface. In general, only information and metadata of the publications are accessible, but Microsoft Academic Search is an exception because it provides the greatest amount of accessible information and also offers the possibility to easily embed data on your own website via JavaScript. Others, like for example Google Scholar or the Collection of Computer Science Bibliographies, offer no way to access their data.

Table 3 also shows that half of the examined databases disallow the crawling of their sites for information. When also taking into account the fact that developing a good crawler is not an easy task, crawling becomes an even less recommendable approach to access data from these bibliographic databases.

4 Personal and Institutional Research Repository Systems

This section describes some widely used existing bibliographic repository systems that can manage or provide access to publication metadata and are intended to be used by individuals, small groups or whole institutions. Note that the list is not exhaustive and the focus lies on repository systems and not on management systems that guide through the whole process of submission, review and publication of manuscripts.

4.1 Open Harvester Systems

The Public Knowledge Project develops a free open source metadata indexing system called Open Harvester Systems (OHS) [66]. It is designed to index and expose all publications of an institution. It allows to harvest metadata from OAI-compliant databases, perform changes or normalization on the harvested data and index the resulting information in order to expose it using a customizable user interface. It also offers browsing and has a simple or advanced search interface. It requires an Apache web server with PHP and MySQL or PostgreSQL installed. OHS is released under an open source license and is available for free.

The Public Knowledge Project also offers Open Journal Systems (OJS)[17], Open Conference Systems (OCS)[18] and Open Monograph Press (OMP)[19] which are designed to help managing a journal, conference or book production, to assist with the submission of manuscripts and the publishing process and to expose the publications to the web.

4.2 Eprints

EPrints is a well-known software for building OAI-compliant bibliographic repositories developed by the School of Electronics and Computer Science at the University of Southampton [91]. It is released under an open source license and is available for free.

EPrints offers a customizable user interface that allows browsing by subject, division, author and year and has a simple or advanced search function. It also shows download statistics and lists of most downloaded items or top authors [91].

Users can register for an account in order to save search results or add and edit own publications. When adding a new entry, duplicate avoidance is used and files can be uploaded. Each publication can have multiple associated files including images, audio or video. A publication page includes meta tags for Google Scholar, metadata of the publication, the abstract and contact information of the author. Publications or even whole publication lists can be exported to Dublin Core, BibTeX, EndNote, RIS, HTML, plain text or as a RSS feed. EPrints also provides an OAI-PMH interface for accessing data [91]. Furthermore, its own meta tags format can be directly crawled by Google Scholar.

4.3 Digital Commons

Digital Commons [12] is a hosted open access institutional repository software for universities, colleges, law schools, and research centers. Furthermore, it combines traditional institutional repository features with tools for peer-reviewed journal publishing,

Content to Digital Commons can be uploaded in batch, by linking to external sites, or via a manual submit form. Digital Commons integrates user notification tools and social sharing. These include RSS feeds and automatic email alerts of new publications, mailing list manager to announce new publications and social shar-

[17] https://pkp.sfu.ca/ojs/
[18] https://pkp.sfu.ca/ocs/
[19] https://pkp.sfu.ca/omp/

Database	How to access data	Crawling
Google Scholar	not possible	no (according to Terms of Service)
CiteSeerX	by OAI-PMH or full download	yes (according to robots.txt)
BASE	by OAI-PMH (need to register)	no (according to robots.txt)
MAS	by API (need to register) or JavaScript	unknown
Arnetminer	by API	yes (according to robots.txt)
DBLP	by API or full download (DBLP XML)	currently yes (according to FAQ)
CCSB	can only become mirror site	yes (according to robots.txt)
arXiv	by API or OAI-PMH	no (according to Help Site)
ResearchGate	unknown	no (registration needed)
Mendeley	not possible	no (registration needed)

Table 3: How to access data from the databases

ing buttons. Digital Commons also collects and presents statistics about individual readership.

Digital Commons supports an OAI-PMH interface. Its own meta tags format called BE Press tags (named after the academic software firm *Berkeley Electronic Press* developing Digital Commons) is directly supported by Google Scholar.

4.4 Publications at Bielefeld University (PUB)

Publications at Bielefeld University (PUB) is a repository for all publications of the university available in German and English [89]. It claims that it is optimized for search engines like Google, Bing and Google Scholar and regularly imports from other databases, for example arXiv. PUB is also a data provider for other databases, as for example BASE.

PUB offers a search API and an OAI-PMH interface to access data. It also provides the export of metadata to Dublin Core, BibTeX and many more. Authors can embed a list of their publications in their own website [89, 90].

PUB is not a system that can be downloaded and installed on your own web server, but it is a good example for an institutional repository.

5 Further Related Work

Research in **digital libraries** is an active field. [43] addresses the challenges in digital library information-technology infrastructures. Besides the already described commercial or open source repository systems (see Section 4), some research prototypes like [61, 44] have been developed to maintain and manipulate bibliographies as well as to manage and share resources and knowledge. Also, [27] describes ways to optimize the **visibilty of publications** for search engine crawlers. Research was also done on **extracting information** from publications. [48] contains research results on extracting citation information, [50] on extracting researcher profiles and [95] describes the development of an information extractor for scholarly PDF documents.

Much has been researched to **utilize the OAI-PMH framework**: Many publications like [25, 46, 92, 1, 53, 51, 64] deal with resource harvesting and further analysis of the data like co-authorship networks [71]. Other contributions like [26] show that OAI-PMH data providers can enhance their repository with simple user interfaces on top of the OAI-PMH interface. [70, 45] discuss the benefits of integrating different protocols and standards with OAI. Some contributions [14, 21, 69] describe the easy integration of different data sources like relational databases or data crawled from webpages in the OAI-PMH framework. [52] describes an implementation for OAI data/service provider of individuals as well as institutional repositories. [63, 55] demonstrate the relative effectiveness of a range of search tools (some of which utilize OAI as well) in finding open access versions of peer reviewed academic articles on the world wide web. [93, 94] analyze the problems occurring during the OAI data-provider registration and validation service process. [73] gives an overview over the developments and trends of the OAI protocol.

6 Summary and Conclusions

This review provides an overview on popular data formats for exchanging bibliographic data about scientific publications in the computer science area. Furthermore, it analyzes bibliographic databases with special regard to their indexing requirements on one side and the possibilities to retrieve publication information from them on the other side. Finally we investigate personal and institutional research repository systems for their support to get their contained publications easily indexed by the previously described bibliographic databases.

Hence this review helps authors and publishers to find ways to increase the visibility of their (published) re-

search. This goal can be achieved by maximizing the inclusion of their publications in bibliographic databases, such that their contributions is easily found and cannot be overseen by other researchers.

Future work covers the implementation of institutional repository systems especially designed for the maximum inclusion in bibliographic databases and the analysis and presentation of the data accessible from bibliographic databases in new forms. Besides obvious issues like the support of widely used data formats and protocols for inclusion in bibliographic databases we want to address special functionalities like the (semi-) automatic filling of web forms of some of these bibliographic databases, such that publications can be quickly registered. There could be first an automatic search in these databases to register only those publications which are still missing. On the other hand we want to investigate how to provide a statistics page with deeper analysis also on the level of an author or institutional repository system. For this purpose we want to include information from other bibliographic databases like the citations of publications, which are included in our developed repository system. Also a systematic and targeted search for those publications citing the publications included in our repository system is a promising new research direction.

REFERENCES

[1] N. Adly, "An adaptive synchronization policy for harvesting OAI-PMH repositories," in *The First International Conference on Advances in Databases, Knowledge, and Data Applications*, Gosier, Guadeloupe, France, March 1-6, 2009, pp. 161–168.

[2] S. Alonso, F. Cabrerizo, E. Herrera-Viedma, and F. Herrera, "h-index: A review focused in its variants, computation and standardization for different scientific fields," *Journal of Informetrics*, vol. 3, no. 4, pp. 273 – 289, 2009.

[3] A. Apps, "Guidelines for encoding bibliographic citation information in dublin core metadata," http://dublincore.org/documents/dc-citation-guidelines/, June 2005, accessed: 2015-02-10.

[4] arXiv, "General information about arXiv," http://arxiv.org/help/general, accessed: 2014-11-18.

[5] arXiv, "Robots beware," http://arxiv.org/help/robots, accessed: 2014-11-18.

[6] arXiv, "To submit an article," http://arxiv.org/help/submit, accessed: 2014-11-18.

[7] Assocation for Computing Machinery, "ACM digital library," http://dl.acm.org/, accessed: 2015-02-20.

[8] BASE, "About BASE," http://www.base-search.net/about/en/index.php, accessed: 2014-11-12.

[9] BASE, "FAQ," http://www.base-search.net/about/en/faq.php, accessed: 2014-11-12.

[10] BASE, "Suggest repository," http://www.base-search.net/about/en/suggest.php, accessed: 2014-11-12.

[11] O. Ben-Kiki, C. Evans, and I. döt Net, "YAML ain't markup language version 1.2," http://yaml.org/spec/1.2/spec.html, October 2009, accessed: 2015-02-21.

[12] Berkeley Electronic Press (bepress), "Digital commons - open access institutional repository software," http://digitalcommons.bepress.com/, accessed: 2015-02-18.

[13] BibSonomy, "Getting started with BibSonomy," http://www.bibsonomy.org/gettingStarted, accessed: 2015-02-23.

[14] S. L. Bruzzo, P. Manghi, and A. Bardi, "Oaizer: Configurable OAI exports over relational databases," in *7th Research Conference o Metadata and Semantics Research*, Thessaloniki, Greece, November 19-22, 2013, pp. 35–47.

[15] CiteSeerX, "CiteSeerExtractor - A RESTful API for extracting information from scholarly documents," http://citeseerextractor.ist.psu.edu:8080/static/index.html, accessed: 2014-11-19.

[16] CiteSeerX, "CiteSeerX crawler," http://csxstatic.ist.psu.edu/about/crawler, accessed: 2014-11-19.

[17] CiteSeerX, "CiteSeerX data," http://csxstatic.ist.psu.edu/about/data, accessed: 2014-11-19.

[18] CiteSeerX, "History," http://csxstatic.ist.psu.edu/about/history, accessed: 2014-11-19.

[19] CiteSeerX, "Submit documents," http://csxstatic.ist.psu.edu/submit, accessed: 2014-11-19.

[20] CiteULike, "Frequently asked questions," http://www.citeulike.org/faq/faq.adp, accessed: 2015-02-19.

[21] A. Coleman, P. J. Bracke, and S. Karthik, "Integration of non-oai resources for federated searching in dlist, an eprints repository," *D-Lib Magazine*, vol. 10, no. 7/8, 2004.

[22] CrossRef, "Publisher fees," http://www.crossref.org/02publishers/20pub_fees.html, accessed: 2015-02-19.

[23] DCMI Wiki, "Creating metadata," http://wiki.dublincore.org/index.php/User_Guide/Creating_Metadata, accessed: 2014-11-18.

[24] DCMI Wiki, "User guide," http://wiki.dublincore.org/index.php/User_Guide, accessed: 2014-11-18.

[25] H. V. de Sompel, M. L. Nelson, C. Lagoze, and S. Warner, "Resource harvesting within the oai-pmh framework," *D-Lib Magazine*, vol. 10, no. 12, 2004.

[26] H. V. de Sompel, J. A. Young, and T. B. Hickey, "Using the OAI-PMH ... differently," *D-Lib Magazine*, vol. 9, no. 7/8, 2003.

[27] J. L. DeRidder, "Googlizing a digital library," *The Code4Lib Journal*, no. 2, 2008.

[28] DOAJ, "Directory of open access journals - about," http://doaj.org/about, accessed: 2015-02-21.

[29] A. Feder, "Bibtex format description," http://www.bibtex.org/Format/, 2006, accessed: 2014-11-11.

[30] A. Feder, "Bibtex special symbols," http://www.bibtex.org/SpecialSymbols/, 2006, accessed: 2014-11-11.

[31] M. Fenner, "Citeproc YAML for bibliographies," http://blog.martinfenner.org/2013/07/30/citeproc-yaml-for-bibliographies/, July 2013, accessed: 2015-02-21.

[32] Google, "About google scholar," https://scholar.google.com/intl/us/scholar/about.html, 2015, accessed: 2015-02-23.

[33] Google Scholar, "Inclusion guidelines for webmasters," http://scholar.google.de/intl/en/scholar/inclusion.html, accessed: 2014-11-11.

[34] R. Guenther and S. McCallum, "New metadata standards for digital resources: Mods and mets," *Bulletin of the American Society for Information Science and Technology*, vol. 29, no. 2, pp. 12–15, 2003.

[35] H. Guo, "How to use RESTful services on arnetminer," http://arnetminer.org/RESTful_service, accessed: 2014-11-12.

[36] J. E. Hirsch, "An index to quantify an individual's scientific research output," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 46, pp. 16 569–16 572, 2005. [Online]. Available: http://www.pnas.org/content/102/46/16569.abstract

[37] IDEAlliance, "PRISM faq," http://www.idealliance.org/specifications/prism-metadata-initiative/prism/faq, accessed: 2014-12-23.

[38] IDEAlliance, "PRISM introduction," http://www.prismstandard.org/specifications/3.0/PRISM_Introduction_3.0.htm, accessed: 2014-12-23.

[39] IEEE, "About IEEE xplore digital library," http://ieeexplore.ieee.org/xpl/aboutUs.jsp, accessed: 2015-02-19.

[40] International DOI Foundation, "DOI handbook - introduction," http://www.doi.org/doi_handbook/1_Introduction.html, accessed: 2015-02-19.

[41] International DOI Foundation, "DOI handbook - numbering," http://www.doi.org/doi_handbook/2_Numbering.html, accessed: 2015-02-19.

[42] International DOI Foundation, "FAQ," http://www.doi.org/faq.html, accessed: 2015-02-19.

[43] Y. E. Ioannidis, D. Maier, S. Abiteboul, P. Buneman, S. B. Davidson, E. A. Fox, A. Y. Halevy, C. A. Knoblock, F. Rabitti, H. Schek, and G. Weikum, "Digital library information-technology infrastructures," *Int. J. on Digital Libraries*, vol. 5, no. 4, pp. 266–274, 2005.

[44] H. N. Jerez, X. Liu, P. Hochstenbach, and H. V. de Sompel, "The multi-faceted use of the OAI-PMH in the lanl repository," in *ACM/IEEE Joint Conference on Digital Libraries*, Tucson, AZ, USA, June 7-11, 2004, pp. 11–20.

[45] J. Kaczmarek and C. C. Naun, "A statewide metasearch service using OAI-PMH and Z39.50," *Library Hi Tech*, vol. 23, no. 4, pp. 576–586, 2005.

[46] H. M. Khan, K. Maly, and M. Zubair, "Similarity and duplicate detection system for an OAI compliant federated digital library," in *9th European Conference on Research and Advanced Technology for Digital Libraries*, Vienna, Austria, September 18-23, 2005, pp. 531–532.

[47] C. Lagoze, H. van de Sompel, M. Nelson, and S. Warner, "The open archives initiative protocol for metadata harvesting," http://www.openarchives.org/OAI/openarchivesprotocol.html, 2008, accessed: 2014-12-04.

[48] S. Lawrence, C. L. Giles, and K. Bollacker, "Digital libraries and autonomous citation indexing," *IEEE Computer*, vol. 32, no. 6, pp. 67–71, 1999.

[49] M. Ley, "DBLP: some lessons learned," *Proceedings of the VLDB Endowment*, vol. 2, no. 2, pp. 1493–1500, 2009.

[50] J. Li, J. Tang, J. Zhang, Q. Luo, Y. Liu, and M. Hong, "Arnetminer: expertise oriented search using social networks," *Frontiers of Computer Science in China*, vol. 2, no. 1, pp. 94–105, 2008.

[51] X. Liu, K. Maly, M. Zubair, and M. L. Nelson, "*Arc* - an OAI service provider for digital library federation," *D-Lib Magazine*, vol. 7, no. 4, 2001.

[52] K. Maly, M. Zubair, and X. Liu, "Kepler - an OAI data/service provider for the individual," *D-Lib Magazine*, vol. 7, no. 4, 2001.

[53] P. Manghi, M. Mikulicic, L. Candela, D. Castelli, and P. Pagano, "Realizing and maintaining aggregative digital library systems: D-NET software toolkit and oaister system," *D-Lib Magazine*, vol. 16, no. 3/4, 2010.

[54] S. H. McCallum, "An introduction to the metadata object description schema (MODS)," *Library Hi Tech*, vol. 22, no. 1, pp. 82–88, 2004.

[55] F. McCown, X. Liu, M. L. Nelson, and M. Zubair, "Search engine coverage of the OAI-PMH corpus," *IEEE Internet Computing*, vol. 10, no. 2, pp. 66–73, 2006.

[56] Mendeley, "Information for publishers," http://www.mendeley.com/import/information-for-publishers/ter-support/, accessed: 2014-11-19.

[57] Mendeley, "What is mendeley?" http://support.mendeley.com/customer/portal/articles/227875-what-is-mendeley, accessed: 2014-11-19.

[58] Mendeley, "Why use the platform?" http://dev.mendeley.com/overview/why_use_platform.html, accessed: 2014-12-05.

[59] Microsoft Research, "About microsoft academic search," http://academic.research.microsoft.com/About/Help.htm, accessed: 2014-11-19.

[60] U. Müller, A. Powell, and P. Cliff, "OAI for beginners - the open archives forum online tutorial," http://www.oaforum.org/tutorial/english/intro.htm, 2003, accessed: 2014-12-04.

[61] A. Naak, H. Hage, and E. Aïmeur, "Papyres: A research paper management system," in *10th IEEE International Conference on E-Commerce Technology / 5th IEEE International Conference on Enterprise Computing, E-Commerce and E-Services*, Washington, DC, USA, July 21-14, 2008, pp. 201–208.

[62] NISO, "JATS: Journal article tag suite," http://www.niso.org/apps/group_public/download.php/10591/z39.96-2012.pdf, accessed: 2014-12-05.

[63] M. Norris, C. Oppenheim, and F. Rowland, "Finding open access articles using google, google scholar, oaister and opendoar," *Online Information Review*, vol. 32, no. 6, pp. 709–715, 2008.

[64] T. Ong and J. J. Leggett, "Collection understanding for OAI-PMH compliant repositories," in *ACM/IEEE Joint Conference on Digital Libraries*, Denver, CO, USA, June 7-11, 2005, pp. 258–259.

[65] O. Patashnik, "BibTEX yesterday, today, and tomorrow," *TUGboat*, vol. 24, no. 1, pp. 25–30, 2003.

[66] Public Knowledge Project, "Open harvester systems," https://pkp.sfu.ca/ohs/, accessed: 2015-02-11.

[67] ResearchGate, "Adding publications," https://explore.researchgate.net/display/support/Adding+publications, accessed: 2014-12-05.

[68] ResearchGate, "Researchgate - who we are," http://www.researchgate.net/aboutus.AboutUsPress.downloadFile.html?name=rg_fact_sheet.pdf, accessed: 2014-11-19.

[69] J. A. Sánchez, A. Razo, J. M. Córdova, and A. Villegas, "Dynamic generation of OAI servers," in *ACM/IEEE Joint Conference on Digital Libraries*, Chapel Hill, NC, USA, June 11-15, 2006, pp. 258–259.

[70] R. Sanderson, J. Young, and R. LeVan, "SRW/U with OAI: expected and unexpected synergies," *D-Lib Magazine*, vol. 11, no. 2, 2005.

[71] P. Schaer, T. Lüke, P. Mayr, and P. Mutschke, "An oai-pmh-based web service for the generation of co-author networks," *CoRR*, vol. abs/1301.7443, 2013. [Online]. Available: http://arxiv.org/abs/1301.7443

[72] SELFHTML, "Meta-angaben nach dublin core," http://de.selfhtml.org/html/kopfdaten/meta.htm#dublin_core, accessed: 2014-12-07.

[73] S. L. Shreeves, T. G. Habing, K. Hagedorn, and J. A. Young, "Current developments and future trends for the OAI protocol for metadata harvesting," *Library Trends*, vol. 53, no. 4, pp. 576–589, 2005. [Online]. Available: http://www.ideals.illinois.edu/bitstream/handle/2142/1754/Shreeves576589.pdf

[74] J. Tang, "Arnetminer: search and mining of academic social networks," http://arnetminer.org/introduction, accessed: 2014-11-12.

[75] The Collection of Computer Science Bibliographies, "About the collection," http://liinwww.ira.uka.de/bibliography/index.html#about, accessed: 2014-11-12.

[76] The Collection of Computer Science Bibliographies, "Add references to the computer science bibliography collection," http://liinwww.ira.uka.de/bibliography/Contrib/direct.html, accessed: 2014-11-12.

[77] The Collection of Computer Science Bibliographies, "Adding a complete bibliography to the computer science bibliography collection," http://liinwww.ira.uka.de/bibliography/Contrib/biblio.html, accessed: 2014-11-12.

[78] The Collection of Computer Science Bibliographies, "Introduction," http://liinwww.ira.uka.de/bibliography/Introduction.html, accessed: 2014-11-12.

[79] The DBLP Team, "Am i allowed to crawl the dblp website," http://www.informatik.uni-trier.de/~ley/faq/Am+I+allowed+to+crawl+the+dblp+website.html, accessed: 2014-11-11.

[80] The DBLP Team, "The DBLP computer science bibliography," http://www.informatik.uni-trier.de/~ley/db/welcome.html, accessed: 2014-11-11.

[81] The DBLP Team, "How can i enter my publications to dblp," http://www.informatik.uni-trier.de/~ley/faq/How+can+I+enter+my+publications+to+dblp.html, accessed: 2014-11-11.

[82] The DBLP Team, "How can i submit meta data for a complete journal or conference?" http://www.informatik.uni-trier.de/~ley/faq/How+can+I+submit+meta+data+for+a+complete+journal+or+conference.html, accessed: 2014-11-11.

[83] The Library of Congress, "MODS: Uses and features," http://www.loc.gov/standards/mods/mods-overview.html, accessed: 2014-12-05.

[84] The Library of Congress, "MARC21 format for bibliographic data - introduction," http://www.loc.gov/marc/bibliographic/bdintro.html, October 2006, accessed: 2015-02-21.

[85] The Open Archives Initiative, "Data provider validation and registration," http://www.openarchives.org/Register/ValidateSite, accessed: 2014-12-04.

[86] Thomson Reuters, "The citaton connection - real facts," http://wokinfo.com/citationconnection/, accessed: 2015-02-21.

[87] Thomson Reuters, "The repository selection process," http://wokinfo.com//products_tools/multidisciplinary/dci/selection_essay/, accessed: 2015-02-21.

[88] Thomson Reuters, ""RIS" format documentation," http://www.refman.com/support/direct%20export.zip, accessed: 2014-11-19.

[89] Universität Bielefeld, "PUB – Publikationen an der Universität Bielefeld," http://pub.uni-bielefeld.de/#about, accessed: 2015-02-11.

[90] Universität Bielefeld, "PUB developer 2.0 documentation," http://pub.uni-bielefeld.de/doc/api/index.html, accessed: 2015-02-11.

[91] University of Southampton, "EPrints - Digital Repository Software," http://www.eprints.org/software/, accessed: 2015-02-11.

[92] U. Waltinger, A. Mehler, M. Lösch, and W. Horstmann, "Hierarchical classification of OAI metadata using the DDC taxonomy," in *Workshop on Advanced Language Technologies for Digital Libraries*, Viareggio, Italy, June 15, 2009, pp. 29–40.

[93] S. Warner, "The OAI data-provider registration and validation service," *CoRR*, vol. abs/cs/0506010, 2005. [Online]. Available: http://arxiv.org/abs/cs/0506010

[94] S. Warner, "The OAI data-provider registration and validation service," in *9th European Conference on Research and Advanced Technology for Digital Libraries*, Vienna, Austria, September 18-23, 2005, pp. 491–492.

[95] K. Williams, L. Li, M. Khabsa, J. Wu, P. Shih, and C. Giles, "A web service for scholarly big data information extraction," in *Conference on Web Services*, June 2014, pp. 105–112.

[96] R. Zelle, "CSL home," http://citationstyles.org/, accessed: 2014-12-18.

AUTHOR BIOGRAPHIES

Arne Kusserow was born in Hildesheim, Germany in 1991. He currently studies Informatik (Computer Science) at the University of Lübeck and writes his bachelor thesis about Online Indexing of Scholarly Publications in which he also develops a web application for exposing scholarly publications to the web and to other bibliographic databases.

Dr. Sven Groppe earned his diploma degree in Informatik (Computer Science) in 2002 and his Doctor degree in 2005 from the University of Paderborn. He earned his habilitation degree in 2011 from the University of Lübeck. He worked in the European projects B2B-ECOM, MEMPHIS, ASG and TripCom. He was a member of the DAWG W3C Working Group, which developed SPARQL. He was the project leader of the DFG project LUPOSDATE, and is currently the project leader of two research projects, which research on FPGA acceleration of relational and Semantic Web databases. His research interests include Semantic Web, query and rule processing and optimization, Cloud Computing, peer-to-peer (P2P) networks, Internet of Things, data visualization and visual query languages.