Project bwDataDiss

Wiebke Beckmann, Matthias Bonn, Tobias Kurze, Klaus Rechert, Saher Semaan, Dennis Wehrle

Project bwDataDiss: bwData for Dissertations

Wiebke Beckmann[§], Matthias Bonn[**], Tobias Kurze[*], Klaus Rechert[‡], Saher Semaan[§], Dennis Wehrle[‡]
[*] Karlsruhe Institute of Technology - Library
Email: kurze@kit.edu
[**] Karlsruhe Institute of Technology - Steinbuch Centre for Computing
Email: matthias.bonn@kit.edu
[‡] University of Freiburg - Institute for Computer Science
Email: firstname.lastname@rz.uni-freiburg.de
[§] University of Freiburg - Library
Email: firstname.lastname@ub.uni-freiburg.de

Abstract

Nowadays research relies more and more on data to achieve progress in various scientific domains. To understand and to be able to reproduce results, it is essential that the underlying research data is available to scientists - even after a relatively long time. bwDataDiss is an effort to provide infrastructure and services for a very specific group of researchers - namely PhD students - to enable them to store and archive their research data and also to make it available to other researchers.

Keywords

dissertation, research-data, long-term storage, characterization, open access

I. Introduction

bwDataDiss is a three year project funded by the Ministry for Science and Art of Baden-Württemberg. Project partners are both the university libraries and computing centres of Freiburg and Karlsruhe.

In the context of their doctoral dissertation, PhD students often produce research data. As the scientific society becomes more and more aware of the importance of verification of research results, the general need to build up digital infrastructures to archive and enable access to research data arises. But currently libraries often lack the digital infrastructure to handle research data as they are often heterogeneous with regard to contents and filetypes.

Traditionally libraries have been involved and play an important role in the dissertation process. They have the respective workflows in place and know the procedures. However, those workflows differ from library to library and may sometimes be relatively complex. To capture some of this diversity the university libraries of Karlsruhe and Freiburg bring in their knowledge to bwDataDiss.

Libraries usually don't have the means to store nor to analyze or work with large amounts of data. The SCC[1] provides the IT infrastructure and systems that permits bwDataDiss to store and archive research data, while the computing centre of the University of Freiburg performs - what we call - a characterization of the research data.

II. Related work

There is a number of projects working on the long-term preservation of research data and of course their results are continuous integrated into bwDataDiss. The following list has a strong focus on Germany:

The project RADAR[2] or Research Data Repositorium, provides infrastructure for research data management and is a joint effort of the FIZ Karlsruhe, the Steinbuch Centre for Computing (SCC), the Ludwig-Maximilians-Universität Munich (LMU) and the German National Library of Science and Technology (TIB).

[1] Steinbuch Centre for Computing
[2] https://www.radar-projekt.org/

MO|REdata is another research data project with a very specific mission in the field of sports sciences. MO|REdata has built up an infrastructure to collect and provide motor skills data as well as normative data.

An extensive survey was undertaken in the project bwFDM[3] by the nine universities in the state of Baden-Württemberg in order to get an idea of what researchers need and expect from institutional research data management.

FreiDok plus[4] is the institutional repository of the University Library of Freiburg. During the past few years it has been developed into a current research information system in order to represent the complete landscape of research at the university. Since then, FreiDok plus also enables the publication and archiving of research data and provides clever workflows for its users.

III. Concept and Integration scenarios

bwDataDiss enables university libraries in the State of Baden-Württemberg to archive research data together with the final dissertation. The main concept is that the associated libraries remain contact point for PhD students and potential users of bwDataDiss at all times, it has not been designed to work independent of libraries - quite the contrary: bwDataDiss relies on libraries to provide crucial services and counselling to PhD students and researchers who want to store and archive their research data. Also librarians are and will remain in charge of capturing and controlling the required metadata.

For sake of simplicity and without loss of generality, from now on we describe the systems, relations and interactions between bwDataDiss and just a single library.

A. Principles

The design of bwDataDiss follows the following principles:

- Keep it easy for our primary customers: the researchers and especially Ph.D. candidates
- Ensure data integrity at all times
- Allow for a flexible integration with library systems

B. Separation of duties

There is a strict separation of duties (also see figure 1) between bwDataDiss and the library:

Library:

- Help-desk and services for Ph.D. candidates: The place to go for Ph.D. candidates
- Acquisition and control of metadata

bwDataDiss:

- Archivation of data
- Provide archived research-data to scientific community
- Perform characterization and provide its results

As libraries already play a crucial role in the dissertation process, it is consistent that the transfer of research data - being considered to be part of the dissertation – is integrated in that same process. The workflow consists roughly of the following steps (order negligible):

- PhD candidate transfers dissertation script (library website)
- PhD candidate provides metadata for dissertation script (library website)

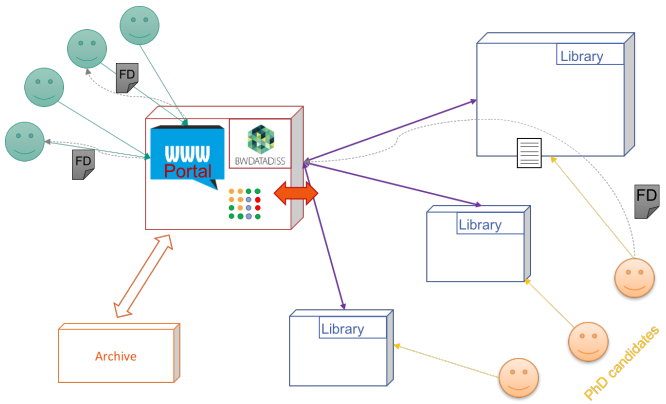

Figure 1. Separation of duties

[3]https://bwfdm.scc.kit.edu/ - Baden-Württemberg Forschungsdatenmanagement = research data management

[4]https://www.freidok.uni-freiburg.de/

- Library checks the provided data and contacts the PhD candidate if necessary

When considering research data, at least two additional steps are needed:

- Transfer research data (to bwDataDiss or to the library)
- Provide metadata for research data (to bwDataDiss or the library)

C. Integration scenarios

1) Upload: This is where bwDataDiss comes in. bwDataDiss supports several possibilities to integrate its services with library systems in order to transfer the research data from the Ph.D. candidate to the long-term storage.

There are two basic models to transfer the data:

- Either the data is transferred directly from the PhD candidate to bwDataDiss or
- The data is temporarily stored on library servers and then later on transferred from the library to bwDataDiss.

Either way, it is entirely up to the library to check the provided metadata.

a) Details on direct data transfer: There are two possibilities to realize the direct data transfer from the Ph.D. candidate to bwDataDiss and it's up to the library to decide which solution to implement. For a more coherent presentation of the library website, the upload component, which actually realizes the data transfer to bwDataDiss can be integrated in the library website. Although the component is on the library website, it will upload the data directly to bwDataDiss.

Another less seamless integration would be, to forward a user to the uploader on the bwDataDiss website. The upload could then happen there, but it would be apparent to the user that he or she has left the library website. See section IV-C for more details on the uploader.

2) Metadata: bwDataDiss supports different models to input metadata. The first and easiest possibility consists of filling in a web form on the bwDataDiss website. This obviously requires a user to navigate to bwDataDiss and to leave the library website.

To keep bwDataDiss in the background, the metadata can either be pushed from the library or actively pulled by bwDataDiss from the library. More details can be found in section V-A and section IV-B.

IV. Implementation

bwDataDiss is implemented using PHP and Symfony, a Web Application Framework. Also JavaScript is used to provide certain functions, such as a reliable upload of huge files. bwDataDiss distinguishes three types of basic user roles: library role, bwdd-admin role and regular user role. A user that possesses the bwdd-admin role can set roles respectively promote regular users to library users or to admin users.

A. Web-Frontend

In principle, bwDataDiss is designed as a self service portal, though a library can restrict its users to certain functions. It can be accessed at https://bwdatadiss.kit.edu. The entire portal is available in German as well as in English. Please note, that at the time of the writing the portal is still under development.

B. API and File Management

We have a state-of-the-art REST xml and json based API which uses key authentication. It provides functions to create datasets, edit their metadata, upload files using file chunks, query datasets by state and query file data, etc. It also provides functions to query archivation tasks and to report their successful execution. Almost all functionality that is provided on the bwDataDiss website can also be leveraged through the use of the API.

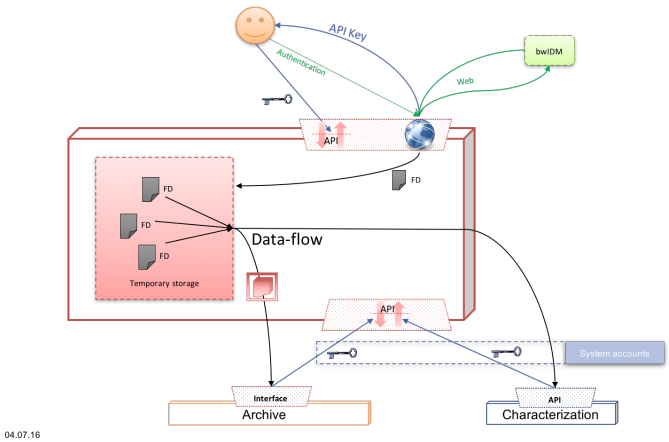

Figure 2. API, keys and components

The API can only be accessed over HTTPS and requires an authentication header (X-API-KEY) to be present and set to a correct API key. Every user has its own unique API key(s) and can thus be identified by it. An API key is generated upon first succesfull login of a user. If the API call returns values, they can either be obtained in JSON or XML format. The default is JSON, but by appending .xml to the URL the format can be set to XML.

Not only external users use the API, but also some internal (asynchronous) components of bwDataDiss use the API to query or set the status of entities.

Figure 2 illustrates the use of the API and the keys respectively.

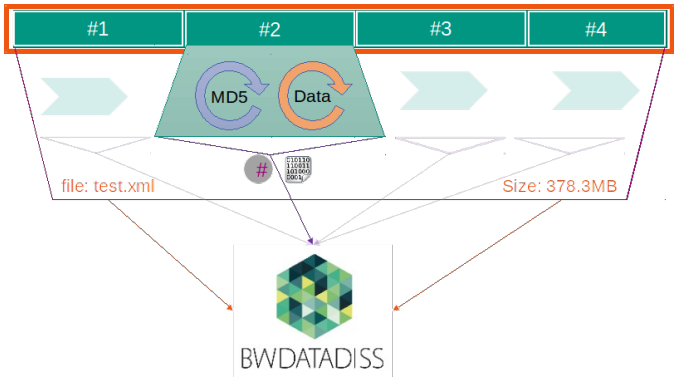

Figure 3. Chunked upload of a a file to bwDataDiss

C. Upload Component

bwDataDiss comes with a potent upload component. As outlined in section III, data integrity is crucial for bwDataDiss and has to be ensured by the uploader. The uploader transfers the data from the Ph.D. candidate to bwDataDiss. It is able to upload files of any size, allows to resume uploads and calculates checksums to ensure data integrity.

To upload a file, it is split into chunks of a certain size and the chunks are then treated separately: first, a checksum is calculated, then the data of the chunk, together with its checksum is transferred to bwDataDiss. An example is depicted in figure 3. To communicate details, such as chunk-size, file-size, checksums and the actual data the uploader talks to the bwDataDiss API. The uploader is written in JavaScript, able to utilize multiple threads and to handle files in parallel.

D. User Authentication and Access using bwIDM

To identify users, bwDataDiss relies on another service called Föderiertes Identitätsmanagement der baden-württembergischen Hochschulen (bwIDM)[5]. bwIDM allows users at a BW-university - to be authenticated using Shibboleth[6] (SAML-based Web-SingleSignOn). For alumni users with no access to such a federated account (or external PhD students, for example) we implemented an invitation mechanism to create local accounts. This invitation has to be triggered for every external student by the corresponding library.

The access to the archive itself is hidden from end users, the archive login is done using a single non-bwIDM LDAP system account with public key authentication, using a directory/file naming scheme to map datasets to end users (see section “Archive Integration” below). As a result, end user’s ACLs cannot be enforced at archive, this must be done on web server level.

E. Authentication and Integration by/with foreign libraries

bwDataDiss provides a range of services that may be used by libraries. The most simple and basic integration scenario just involves the archivation of data and the harvesting of the corresponding metadata.

As mentioned in section III the associated libraries are the starting point of their PhD students, it means after authentication (as described below) the user submits the publication (PhD thesis), the research data and the related metadata according to the library workflow. The library can afterwards and in an asynchronous transmission push the data to bwDataDiss using the API described in section IV-B. Another advanced integration can adapt a distinguished bwDataDiss service in the library publication workflow, such as the characterization (chapter 4.5). After submission the complete data set (PhD thesis, corresponding research data and metadata for both) through the user, the library pushes

[5]https://www.bwidm.de/
[6]http://shibboleth.net/

the research data to bwDataDiss using the API for characterization and then querying the result from bwDataDiss again through the API to make local decisions according to the research data before the final push of the whole data set to bwDataDiss.

To delegate authenticated user-contexts from a library frontend to the API-key based bwDataDiss backend, a library first authenticates a user with Shibboleth Web-SSO. Then it can authenticate itself using an HMAC-SHA256 based signature to pass the user's Shibboleth SAML token to bwDataDiss and to query a bwDataDiss API-key for the user. So asynchronous API calls can be done from the library frontend to the bwDataDiss backend without losing the initially established end users context.

Another scenario is described in figure 4. Users can be delegated for authentication to their home university Identity Provider (IdP) using the federated bwIDM infrastructure. In contrary to the previous scenario user-contexts will not be delegated from the library on behalf of users, but users will be delivered to home the Identity Provider (IdP). In our case Shibboleth IdPs, which are all in the same bwIDM federation.

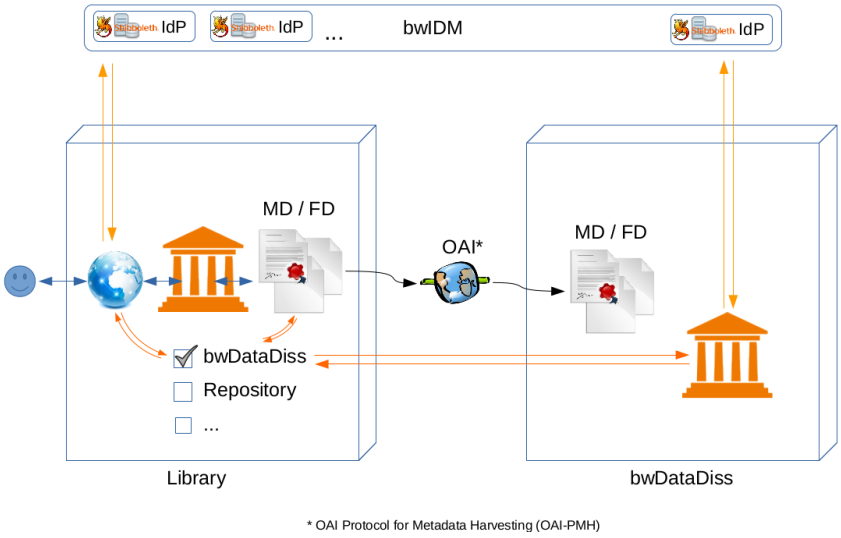

Figure 4. Delegation to home IdP

F. Data Characterization

Content and data formats vary widely between different disciplines. Since the bwDataDiss service is planned as a university-wide, interdisciplinary service offer, the service provider needs to accept any domain-specific data format while still maintaining quality standards (e.g. completeness, usability and accessibility of stored data) as well as prepare for long-term preservation activities. In particular long-term preservation (in today's fast technical lifecycle, even the mandatory 10 years data redemption policy[7] may become a major challenge) activities to maintain accessibility and ideally allow re-use (e.g. for reproduction of scientific results or comparison with recent research) of the data stored, risks posed by accepted data-sets and in particular risks of enclosed data formats need to be assessed and maintained. For this, a data format characterization service has been integrated into the bwDataDiss service model.

1) Towards a Data Risk Policy: The goal of the characterization service is to provide an overview on preservation risks w.r.t. access, re-use and verification of the data set's content. For this, the research data's logical and structural representation - the file formats - need to be assessed, in particular, if the format documentation is available and what kind of software is necessary to render or retrieve the file's information content. Based on this information predictions on (long-term) (re-)usability can be made, a risk register can be set up and, if appropriate, preservation tasks can be planned and prepared (e.g. preparations for file format migration).

The results of the characterization service can be used either as pre-ingest check, e.g. as a tool for feedback to an initial submission, i.e. flagging unsustainable, unknow or otherwise difficult file-formats. Based on this feedback, individual researchers can be advised to re-consider their file-format choices (if possible) and their awareness can be raised on the un-sustainability of their format choices. Furthermore, the characterization results may be used to guide a software collection, required to render certain data sets or to prepare an emulation or virtualization strategy.

[7]DFG Empfehlungen zur Sicherung guter wissenschaftlicher Praxis, http://www.dfg.de/download/pdf/dfg_im_profil/reden_stellungnahmen/download/empfehlung_wiss_praxis_0198.pdf

2) Data Characterization: bwDataDiss provides a RESTful characterization service for analyzing data sets. We have chosen FITS[8] as a file characterization meta-tool, as it wraps various file characterization tools into a single and customizeable Java framework.

A characterization request is issued by POSTing a JSON object containing a reference (URL) to the data set and a reference to an optional policy file. For efficiency reasons we require to unpack data sets and wrap the resulting file-tree within an ISO9660 container (CD-ROM / DVD format). This way, a prior download of the data set for characterization is not necessary. Instead, the remote file is mounted to the local filesystem and data required for file format characterization is only transferred on request. For HTTP Urls we use range requests[9] for random access to files with a data set. Since user data is only cached in memory, parallel requests can be handled without considering temporary disk space constraints.

Example request:
http://bwdatadiss.eaas.uni-freiburg.de:8080/bwdatadiss/FileFmtCheck/init

```
{
    "objectUrl": "http://bwdatadiss/myset.iso",
    "policyUrl": "http://bwdatadiss/base-policy.txt"
}
```

The service will immediately return a session id, which can be used for querying the status of the characterization request. Depending on the object size, the characterization may take some time to finish. By using the session id, the requesting client is able to retrieve the characterization result. If the characterization is not finished, the client is required to repeat the request later.

The bwDataDiss' characterization result is a file format distribution, i.e. the number of files found per file format (PRONOM ID[10]).

Example request:
http://132.230.3.211:8080/bwdatadiss/FileFmtCheck/getResultSummary?sessId=5

```
{
    "summary": [
    {
      "type": "x-fmt/111",
      "value": "GREEN",
      "count": "242"
    },
    {
      "type": "fmt/16",
      "value": "GREEN",
      "count": "2"
    },
    {
      "type": "x-fmt/411",
      "value": "RED",
      "count": "1"
    }
}
```

Also a detailed result can be requested, which contains a list of files (including their relative path) for each PRONOM ID. If also a policy file was provided, a "verdict" on for each format is added. In the example above, the policy file contains traffic-light coloring, assigning "x-fmt/111" (Plain Text) and "fmt/16" (PDF) the color "GREEN"

[8]File Information Tool Set (FITS), http://projects.iq.harvard.edu/fits/home
[9]Hypertext Transfer Protocol (HTTP/1.1): Range Requests, https://tools.ietf.org/html/rfc7233
[10]PRONOM file format registry, http://www.nationalarchives.gov.uk/PRONOM/Default.aspx

and “x-fmt/411” (Windows Executable COFF) the color “RED”. In a future version, it will be possible to reference multiple policy files, published by different institutions. This way, it becomes possible to share the work maintaining a comprehensive file format policy and covering the great variety of file formats typically found in research data sets.

G. Archive Integration

The High Performance Storage System (HPSS[11]) based archive offers a standard hierarchical file system access by SFTP to its integrated large disk cache, which hides the underlying tape-based long term storage system. So the usage of the archive by the bwDataDiss (bwDD) system has to be done using remote access to the SFTP-Interface. This integration is done via an automated FUSE based SSHFS mount point in the bwDD host system, which hides the SFTP protocol and offers a standard local directory which can be accessed with any local file system tools or programming libraries; but with one essential constraint: Access to this mount point can be significantly slower than local filesystem or NFS/CIFS mounts, especially when remote files are stored on archive tape instead of archive disk cache (HPSS does both, depending on file size and access frequency/profile). As a result, writing new files to the archive normally is performed with sufficient speed, but reading a file which has already been moved from the disk cache to the tape system can be a long running task with unpredictive latencies. So we had to realize an asynchronous coupling of the archive to the web frontend.

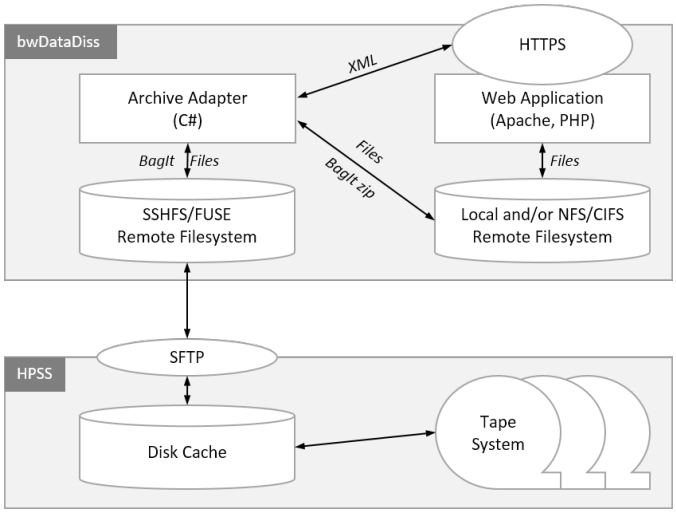

Figure 5. Archive Integration (HPSS System strongly simplified)

The decoupling is done by detaching the archive integration/connection from the apache/php based web service. Because asynchronous background threads are not possible in php and provide the risk of blocking the web service request handling, we implemented a separate backend worker process. This backend service worker uses the bwDD REST API to query archivation tasks (insert, read) and maps the internal bwDD data model to remote SSHFS-mounted (potentially delayed and slower) filesystem (figure 5).

There, it creates (uncompressed) BagIt[12] directories where it copies uploaded files from local filesystem (see figure 6). Every metadata (bibliographical and technical) including transfer logs are also written to the archive, which leads to a self-describing archive file system structure (grouped by libraries), which is an essential point for a long term archive. In the naming scheme, [Library-Name] and [DataSet-ID] are unique within the bwDD system, [User-EPPN] is unique at least within the user’s home-library.

To read an archived dataset, the complete BagIt is asynchronously copied back from remote SSHFS mount point to a fast local filesystem cache where the web server has direct access to serve the files for download, optionally as zip containing all files including the BagIt description and all other metadata files (but no transfer

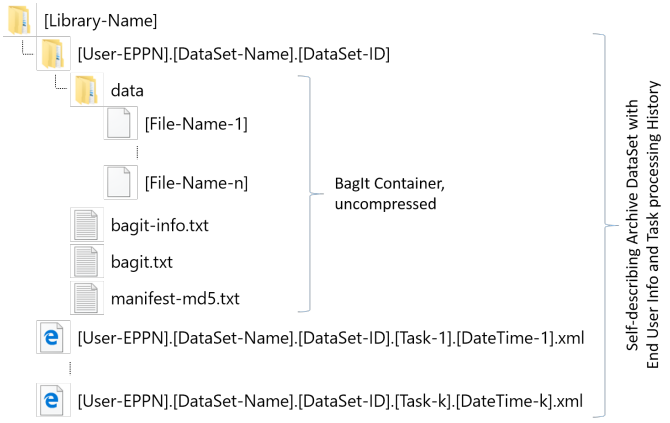

Figure 6. Archive Directory and File Structure

[11] http://www.hpss-collaboration.org/

[12] https://tools.ietf.org/html/draft-kunze-bagit-13

logs). All archivation tasks queried from the bwDD REST API are verified by comparing checksums. When a transfer is completely done and fully verified, the success is reported to the API, where the web server presents the results to the user. If an error occurs during an archivation task, the failure result is not reported to the API and the task is replayed after an increasing waiting period.

V. Metadata

bwDataDiss is open be be used by the university libraries in the State of Baden-Württemberg and bwDataDiss' metadata scheme pays tribute to this fact by being relatively generic. The scheme is partially inspired by the RADAR metadata kernel[13]. Some metadata has to be provided coercively, some not. Table I lists the available metadata items.

TABLE I. Metadata items

Mandatory	**Optional**
Title	Additional Title
Creators	Contributors
Abstract	Resource type
Keywords	Rights holder
Readme	Embargo date
Creation Year	Additional metadata
Publisher	
Publication	
Classification	
License	

A few of the mentioned items might need some explanation:

Readme: We believe, that especially for research data, it is important to explain how files are organized and how the data can and should be used. The text that is provided as readme will not just be stored in the database, but also be added as a file to the research data. This ensures, that potentially important information about the research data is always available as a text file besides the other files.

Classification: bwDataDiss uses a quite unusual classification from the DFG. This classification is more detailed concerning technical disciplines and therefore also suited for universities with a technical focus. Still it is generic enough to be used by any (non-technical) university as well.

Resource Type: Borrowed from the RADAR metadata kernel, it is one of the following items, specifying the type of the research data: audiovisual, collection, dataset, image, model, software, sound, text, workflow, other.

License: bwDataDiss promotes and is committed to Open Access and thus comes with CC-BY and CC-BY-SA licenses to be chosen from. However we understand that there might be circumstances when those licenses are not eligible and therefore allow libraries to add new licenses.

Embargo date: There are certain circumstances when it might be necessary to forbid access to research data, for example when a patent application is still pending. Thus, bwDataDiss allows to specify an embargo data, prohibiting the access to the research data until the given date.

A. Acquisition

As mentioned in section III, it is up to the library to collect and check the metadata. bwDataDiss keeps a copy of the metadata along with the corresponding research data. To obtain the metadata, bwDataDiss regularly queries the library repositories using OAI-PMH[14]. This way, even if changes should not happen very often, bwDataDiss always has the latest metadata.

If the library does not provide an adequate interface, it can push the metadata using the bwDataDiss API.

VI. Outlook

bwDataDiss is still under development und currently not yet ready for production. The characterization component is not completely finished and needs some attention during the next few weeks. However we plan to go live before the end of the year. Currently we focus on the integration with the library systems at KIT and at the university of Freiburg - a crucial step for bwDataDiss to be put into service.

[13]https://www.radar-projekt.org/display/RE/2014/11/14/RADAR+Metadata+Kernel+v+0.2

[14]https://www.openarchives.org/pmh/ Protocol for Metadata Harvesting

About the authors

Matthias Bonn Since 2002, Matthias Bonn works at the KIT and its predecessors, the University of Karlsruhe and the Forschungszentrum Karlsruhe, respectively. Starting with the first Germany wide e-learning projects, he did his PhD in 2008 on distribution of computing-intensive scientific simulation tasks in heterogeneous and unreliable environments. Since 2009 he is employed at the KIT Steinbuch Centre for Computing, mainly involved in Cloud computing projects with focus on self-service VM portals, based on VMware vSphere. Since 2014, he is working in the SCC's scientific data management group, is responsible for the archiving parts of bwDataDiss and realized the archive adapter which connects the bwDataDiss web parts to the backing storage system HPSS.

Tobias Kurze studied computer sciences at Karlsruhe Institute of Technology (KIT) (former University of Karlsruhe) from 2004 to 2009 and at Institut national des sciences appliquées de Lyon (INSA Lyon) from 2007 until 2009. Works at KIT since 2010 and is a former research associate at the Steinbuch Centre for Computing (SCC). Until 2015 he was part of the research group Cloud-Computing with focus on distributed-, grid- and cloud-computing, operated and tested an OpenStack cluster, and researched the efficient use of cloud resources. Contributed to tho Software Cluster projects Emergent and InDiNet and led work packages in aforementioned projects. Now research associate at the KIT library and coordinator of the project bwDataDiss, which provides infrastructure and services to libraries and researchers to store and archive research data.

Klaus Rechert is currently a postdoctoral researcher at the professorship in communication systems of the Institute for Computer Science at Freiburg. As a project manager he currently oversees a federal project "Statewide development of coordinated structures for indexing and re-use of research data" and is scientific consultant in the DFG-LIS project "Reading Room Access of Multimedia-Object using Emulation". His research focus is on functional preservation of scientific processes, privacy protection in context of complex data and digital forensics. From Oct 2011 - Dec 2013 Klaus was the project manager of the bwFLA project, a two-year project funded by the state Baden-Württemberg, leveraging emulation for access and migration tasks in digital preservation. From 2006-2009 Klaus was a lecturer at the University of Freiburg teaching "Programming with C" and "Introduction to C++". In 2010 Klaus was a guest lecturer at Malta College of Art, Science & Technology. From Oct. 2010 - Mar. 2011 he was a visiting researcher at the National Institute of Informatics (NII) in Tokyo, Japan. 2008-2010 he was a software developer on the PLANETS EU-FP6 project. In 2007 and 2008 Klaus was as open source software developer and maintainer on the MING-project, sponsored by Lulu.com Inc. Raleigh, NC and OpenMediaNow Foundation, Rollinsville, CO. In 2006 Klaus received the EXIST-Seed scholarship sponsored by the Federal Ministry of Economics and Technology. Klaus studied Computer Science and Economics at the University of Freiburg and received a Diploma in Computer Science in 2005. Since 2013 Klaus holds doctoral degree from the University of Freiburg.

Saher Semaan joined in 2015 the eScience department at the Library of the University of Freiburg. He was from 2007-2014 scientific assistant at the professorship in Communication Systems of the Institute for Computer Science at Freiburg. Amongst others he coordinated 2011-2013 the bwIDM-Project (https://www.bwidm.de/) at the University of Freiburg and was from 2008-2014 a lecturer at the Center for Key Qualifications (Zentrum für Schlüsselqualifikationen, ZfS) at the same university. His background spans the fields of federated identity management, scientific IT-infrastructures, databases and operating systems. He received his diplom degree in computer science from the University of Freiburg, in 2007. His current research interests include text and data mining in research information systems and (Arabic) natural language processing (NLP).

Dennis Wehrle studied computer science at the Albert-Ludwigs-University of Freiburg from 2004-2009. Since 2009 Dennis Wehrle is researcher at the professorship in Communication Systems of the Institute for Computer Science at Freiburg. In the beginning he was working at the field of telecommunication such as mobile communication (GSM) as well as VoIP and ISDN. From 2011-2013 he was working on distributed storage systems within the bwLSDF project. His current research is in the area of research data management. For this he was working on the bwFDM-Communities Project (2014/2015; developing recommendations in order to meet the increasing demands of research data management of researchers at the Universities of the state of Baden-Wuerttemberg), IQF-Project "Landesweit koordinierte Strukturen für Nachweis und effiziente Nachnutzung von Forschungsdaten" (2014-2016; developing of location independent and interdisciplinary research data management concepts) as well as on the Project bwDataDiss (since 2014; developing concepts and infrastructure in order to archive and characterize research data from dissertations).