

**CRC806-Database:
A semantic e-Science infrastructure for an
interdisciplinary research centre**

Inaugural-Dissertation

zur
Erlangung des Doktorgrades

der Mathematisch-Naturwissenschaftlichen Fakultät

der Universität zu Köln

vorgelegt von

Christian Willmes

aus Köln

Köln 2016

Berichterstatter:
(Gutachter)

Prof. Dr. Georg Bareth

Prof. Dr. Ulrich Lang

Tag der mündlichen Prüfung:

24. Mai 2016

Contents

Abstract	7
Zusammenfassung	9
Abbreviations & Acronyms	11
I Introduction	15
1 Aim, subject and outline of the thesis	17
1.1 Geography, the science of integration	19
1.2 Collaborative Research Centre 806	20
1.3 Research objectives	23
1.4 Outline of the thesis	24
2 Related work and theoretical background	27
2.1 On data, information, and knowledge	27
2.2 Data modeling, knowledge representation, and metadata	33
2.3 On Networks, the Internet and interoperability	50
2.4 Research Data Management	56
2.5 e-Science	66
2.6 Geographical Information Systems	74
II Design, Methods and Technology	81
3 Demands and Design	83
3.1 Demands for e-Science infrastructure	83
3.2 Research Data Management	89
3.3 System architecture	94
3.4 System integration and interoperability	98
3.5 Copyright and licensing	102
4 Development Methods	107
4.1 Web development	107
4.2 System administration	110
4.3 Top-down vs. bottom-up development approach	111
4.4 Data model development	113
4.5 Prototyping	114
4.6 Linked Data	116

5	Technology	119
5.1	Server infrastructure and technology119
5.2	Software infrastructure and technology122
III	Implementation	137
6	CRC806-RDM Infrastructure	139
6.1	Data catalog142
6.2	Publications157
6.3	Members directory163
6.4	News & Blog165
6.5	Integrated Search166
6.6	Continuous integration and testing167
7	Spatial Data Infrastructure	169
7.1	GeoNode170
7.2	MapServer and MapProxy175
7.3	Typo3 Extension178
8	Knowledge base	181
8.1	Data model development181
8.2	Form based data entry188
8.3	Data import189
8.4	Data queries and display193
8.5	Data export196
IV	Results	201
9	CRC806-RDM	203
9.1	Frontpage, news and blog204
9.2	Data catalog206
9.3	Publications DB211
9.4	Members directory214
9.5	Integrated search215
9.6	Data metrics and visitor statistics216
9.7	API endpoints219
9.8	Administration console220
10	CRC806-SDI	223
10.1	Maps223
10.2	OWS Interfaces228
10.3	Backend and data management231
11	CRC806-KB	233
11.1	Paleoenvironment GIS data collection233
11.2	Domain Knowledgebases238
11.3	Integration242

V	Synthesis	245
12	Discussion	247
12.1	The CRC806-Database semantic e-Science infrastructure	247
12.2	Discussion and evaluation of the demands	253
12.3	Licensing and data policy	259
12.4	CRC 806 Repository lifecycle	260
12.5	Open Science	262
12.6	Data metrics	270
12.7	Cooperation with related Infrastructures	272
13	Conclusion	275
13.1	Development and adaption of the CRC806-Database	275
13.2	Data sharing and data reuse	276
13.3	Open Science	278
13.4	Semantic e-Science	279
14	Outlook	281
14.1	Data preservation	282
14.2	Post-Project phase and availability of the systems	282
VI	References	285
	Bibliography	287
	Tables	303
	Figures	304
	Listings	307
	Danke	307
	Erklärung	310

Abstract

Well designed information infrastructure improves the conduct of research, and can connect researchers and projects across disciplines to facilitate collaboration. The topic of this thesis is the design and development of an information infrastructure for a large interdisciplinary research project, the DFG-funded Collaborative Research Centre 806 (CRC 806).

Under the name *CRC806-Database* the presented infrastructure was developed in the frame of the subproject "Z2: Data Management and Data Services", a so-called INF project, which is responsible for the research data management within a DFG funded CRC.

During the design, development and implementation of the CRC806-Database, the complex requirements for sound data management in the context of a large interdisciplinary research project were considered theoretically, as well as practically during the implementation. The presented infrastructure design is mainly based on the requirements for research data management in CRC's, that is mainly the secure storage of primary research data for at least ten years, as well as on the further recommendations, that are about support and improvement of research and facilitation of Web-based collaboration, for information infrastructure by the DFG.

The CRC806-Database semantic e-Science infrastructure consists of three main components, i.) the CRC806-RDM component that implements the research data management, including a data catalog and a publication database, ii.) the CRC806-SDI component that provides a Spatial Data Infrastructure (SDI) for Web-based management of spatial data, and additionally, iii.) the CRC806-KB component that implements a collaborative virtual research environment and knowledgebase.

From a technical perspective, the infrastructure is based on the application of existing Open Source Software (OSS) solutions, that were customized to adapt to the specific requirements were necessary. The main OSS products that were applied for the development of the CRC806-Database are; Typo3, CKAN, GeoNode and Semantic MediaWiki. As integrative technical and theoretical basis of the infrastructure, the concept of *Semantic e-Science* was implemented. The term e-Science refers to a scientific paradigm that describes computationally intensive science carried out in networked environments. The prefix "Semantic" extends this concept with the application of Semantic Web technologies. A further applied conceptual basis for the development of CRC806-Database, is known under the name "Open Science", that includes the concepts of "Open Access", "Open Data" and "Open Methodology". These concepts have been implemented for the CRC806-Database semantic e-Science infrastructure, as described in the course of this thesis.

Zusammenfassung

Die vorliegende Dissertation behandelt die Konzeption und Entwicklung einer Informationsinfrastruktur für ein großes interdisziplinäres Forschungsprojekt, den DFG geförderten Sonderforschungsbereich 806 (SFB 806). Unter dem Namen CRC806-Database wurde die vorgestellte Infrastruktur im Rahmen des Teilprojekt "Z2: Data Management and Data Services", einem so genannten INF Projekt entwickelt, das für das Forschungsdatenmanagement innerhalb eines SFB zuständig ist.

Während der Konzeption, Entwicklung und Umsetzung der Infrastruktur wurde auf die komplexen Anforderungen für das Datenmanagement im Rahmen eines interdisziplinären Forschungsprojekt sowohl theoretisch, als dann auch mit der praktischen Umsetzung eingegangen. Ziel der Arbeit ist die Beschreibung und Dokumentation aller Komponenten der Infrastruktur, inklusive ihrer Entwicklung und den dieser Entwicklung zugrunde liegenden Forschungen. Die Konzeption der Infrastruktur basiert auf den Anforderungen der DFG für das Forschungsdatenmanagement in SFB's, sowie den erweiterten Empfehlungen zur Umsetzung der Forschungsinfrastruktur. Des Weiteren wurden die speziellen Anforderungen der im Projekt beteiligten wissenschaftlichen Disziplinen, als auch auf den Wünschen der am SFB beteiligten Teilprojekte und Wissenschaftler berücksichtigt. Die CRC806-Database besteht aus drei Hauptkomponenten, i.) der CRC806-RDM Komponente, die das Forschungsdatenmanagement, inklusive Datenkatalog und Publikationsdatenbank implementiert, ii.) der CRC806-SDI Komponente, die eine Spatial Data Infrastructure (SDI) zur Web-basierten Verwaltung von Geodaten zur Verfügung stellt, und schließlich iii.) die CRC806-KB Komponente, die eine kollaborative virtuelle Forschungsumgebung umsetzt. Aus technischer Perspektive, wurden für die Entwicklung der Anwendungen existierende Open Source Software Lösungen den spezifischen Anforderung, z.B. durch die Entwicklung von eigenen Komponenten angepasst und eingesetzt. Die wichtigsten eingesetzten Open Source Software Produkte sind, Typo3, CKAN, GeoNode und Semantic Mediawiki.

Als integratives technisches und theoretisches Konzept der Infrastruktur wurde *Semantic e-Science* umgesetzt. Unter e-Science ("enhanced Science") versteht man ein Wissenschaftliches Paradigma, dass kollaborative Anwendungen auf der Basis von digitalen Infrastrukturen umfasst. Durch den Präfix "Semantic" wird dieses Konzept um die Anwendung von Semantic Web Technologien zur Umsetzung der digitalen Infrastruktur erweitert. Eine weitere wichtige konzeptionelle Grundlage für die Entwicklung der CRC806-Database, sind die unter dem Namen "Open Science" bekannten Konzepte zu "Open Access", "Open Data" und "Open Methodology". Diese Konzepte wurden, soweit möglich umgesetzt und angewendet, wie im Verlauf dieser Arbeit beschrieben.

Abbreviations & Acronyms

AI	Artificial Intelligence	DEM	Digital Elevation Model
ALM	Application Lifecycle Management	DFG	German Research Foundation
AFS	Andrew File System	DIN	Deutsches Institut für Normung
AMH	Anatomically Modern Human	DL	Digital Library
AP	Application Profile	DMP	Data Management Plan
API	Application Programming Interface	DNS	Domain Name Service
BDD	Behaviour Driven Development	DOI	Digital Object Identifier
BSCW	Basic Support for Cooperative Work	DRY	Don't repeat yourself
CAA	Computer Applications & Quantitative Methods in Archaeology	EU	European Union
CC	Creative Commons	FOSS4G	Free and Open Source Software for Geospatial
CERN	Conseil Européen pour la Recherche Nucléaire	FTP	File Transfer Protocol
CI	CyberInfrastructure	GFZ	Deutsches GeoForschungsZentrum
CKAN	Comprehensive Knowledge Archive Network	GI	Geographic Information
CLI	Command Line Interpreter	GIS	Geographical Information System
CMS	Content Management System	GIScience	Geographical Information Science
CNRI	Corporation for National Research Initiatives	GML	Geography Markup Language
CRC	Collaborative Research Centre	GPS	Global Positioning System
CSS	Cascading Style Sheets	GRDDL	Gleaning Resource Descriptions from Dialects of Languages
CSW	Catalog Service for the Web	HTML	HyperText Markup Language
CSV	Character-Separated Values	HTTP	Hyper Text Transfer Protocol
DB	Data Base	IBM	International Business Machines Corporation
DBMS	Data Base Management System	ICANN	Internet Corporation for Assigned Names and Numbers
DCAT	Data Catalog Vocabulary	IETF	Internet Engineering Task Force
DCC	Digital Curation Centre	INSPIRE	Infrastructure for Spatial Information in the European Community
DCMI	Dublin Core Metadata Initiative	IoT	Internet of Things
DDD	Domain Driven Design	IP	Internet Protocol

ABBREVIATIONS & ACRONYMS

IRTG	Integrated Research Training Group	OSM	Open Street Map
IS	Information System	OSS	Open Source Software
ISBN	International Standard Book Number	OWL	Ontology Web Language
ISDM	Information System Development Methodology	OWS	OpenGIS Web Services
ISO	International Standardisation Organization	PDF	Portable Document Format
ISSN	International Standard Serial Number	PMIP	Paleoclimate Modelling Intercomparison Project
IT	Information Technology	PURL	Persistent Uniform Resource Locator
JRC	Joint Research Centre	RAID	Redundant Array of Independent Disks
JSON	JavaScript Object Notation	REST	REpresentational State Transfer
KB	Knowledge Base	RDBMS	Relational database management system
KM	Knowledge Management	RDF	Resource Description Framework
KML	Keyhole Markup Language	RDFa	RDF in Attributes
KMS	Knowledge Managment System	RDFS	RDF Schema
KR	Knowledge Representation	RDL	Research Data Lifecycle
LBS	Location Based Services	RDM	Research Data Management
LAMP	Linux, Apache, MySQL, PHP server	RHEL	RedHat Enterprise Linux
LGM	Last Glacial Maximum	RIF	Rule Interchange Format
MDE	Model-Driven Engineering	RLC	Repository Life Cycle
MIS	Marine Isotope Stage	RPC	Remote Procedure Call
MOOC	Massive Open Online Course	RRZK	Regionales RechenZentrum Köln
MVC	Model View Controller	RSS	Rich Site Summary
MW	MediaWiki	SAN	Storage Area Network
NGO	Non-Governmental Organization	SDD	Schema-Driven Development
NISO	National Information Standards Organization	SDI	Spatial Data Infrastructure
OA	Open Access	SF	Semantic Forms
OCLC	Online Computer Library Center	SFTP	SSH File Transfer Protocol
OGC	Open Geospatial Consortium	SKOS	Simple Knowledge Organization System
OKFN	Open Knowledge Foundation	SLD	Styled Layer Descriptor
OOP	Object Oriented Programming	SMTP	Simple Mail Transfer Protocol
OS	Operating System	SMW	Semantic MediaWiki
OSGeo	Open Source Geospatial Foundation	SOA	Service Oriented Architecture
OSI	Open Source Initiative	SPARQL	SPARQL Protocol And RDF Query Language
		SRF	Semantic Result Formats
		SSH	Secure Shell

ABBREVIATIONS & ACRONYMS

SWT	Semantic Web Technology	VCS	Version Control System
SQL	Structured Query Language	VM	Virtual Machine
TER	Typo3 Extension Repository	VPN	Virtual Private Network
TCP	Transmission Control Protocol	VRE	Virtual Research Environment
TMS	Tiled Map Service	W3C	World Wide Web Consortium
UDP	User Datagram Protocol	WCS	Web Coverage Service
UI	User Interface	WebGIS	Web-based GIS
UKLAN	Universität zu Köln LAN	WFP	World Food Programme
UN-GGIM	United Nations Committee of Experts on Global Geospatial Information Management	WFS	Web Feature Service
UoC	University of Cologne	WHO	World Health Organization
UoD	Universe of Discourse	WMS	Web Map Service
URI	Uniform Resource Identifier	WMTS	Web Map Tile Service
URL	Uniform Resource Locator	WWW	World Wide Web
URM	User Rights Management	XML	eXtensible Markup Language
		YAML	Yet Another Markup Language

Part I.

Introduction

1. Aim, subject and outline of the thesis

Semantic e-Science infrastructure for an interdisciplinary research project about *Culture-Environment Interaction and Human Mobility in the Late Quaternary*¹ is the topic of this study. It includes the design, development, implementation as well as the management, organization, administration and maintenance of a considerably complex Information Technology (IT) infrastructure. The here presented *CRC806-Database* is an innovative semantic e-Science infrastructure implementation. As described and discussed in detail in the following of this thesis, it offers a wide range of solutions for an even wider range of demands and requirements. These include technological, policy, and financial constraints and possibilities, as well as discipline specific considerations. The research approach, applied for this study, combines basic Computer Science and software development methodologies and technologies with concepts of Geoinformatics – including GIS and WebGIS –, GIScience and Geography – including models of space and time –, as well as basic concepts of the further Collaborative Research Centre (CRC) 806 participating disciplines of Archaeology, Geosciences (Geology, Meteorology) and Anthropology.

As the main theoretical background for the implementation of the *CRC806-Database* is the concept of *Semantic e-Science* (Hey et al. 2005; Fox et al. 2009; Ma et al. 2015). The term *e-Science*, first defined informally as "where IT meets scientists", was coined by Jim Gary, an US American computer scientist, in the late 1990'ies as Hey et al. (2009) described it in the book *The Fourth Paradigm* (Hey et al. 2009). In this book, which is a scholarly collection of chapters about the application of computational methods for data intensive science (i.e. e-Science), Hey et al. (2009) founded the basis for this new field of research, or maybe even a new research discipline.

The *fourth paradigm* is understood as the next or current scientific paradigm in the succession from the *first paradigm*, the empirical sciences, describing natural phenomena from observations, which has been common to humans for at least the last thousand years. The *second paradigm*, that is understood as the theoretical branch of science in which models, mathematics, and formalizations revolutionized science, at least since Newton. The *third paradigm* is known as the computational branch that facilitates simulations of complex phenomena, which started during World War II. Now, in the early 21st century, we are at the beginning of the *fourth paradigm*, understood as data-intensive or data-exploring science. This work aims to facilitate technology for the application and conduct of the *fourth paradigm* within the CRC 806 project, and for the wider community.

The term *semantic* before e-Science in *Semantic e-Science* stems from the application of

¹<http://www.sfb806.de>, accessed: 2016-03-26.

Semantic Web Technology (SWT) in the context of e-Science. Semantic e-Science concerns state-of-the-art technologies in knowledge representation, data interoperability, vocabulary and data services, and data processing (Fox et al. 2009; Ma et al. 2015). It is the next evolutionary step, to improve data interoperability and, thus, sharing and reuse of data. Like for the architecture of the World Wide Web (WWW), the *link* is the basic and most important concept of this work. Be it linking of Open Source Software technologies, or data, or infrastructures. This concept of the *link* will be one central theme of this study. This approach is also advocated by renowned institutions, for example by the United Nations Committee of Experts on Global Geospatial Information Management (UN-GGIM):

Most significant changes in the geospatial realm will come not through a single technology, but rather from *linking* multiple technologies and policies (Norris 2015).

In this regard, the preferred application and implementation of networks instead of hierarchical models, support the *emergence* of new or additional information and knowledge, from its intrinsic structure of linked annotations. This feature is the main argument for the application of SWT in context of the CRC806-Database.

The practical aim of this study is the creation of a Semantic e-Science infrastructure for the interdisciplinary research project CRC 806. That allows the management, including storage, organization and publication of research data, and to support and facilitate collaborative research within the project. This includes the design, architecture, implementation, and ongoing improvement of systems to support these goals, and also to communicate these with the scientific community and the interested public. The need for infrastructures like the here presented CRC806-Database for the management of data, to enable data reuse by the scientific community, for education, and for the wider public is increasingly acknowledged and appreciated nowadays. Another important aim of this work is to contribute to implement the concepts of *Open Science*. Open Science combines the concepts of *Open Access* and *Open Data* with approaches to *Open Peer-review* and *Open Methodology* with the application of *Open Source Software*. These concepts are applied and offered were possible throughout the CRC806-Database infrastructure. As further will be shown in the course of this study, there exists an enormous wealth of information in traditionally published research, in literature, in data collections published on the Internet, and in almost all domains of interest. The domains of concern to the CRC 806 are no exception to this. Consequently, the problem is not primarily the creation of new data; rather, it is to locate relevant existing data, integrate those with existing and new research questions and, thus, reuse the data for the creation of additional information, meaning, and — most important — knowledge. The work delivered for this thesis aims to contribute to a better solution for locating and reusing data to contribute to the solving of this matter.

However, a major problem for data sharing and publication is not of technical nature. Rather, it is of cultural, or sociological or even of political nature. Because of these heterogeneous obstacles and problems, it is still common behavior not to publish the primary data of research analyses and its results (Nelson 2009). An additional main problem is the lack of incentive

for the publication of data. For many scientists, the publication of the raw research data has more disadvantages than advantages. Having the data in the public means that there are more possibilities to find errors in the applied methods and computations, thus exposing a potentially vulnerable surface that is simply prevented by not granting access to the data, where it is not asked for in the first place. By providing and advertising the benefits of the concepts, methods and technology known under the term *Open Science*, it is another major aim of this work to help improve the situation according to data sharing and data reuse.

It is commonly agreed that these problems need to be addressed and solved in the near future for the common good and to guarantee progress. Thus, the development of an Research Data Management (RDM) infrastructure implementing a list of basic functionality is demanded by the funding agency, the German Research Foundation (DFG), for Collaborative Research Centers like the CRC 806. The RDM infrastructure facilitates the storage, archiving, organization, and publication of research data created in the CRC 806, and implements these basic functionalities. Additionally to the design and implementation of the RDM infrastructure, an Spatial Data Infrastructure (SDI) for managing geospatial data, as well as a Knowledge Base (KB) system consisting of a collaborative Research Data Base supporting the collection, sharing, and analysis of the actual research data of the project, was implemented. The integration of these three applications, including interfaces for the researchers to build data collections and to discover what data are available, is understood as the Semantic e-Science infrastructure.

1.1. Geography, the science of integration

As this thesis is apparently a thesis in Geography, but from the title and a superficial view on the contents, it seems more like a thesis in Information Science or even Computer Science, some considerations about why this thesis is a thesis in Geography are due.

You may ask, "How is Building e-Science infrastructure for an interdisciplinary research project Geography?". Basically this work creates an IT infrastructure for the handling of data, information, and knowledge about past environments and past human cultures and their migrations, expansion, and dispersal over time. Creating a data model, and an according infrastructure, is nothing else than describing some domain — in technical rigor, of course — and thus describing a subset of the earth. The earth is the subject of the discipline of Geography, as de Geer (1923) defined the discipline more than 90 years ago: "*the science and art of describing planet earth*". Additionally, Geography is a most Newtonian discipline, rigidly framed in space and time (Goodchild 2013: 1072). And this rigid Newtonian frame of space and time has a most important role in the design, implementation and application of the here presented CRC806-Database infrastructure, as explained in detail in the following of this work. Furthermore, this research can be considered part of Geographical Information Science (GIScience), and Geoinformatics, which is how the discipline is usually called in Germany, because the handling of spatio-temporal geographic data and the application of Geographical Information System (GIS) concepts is of most importance at the core of this study.

Geography also stakes a claim to the title of “integrating discipline” or an “integrated science,” because of its concern with both social and environmental phenomena and processes (Goodchild 2013; Gebhardt et al. 2011). If one takes into account that the research of **CRC 806!** (**CRC 806!**) also has both aspects — the (paleo) environmental and the anthropological (social) phenomena, as well as processes of migration — the research in the project and thus the infrastructure to support this endeavor is of a geographic nature. Also, because the problem of integrating data from quite heterogeneous sources is a key topic of this thesis, the notion of Goodchild (2013: 1073) that “geography can be seen as the science of integration rather than integrating sciences”, and the assertion that “[...] it seems clear that the most useful role that geography and GIScience can play is in exploring a science of integration,” (Goodchild 2013: 1076) supports the claim, that the research presented is geography.

Additionally, the research carried out in this dissertation is essentially about providing methods and technology for describing planet earth better and thus contributes to the advances of geography and GIScience.

Integration requires the ability to represent the variation of context, in the form of structured geographic databases; to represent processes in the algorithms of software; to couple these representations using identical discretization of space and time; and to analyze the impacts of uncertainties in the representations. None of these forms of generic expertise and knowledge are traditionally recognized as elements of either economics or ecology, but they are all fundamental to geography and GIScience (Goodchild 2013: 1073).

And finally, in times of the fourth paradigm all research disciplines are subject of a digital transformation in terms of how research is conducted. Finding out what kind of infrastructure and technology answers certain demands and questions of a given scientific discipline, is basic research for this discipline.

1.2. Collaborative Research Centre 806

The overall framework setting the constraints and the domain of this study is the CRC 806, a large DFG-funded, interdisciplinary and inter-institutional research project. From the universities of Cologne, Bonn, and Aachen, about 80 researchers from the disciplines of Geosciences and Geography, Archaeology, and Anthropology are working together. DFG Collaborative Research Centers are funded in four-year phases, which are evaluated after the third year of each phase for a decision about further funding. A CRC can have up to three phases, which results in a maximum runtime of 12 years.

The CRC 806 is designed to capture the complex nature of chronology, regional structure, climatic, environmental, and socio-cultural contexts of major intercontinental and transcontinental events of dispersal of Anatomically Modern Human (AMH) from Africa to Western Eurasia, and particularly to Europe (Richter et al. 2012a). Furthermore, the project concentrates on the time

span between the dispersal of AMH from Africa and the permanent establishment of man in Central Europe (Richter et al. 2012b; Schuck et al. 2009). This time span covers the last 190,000 years, including the last two glacials, from Marine Isotope Stage (MIS) 6 during the according inter-glacial period of MIS 5 to MIS 2 until the Holocene MIS 1 (Haidle et al. 2012; Schuck et al. 2009).

Geographically, the region of interest is North Africa and Western Eurasia (the Levante region and Europe), with a western corridor of dispersal along the southern Mediterranean coast over the Strait of Gibraltar or the Lampedus route from today's Tunisia to today's Italy, and an eastern route along the Levant region into the Balkans or even east of the Black Sea through the Caucasus (see Fig. 1.1).

1.2.1. Research Scope of CRC 806

The research within the CRC 806 is mainly based on the "Out of Africa II" Theory (Mellars et al. 1989; Stringer et al. 1994; Richter 1996), which assumes that the *Homo Sapiens Sapiens*, also known as AMH, originated from East Africa around 190,000 years ago (Richter et al. 2012a) even though, it shall not be concealed, that this theory is highly debated within the paleoanthropological community (Richter 1996; Templeton 2002; Dennell et al. 2005; Richter et al. 2012b).

The CRC 806 focuses on three major research themes:

1. The climatic, environmental, and cultural context,
2. Secondary occurrences of expansion and retreat,
3. Population changes, mobility, and migration in coupled cultural and environmental systems.

The first theme tests the eastern and western corridors of dispersal from East Africa to Europe (see Fig. 1.1). The research and excavation sites along these corridors gather data to help the detection of climatic and environmental history in order to discover the impact on the dispersal of AMH to Europe.

The second theme looks into secondary occurrences of expansion and retreat of AMH, induced by environmental or cultural changes, which is about occupation, extinction, and re-occupation of focus areas by AMH over time.

The third theme concerns population changes, mobility and migration in cultural-environment systems. Particular interest is paid to the impact of human agency on the environment and internal mobility among sedentary prehistoric societies (Schuck et al. 2013: 13).

1.2.2. Structure of the CRC 806

The CRC 806 consists of 21 research projects organized in seven research clusters in its second funding phase (2013-2017). Those projects are led by 30 Principal Investigators from the Geosciences and humanities at the Universities of Cologne, Bonn and the RWTH Aachen University (Schuck et al. 2013: 11). The 21 research programmes are organized in four regional clusters



Figure 1.1.: Our Way. Source: (Richter et al. 2012a).

(A, B, C, D, see Fig. 1.2), two research clusters (E, F) dedicated to supra-regional questions and one cluster (Z) providing the centralized tasks and services of the CRC.

Project Z2: Data Management and Data Services

Within the context of the centralized tasks and services cluster Z, the Z2 project facilitates the research within the CRC 806 by providing data management infrastructure and services (Bareth et al. 2013). In the CRC 806 second-phase funding proposal project Z2 is introduced as follows:

The main purpose of an Information Infrastructure Project in a Collaborative Research Centre is the management of relevant data collected by the CRC with the aim of enabling systematic and long-term use of such data. Within the CRC 806, project Z2, headed by Georg Bareth and Olaf Bubenzer, is the main and centralized Information Infrastructure Project (Schuck et al. 2013: 24).

Thus the following aims of the project are identified:

1. Long-term archival of research data sets.
2. Collection of relevant data.
3. Enabling systematic use of data.

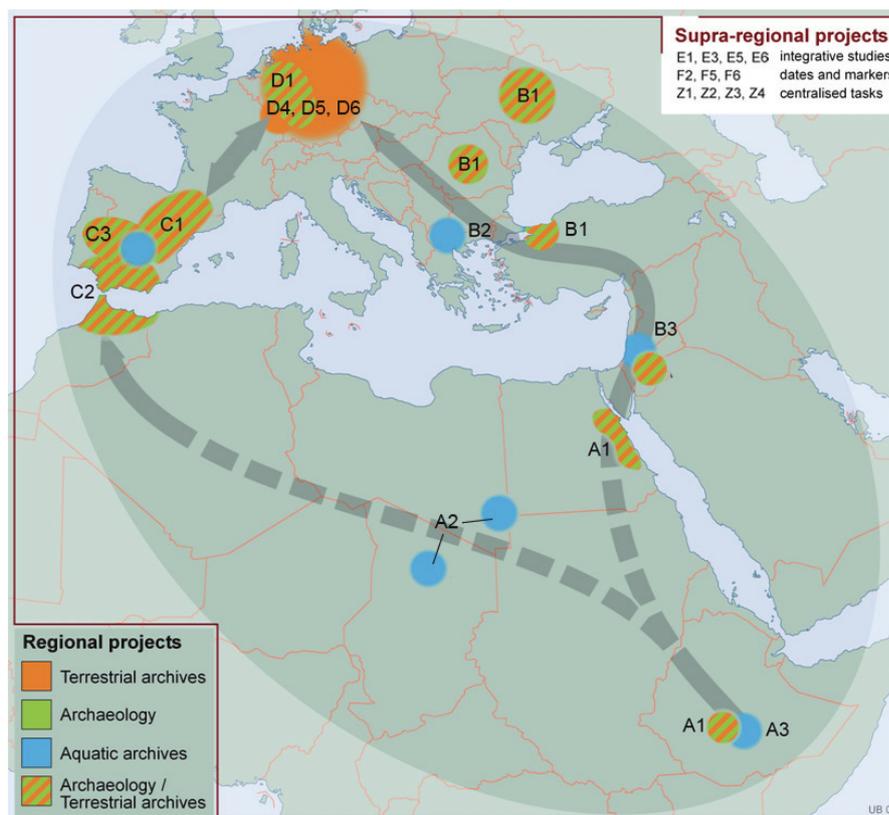


Figure 1.2.: Geographical focus regions of the second CRC 806 funding phase 2013 - 2017 (Schuck et al. 2013: 12).

4. Central information infrastructure of the CRC 806.

To implement these aims, the CRC806-Database² was developed by project Z2, and further described in all detail in the following of this thesis. According to the DFG proposal guideline 60.06, it is expected that a CRC/TRR has a project section focusing on sustainable data storage and management (Bareth et al. 2013). Additionally, Z2 consults and provides GIS- and remote sensing analyses and the acquisition and preparation of different high- and low-resolution satellite data for the CRC 806 projects. Furthermore, using Terrestrial Laser Scanning (TLS), the creation of high-resolution point clouds and surface models for different aims could be carried out (Bareth et al. 2013).

1.3. Research objectives

To begin with a clear statement for the main objective, of improving the situation of data sharing and data reuse, a quote from the UN-GGIM report "Future trends in geospatial information management: the five to ten year vision" (Norris 2015) is cited here:

Our ability to create data is still, on the whole, ahead of our ability to solve complex

²<http://crc806db.uni-koeln.de>

problems by using the data. There remains no doubt that there is a huge amount of value still to be gained from the information contained within the data generated. The growth in the amount of data collected brings with it not only a growing requirement to be able to find the right information at the right time, but also challenges of how to store, maintain and use the data that is created (Norris 2015).

The quote shares the observation, that data is in general abundantly available, but access and reuse constraints for the data by the scientific community is part of the problem. And the last sentence, about "how to store, maintain and use the data" of the above quote is at the central concern of this work, applied for the demands and requirements of the CRC 806.

In this regard, the research objective is to deliver infrastructure and tools that help to support and facilitate the research of the CRC 806. This includes the four goals, as identified in the CRC 806 project proposal of Z2 (Bareth et al. 2013):

- Long-term archiving of research data,
- Collection of relevant data,
- Enabling systematic use of data, and
- Central information infrastructure of the CRC 806.

A further objective of this study is to improve the handling of spatio-temporal research data. Thus, combining models and knowledge from the geography and Archaeology domains that are the core disciplines of the framing research project, are applied for designing and implementing the presented infrastructure. Consequently, it is also an important focus of this thesis to develop tools and data models, that improve the integration of data and research across the different disciplines involved in the CRC 806.

Another important research objective of this thesis is to contribute to the question of how innovative data management and e-Science infrastructure such as collaborative KB applications can be implemented to facilitate and improve the conduct of research in a large interdisciplinary research project.

1.4. Outline of the thesis

This thesis is structured in five parts and 14 chapters. The first part (Introduction) has two chapters, the first chapter, "Aim and subject of Study," explains the basic aims, objectives, and ideas of this thesis and introduces the overall academic and institutional and setting of the research project.

In the remainder of this thesis, the theoretical background and related work will be discussed first in chapter 2 "Related work and theoretical background." The related work covers a wide ground: from the basics about data (2.1.1, p. 29), information (2.1.2, p. 29), and knowledge (2.1.3, p. 31), to the essentials of data modelling (2.2, p. 33), metadata (2.2.1, p. 34), standards (2.2.1), and interoperability (2.3). There follows an introduction to Semantic Web concepts and

SWT (2.2.4, p. 42), including the concept of Linked Data (2.2.4, p. 45). This leads to an overview of the basics of networks (2.3.1, p. 50) and the Internet (2.3.2, p. 54). Based on these grounds, the central concepts of RDM (2.4, p. 56) and e-Science (2.5, p. 66) are introduced. The related work is then concluded with an introduction to the concepts of web based spatial data handling and GIS in section 2.6.

Part II, "Design, Methods and Technology," is divided into three chapters. Chapter 3 "Demands and Design," gathers the demands that build the basis to implement the presented infrastructure. First, overall demands for e-Science Infrastructure are outlined (3.1), including funders demands (3.1.2), demands from the project partners (3.1.3), and basic requirements for building a KB (3.1.4). Then, the design of the applied RDM approach is given in section 3.2, followed by the technological overall system architecture of the CRC806-Database infrastructure in section 3.3. Section 3.4 provides a layout of how the subsystems of the presented infrastructure are integrated, and which technologies are provided to enable interoperability for access from third party infrastructures. This chapter is concluded with the copyright and licensing policy implemented in the presented system (3.5). Chapter 4, "Development Methods," explains the methods applied to implement the CRC806-Database infrastructure. Because the CRC806-Database is a web-based system, the first method explained is "Web development" (4.1). The second most important technology for building and maintaining a web based infrastructure is condensed in the term "System administration", as explained in section 4.2. The methods applied to develop the data models applied in the system are described in section 4.4, followed by an overview, in section 4.4.1, of the most important datasets used to build the basic data model and KB. In chapter 5, the technology stack applied for creating the infrastructure components is introduced. The chapter is divided into a section about the physical server infrastructure (5.1) and a section about the software infrastructure (5.2).

Part III "Implementation," concerns the mainly technical implementation details. This part is structured in three chapters, each about one of the three main building blocks of the CRC806-Database. Chapter 6 describes the implementation of the CRC806-RDM infrastructure, containing details about the data catalog (6.1), the publications database (6.2), the user management (6.3), the "News & Blogs" application (6.4), the integrated search and browse interface (6.5), and the Application Programming Interface (API) of the RDM infrastructure. In chapter 7, the implementation of the CRC806-SDI is explained. Section 7.1 describes the setup of the GeoNode backend. The setup of the additional SDI components of MapServer and MapProxy are described in section 7.2. The integration of the SDI components into the Typo3 based web application is described in section 7.3. The implementation of the Semantic MediaWiki (SMW) based CRC806-KB is described in section 8. Section 8.1 describes how the data model and data schema are implemented in SMW, followed by the description of how the automated data import into the KB is facilitated (see section 8.3). The implementation of web based User Interface (UI) to enter data manually is described in section 8.2.

Part IV "Results," presents the result of this thesis, that is the overall CRC806-Database semantic e-Science infrastructure application. At first the result and features of the CRC806-RDM

infrastructure are presented in chapter 9. In chapter 10, the CRC806-SDI application is presented, including its interfaces and features for geospatial data management. The resulting CRC806-KB system is presented in chapter 11.

Part V "Synthesis" is again structured in three chapters, "Discussion of the Results" in chapter 12, the "Conclusions" drawn from this work in chapter 13, and an outlook for the further development and future of the CRC806-Database in chapter 14.

2. Related work and theoretical background

This chapter will give an introduction to the two basic concepts, of RDM, in particular to knowledge and data management and to the field of *Semantic e-Science*, fundamental to this work. To introduce these more general and complex topics, first the basics of data modelling, information technology and web technology, are introduced. Additionally, an overview of and introduction to GIS and SDI concepts are given because the capabilities in spatial data handling are a key feature of the presented infrastructure.

First, we take a look at how data, information and knowledge are defined and how they relate to each other in section 2.1. The emphasis of this section is on the investigation of the nature of data and aspects of emergence of information and knowledge from linking and contextualisation of data. This is followed by an introduction to data modeling, metadata, knowledge representation, and knowledge management in section 2.2. Because this thesis is about web-based infrastructure to facilitate research, basic concepts of networks, the Internet, the WWW and interoperability, including an introduction to semantic web technology, are given in section 2.3. After introducing these basic concepts, the field of RDM is detailed in section 2.4, followed by an introduction to e-Science and, in particular, Semantic e-Science and the consequences of the fourth paradigm, in section 2.5. The related work and theoretical background concludes with an overview of GIS and SDI technologies in section 2.6.

2.1. On data, information, and knowledge

Because this thesis is basically about handling *data*, *information* and *knowledge*, some thoughts about and definitions of these terms are given in this section. Data, information and knowledge are closely related terms, but each has its own role in relation to the other. This relation is formalized in a concept called the *Knowledge Pyramid* (Ackoff 1989) (see figure 2.1), to represent the structural and functional relationship between data, information, and knowledge. Some models of the *Knowledge Pyramid* are extended to include the concept of wisdom above the concept of data, in this case it is called the *DIKW Pyramid*, for *Data*, *Information*, *Knowledge*, and *Wisdom*.

The inference from figure 2.1 is that data begets information begets knowledge begets wisdom. An additional inference is that there is more data than information, more information than knowledge, and more knowledge than wisdom (Jennex 2009). A different formalization of that same concept would be to express these relations in summations, of the form:



Figure 2.1.: The wisdom hierarchy (Rowley 2007), or the DIKW pyramid (Jennex 2009; Frické 2009) based on the Knowledge Pyramid (Ackoff 1989).

$$I = \sum(D)$$

$$K = \sum(I) = \sum \sum(D)$$

$$W = \sum(K) = \sum \sum(I) = \sum \sum \sum(D)$$

With: W =Wisdom, K =Knowledge, I =Information, D =Data.

One important purpose of the *Knowledge Pyramid* concept is to reflect that the level of abstraction increases from data upwards to wisdom. Thus, the concept puts the relationship between data, information and knowledge into a hierarchical arrangement based on the level of abstraction.

This concept has been part of the common canon for information science for many years, but it is not clear who came up with it first (Wallace 2007). However, the concept is not undisputed. Jennex (2009) criticizes the concept as too basic and fails to represent reality and she presents a revised knowledge pyramid in which *knowledge management*, as an extraction of reality with a focus on organizational learning, is added to the concept. Frické (2009) has a strong critique of the model and even demands that the Knowledge Pyramid should be abandoned in the canon of Information Science because he doubts that it is "a useful and intellectually desirable concept to introduce." In the next paragraph, however, he also admits that "disregarding DIKW would leave an intellectual and theoretical vacuum over the nature of data, information, knowledge, and wisdom" (Frické 2009: 132).

2.1.1. Data

The word "data" is the plural of "datum," the past participle of the Latin term "dare," which means "to give," hence datum means "something given" (Checkland et al. 1998: 86). Data can be described as the lowest level of abstraction or order (Elmasri et al. 2011: 2). A common general definition of data is:

"Basic, discrete, objective facts such as who, what, when, where, about something."
(Jennex 2009)

In a computer science and IT context, data is understood as machine readable and processable digital representations of information. Data is, in this context, always bound to a format specific to the kind of information or the software used to create or collect this data. The following definition has more emphasis on this IT point of view:

"Data is factual information (as measurements or statistics) used as a basis for reasoning, discussion, or calculation." (Longley et al. 2005).

With ease, the following definition by the ancient Greek philosopher Plato emphasizes the purity of data:

"Data has no meaning or value because it is without context and interpretation." (Plato 400 BC), as cited in (Rowley 2007)

According to the many existing definitions of data, Rowley (2007: 171) interestingly asserts, that "[...] these definitions are largely in terms of what data lacks; data lacks meaning or value, is unorganized and unprocessed. They lay the foundations for defining information in terms of data."

To conclude, data are plain facts. When data are processed, organized, structured or presented in a given context so as to make them useful, they are called information. Data in themselves are fairly useless. However, when these data are interpreted and processed to determine their true meaning, they become useful and can be called information. Data is the computer's language. Information is our translation of this language.

2.1.2. Information

The word *information* is derived from Latin *informare*, which means "give form to" (Sveiby 1994). In this sense, information can be understood as *formatted* data and as a representation of reality (Jessup et al. 2005: 7). The way the word information is used can refer to both "facts" in themselves and the transmission of the facts (Sveiby 1994). The anthropologist and ecologist Gregory Bateson, defined the term *information* in a brilliantly simple sentence as:

"Information is a difference that makes a difference." (Bateson 1987: 123)

According to that definition by Bateson (1987), the concept of *data* can be interpreted as the first difference in that definition of *information*. Emphasis here is on the difference in the semantic sense, by adding meaning to data, for example, through formatting, annotating, or contextualisation. This is also expressed in the following wider definition:

"Data that are related to each other through a context such that they provide a useful story, as an example, the linking of who, what, when, where data to describe a specific person at a specific time." (Jennex 2009)

In this definition, the term linking is used to define information. Semantically (meaningfully), purposeful and/or structured linking of data with other data creates information. Jashapara (2005: 16) formulates this same thought as follows: "Meaning in data often occurs through some form of association with experience or relationships with other data." Here, it is worthy to note that *association* and *relationship* are forms of linking.

A crucial point about the difference between information and data is that the human receiver determines whether a message or signal is data or information (Rowley 2007: 172). Here, information is defined in terms of data and is seen to be organized or structured data. The difference between *data* and *information* is elaborated in the following assertion:

"Information is differentiated from data by implying some degree of selection, organization, and preparation for particular purposes – information is data serving some purpose or data that have been given some degree of interpretation." (Longley et al. 2005)

According to Bartelme (2005: 13), we speak of *information* as in the Geographic Information (GI) sense, if we receive an answer to a specific question, that increases our knowledge and helps to reach a goal behind the question. Bartelme (2005: 13) further emphasizes three important aspects of information:

- structural and syntactic (form) aspect,
- semantic (content) aspect,
- and the pragmatic (applied) aspect.

Bartelme (2005) illustrates these aspects in context of a letter. If you receive a letter in a foreign language that you are not educated in, you can understand the structure and syntax (address, sections, greetings, sentences, words, etc.), but you cannot understand the sense of the written text. On the other hand, if you receive a letter in English, where you can understand the content, but the content is irrelevant because it is an advertisement for example, it maybe qualifies as entertainment but not as information in the above-defined sense.

In the context of the knowledge pyramid, information is not regarded in the pure syntactic sense of encoding but in the semantic sense of meaning and truth (Frické 2009: 139). It is to note that this is dependent on the level of abstraction within the model of the knowledge

pyramid itself. Concepts of syntactic informations — such as, binary codes, Huffman trees, or Hamming codes known from computer science — also carry semantic information that is intrinsically linked to the definitions of these syntactic encodings and, through this linking to other data and information, in this sense the encoding definition, is also information in the sense discussed here.

2.1.3. Knowledge

Naturally, the definition of the concept of *knowledge* is much more complex than the definitions of *data* and *information*. It is also important to note, that we speak here of explicit knowledge (know what) that can be clearly formulated and recorded, and not of tacit knowledge (know how), that cannot be recorded since it is part of the human mind (Rowley 2007: 172). Knowledge is codifiable if it can be written down and transferred relatively easily to others. Tacit knowledge is often slow to acquire and much more difficult to transfer (Longley et al. 2005). The following definition gives an understanding of the complexity of the concept of knowledge:

"Knowledge is information that has been culturally understood such that it explains the how and the why about something or provides insight and understanding into something." (Jennex 2009)

Thus, *knowledge* is the understanding, awareness, or familiarity with some information, data, fact, or skill that is acquired through experience or education. Knowledge is also a network of information (Seemann 2014: 18); according to this idea, knowledge consists of information that is linked or related to other information. The information is thus embedded in a net of related and linked information or knowledge; this concept is also known as *context* and is included in the following definition:

"Knowledge is the ability to interpret data in context and thereby gain information."
(Dengel 2012: 5)

This definition also brings the human subject as an interpreter into the game. According to Bartelme (2005: 14), knowledge is the ability to use a number of single informations, to solve a given problem by a combination of these informations. He further emphasizes, that knowledge is also the ability to compare informations along defined criteria and to learn from them. This last aspects sounds very much like pure tacit knowledge, but it is the core area of interest for the Artificial Intelligence (AI) community, that works on building systems to deliver this kind of knowledge in a computerized manner. On the other hand, Boddy et al. (2005)[14] asserts that "the amount of human contribution increases along the continuum from data to information to knowledge." This is also very well expressed in the following definition:

"Knowledge is the synthesis of multiple sources of information over time." (Rowley 2007: 173)

Concluding the pattern that linking entities of lower abstraction yield abstractions of higher order, as seen for linking data with data yields information, and linking information with information yields knowledge seems to make sense and points again to the importance of the concept of linking and networks for Information System (IS) and infrastructures.

Also, maybe even the concept of tacit knowledge could be described as an effect of emergence or synthesis, from connecting and relating informations and networks of information into new knowledge. But it is still noteworthy, that tacit knowledge is also highly subjective and depends on the personality, skills, technique, education, common sense, and experience of the individual.

2.1.4. Wisdom

To complete the introduction of the knowledge pyramid concept, we take a brief look at the concept of wisdom here. Data may present very little processing, in which, for example, observations are directly recorded from a science experiment; information may involve an analysis of data; knowledge may represent an integration of different pieces of information and a conclusion; and wisdom may represent a reflection on this conclusion, in the light of other conclusions and experience (Hider 2012: 2).

"Wisdom is placing knowledge into a framework or nomological net that allows the knowledge to be applied to different and not necessarily intuitive situations." (Jennex 2009)

If knowledge has already a significant part of the interpreters role in its concept, then wisdom is an even more subjective concept. Jashapara (2005) states that "wisdom is a very elusive concept. It perhaps has more to do with human intuition, understanding, interpretation, and actions than with systems." Almost invariably, wisdom is highly individualized rather than being easy to create and share within a group. It is, in a sense, the top level of a hierarchy of decision-making infrastructure (Longley et al. 2005). Longley et al. (2005) further asserts that wisdom cannot be shared formally.

Because this thesis is mainly about developing systems, we do not bother any further with the concept of wisdom in a technical sense here. The author leaves it to the reader to create wisdom from the concepts and systems described here.

In summary of this section on data, information and knowledge, data is the manifestation or discretization (materialization) of measurements or observations. Data is created by assigning observations to a system of assigning values according to the objects and concepts that are central to the problem under study (Boslaugh et al. 2008). Thus, data depicts differences; data is everything that can be expressed in ones and zeros (Seemann 2014: 18). According to this insight, *information* consists of *data*. Now, where does this *data* make a difference according to the definition of Bateson (1987)? It makes a difference in the sense of the second difference in the Bateson (1987) definition, in the concept known as *knowledge*. *Data* is *information* if it makes a

Table 2.1.: A ranking of the support infrastructure for decision making (Longley et al. 2005).

Decision making support infrastructure	Ease of sharing with everyone	GIS example
Wisdom	Impossible	Policies developed and accepted by stakeholders
↑ Knowledge	Difficult, especially tacit knowledge	Personal knowledge about places and issues
↑ Evidence	Often not easy	Results of GIS analysis of several datasets and scenarios
↑ Information	Easy	Contents of a database assembled from raw facts
↑ Data	Easy	Raw (geographic) facts

difference according to existing *knowledge* (Seemann 2014: 18). Thus, data has no information without knowledge, because data is not interoperable or useful without knowledge about what the data represents (Dengel 2012: 4).

Because knowledge is also a subjective experience, we may speak of explicit knowledge, as a relation of formalized informations, in context of this thesis and to the wider field of IT based IS and knowledge management systems.

An important concept that can describe the connections and relations between the sub-concepts of the knowledge pyramid is called *emergence* (Warnke 2011). This means that from the combination and integration of a lower concept, a concept of higher order can emerge. By the combination of data and information, for example through the formulation of an algorithm, knowledge can emerge.

2.2. Data modeling, knowledge representation, and metadata

In this section on data modeling, knowledge representation, and metadata, a fundamental building block of the CRC 806 e-Science infrastructure will be explored. Metadata schemes are a well-established approach for enabling interoperability and reuse of data, but it will be shown that metadata schemes are often too informal because they are often not applied in a machine-interpretable way in direct relation to the resources they describe. Some general investigations of the Knowledge Representation (KR) concept will be made to show what traditional metadata handling lacks in comparison to what is understood by the concepts of KR and Knowledge Management (KM). In essence, the take-home message of this section on data modeling, knowledge representation, and metadata is, that the combination of traditional metadata models with SWT is the key to enable interoperability and data reuse in a more advanced level.

The goal of any IS is to map the reality (or a relevant subset of it) to a — more or less — precise model (Bartelme 2005: 43). In the process of mapping the reality, henceforth called *data modelling*, a part of reality is abstracted and formalized. We call this relevant subset of reality the Universe of Discourse (UoD). A *data model* is the result of this process. As such, a *data*

model, a *schema*, or an *ontology* is an intersubjective map of the UoD. It is of vital importance to be clear about the fact, that the map shall not be confused with the actual territory (Korzybski 1933), e.g. the reality. A more general and generic definition of the terms modeling and model is:

Modeling, in the broadest sense, is the cost-effective use of something in place of something else for some cognitive purpose. It allows us to use something that is simpler, safer, or cheaper than reality instead of reality for some purpose. A model represents reality for the given purpose; the model is an abstraction of reality in the sense that it cannot represent all aspects of reality. This allows us to deal with the world in a simplified manner, avoiding the complexity, danger, and irreversibility of reality (Rothenberg et al. 1989).

The concept of KR facilitates formalisms to capture knowledge, in the sense of formulating informations about relationships, implications, and conventions between informations and data. Some examples of these formalisms are *ontologies*, *semantic nets*, and —to some extent— *information schemata*.

In this section, the basics of data modeling and knowledge representation are introduced. The goal is to define the processes and methods of data, information, and even knowledge handling. In this work, we do not dare to tackle a formalization of wisdom representation. Anyway, for the representation of data we have many mature approaches, formats, and standards. For the representation of information this is a bit more complex; a main concept of the art of information representation is known as metadata annotation of data (see section 2.2.1). However, the representation of knowledge is even more complex than the representation of information. Approaches to achieve this are not yet state-of-the-art; the best known term to describe this level of information technology is the term *Semantic Web*.

2.2.1. Metadata

The term *metadata* stems from two words, *meta* and *data*. The term *meta* is of ancient Greek origin and means *about*, *relating to*, *based on*, or *after* (Miller 2011). The term *data*, as introduced in section 2.1.1, has a latin origin, and refers to *basic facts* (Jennex 2009). Summarized, it is possible to assert:

Metadata relates a collection of facts about other facts.

In this understanding, metadata is the link between a data collection and its application (a user or a system).

As described above, one of the essential pieces of the development of an IS, SDI, CyberInfrastructure (CI), or Digital Library (DL) is the appropriate annotation or documentation of its data and services. The concept of *metadata* is quite *meta*, and can be defined in many valid ways. As a first approach, we start with the following definition from an SDI and Service Oriented Architecture (SOA) perspective:

Metadata is "structured data about data" or "data that describes attributes of a resource," or just "information about data" (Nogueras-Iso et al. 2005: 11).

This concept is not new at all. Metadata has existed for hundreds of years to organize, inventory, and catalog all sorts of data collections (Blumauer et al. 2006: 11). Any organized inventory of books, documents, or any objects could be considered as metadata. Thus, the concept of metadata is at least as old as the first libraries (Lubas et al. 2013: 4). The concept was applied to trade and military inventories, as well as in the domain of state administration and demographics since a long time ago.

Anyway, the concept of *metadata* can be described in summary in two different aspects and in almost complementary ways:

1. As **data describing issues related to the content of data**. We divide this category into two orthogonal dimensions: the formality of the data and the containment of the metadata. In the first dimension, metadata might range from very informal descriptions of documents (like free-text summaries of books) to very formal descriptions (like ontology-based document annotation). In the second dimension, parts of metadata might be internal to the data that is described (like an HTML author tag) while others might be stored completely independently from the document they describe (like a bibliography database that describes the documents it refers to but does not contain them) (Staab et al. 2001).
2. As **data that describes the structure of data**. In our case, you can call this type of metadata "meta metadata" because it describes the structure of metadata. This distinction boils down to an ontology that formally describes the domain of the KM application, possibly including parts of the organization and the information structures (Abecker et al. 1998).

Metadata can help condense and codify knowledge for reuse in other steps of the KM process. It can also help link knowledge items of various degrees of formality together, thus allowing a sliding balance between depth of coding and costs (Staab et al. 2001). Metadata can also express the meaning or, better, the semantics of data (Blumauer et al. 2006: 11), if it is formalized to facilitate this purpose.

Metadata document

If seen more from a library and *digital curation* point of view, that is the more *semiotic* view, data is seen as a *digital message* contained in a *document*, where the aspects of documentation and preservation are primal (Voß 2013).

A document is everything, that can be preserved or represented in order to serve evidence for some purpose (Buckland 1998).

In this aspect, Voß (2013) asserts that it is not the goal to make use of the contents of the document (the data), for example, to interpret or compute to gain new information and insights.

The interest lies here primarily in the published message itself, the document as is. The data content of the document is categorized according to the standard or schema. Thus, the *correct* documentation, by confirming standards and policies, of data documents is the goal.

Also in this context, Nogueras-Iso et al. (2005: 12) identifies the following main obstacles to correct metadata annotation and documentation, that can hinder the correct utilization of metadata:

- Difficulty of cataloging data "correctly,"
- Diversity and heterogeneity of metadata standards,
- Heterogeneity of metadata content (semantic interoperability, meaning values given to a metadata element in two different records are meaning the same concept).

This librarian and digital curation point of view on metadata is both, limiting and useful. Limiting in the sense of direct data reuse or even facilitation of further reuse or based upon research in a KB, because the data and information of the annotated data document is not directly accessible by the KB without actual data integration. This may be useful for creating catalogs of data documents that do not aim to make direct use of the cataloged data, but aim to publish and provide access to these data documents, by achieving concentrated and clear assertions about the content that can be looked up or filtered through queries on the catalogue.

Metadata standards

The concepts of *metadata standard* and the *metadata schema* fit the second definition of the two almost complementary definitions of metadata from above as "meta metadata," describing the structure of data documents. This is also the most widely used connotation of the metadata concept.

To improve interoperability, reduce siloing, and facilitate the reuse of datasets, metadata standards are developed for almost any data domain. A metadata standard is a metadata schema, that has been formally approved by an institution. Institutions that approve and publish such standards are manifold. Some of these institutions are, the World Wide Web Consortium (W3C), International Standardisation Organization (ISO), Open Geospatial Consortium (OGC), Dublin Core Metadata Initiative (DCMI), or Deutsches Institut für Normung (DIN). Many more institutions that deal with the standardization of data vocabularies and metadata schema's exist. It is possible to distinguish two major kinds of metadata standards (Miller 2011):

- i. **data structure** standards, e.g. Dublin Core,
- ii. **controlled vocabularies**, e.g. Thesauri and Gazeteers,

Other studies, for example (Miller 2011) or (Curdt 2014b), distinguish more kinds of metadata standards. They also distinguish *data content standards* (e.g. Anglo-American Cataloging Rules), and *data encoding standards* (e.g. eXtensible Markup Language (XML), JavaScript Object Notation (JSON)). These two kinds of metadata would lead too far for the scope of this study because

we will not get into these two kinds of standards in the context of metadata, but they will be discussed later on in the technology and implementations section of this study. There are also other classifications of metadata standards, for example, by domain (Riley 2009) or by discipline (Ball 2013).

However, there exist many metadata standards for almost any domain; a single standard is often too narrow to express all information as intended by a particular application. Thus, often two or more standards are combined to cover the complete information. If two or more metadata standards are combined to describe information records in an application, the resulting metadata set is referred to as an Application Profile (AP). An example of an AP is the TR32DB Metadata Schema (Curdt 2014a), developed for the TR32DB (Curdt 2014b). AP are in some sense related to the concept of *Linked Data* because they link different vocabularies (metadata standards), also to foster interoperability (Curdt 2014b: 31), but they lack in comparison to Linked Data (as described in section 2.2.4) a formalism for describing the linking itself. Although, the linking in the context of AP has a name, *crosswalk*, the linking is described in an informal way, visually, for example, and/or in free-text form. Whereas as in the Linked Data realm, the linking is formalized through RDF Schema (RDFS) *sameAs* relations, or other custom but formally implemented triples (subject, object, predicate relations).

To ensure, that metadata can be automatically or at least inter-subjectively processed (by machines), some common metadata standard is needed. Such a standard is a set of agreed-on criteria for describing data (Yu 2007: 10). A standard may specify that a record of metadata to describe a dataset shall consist of a set of defined attributes and some additional optional attributes. This kind of standard is called a metadata schema. Metadata can be seen as the description of the schema the data is modeled in. From this perspective, a vocabulary or ontology is just metadata, but in a higher level of organization, or formalization. There are several metadata schemas available for almost all domains of knowledge representation.

Metadata annotation is a type of KR, that is introduced in the following section in detail.

2.2.2. Knowledge representation

Natural language is, of course, the most common Knowledge Representation (KR) technology. But it is too informal to be facilitated by computer systems. Human communication, as a goal for modeling, allows it to play a role in the ongoing collection of human knowledge (Allemang et al. 2011). The levels of communication can be quite sophisticated, including the collection of information used to interpret other information. In this sense, human communication is the fundamental requirement for building a *Semantic Web*. It allows people to contribute to a growing body of knowledge and then draw from it. But communication is not enough. To empower a web of human knowledge, the information in a model needs to be organized in such a way that it can be useful to a wide range of consumers (Allemang et al. 2011: 16). The facilitation by computers is directly dependent on the degree of formalization and, thus, implicit or intrinsic knowledge of the KR method. This aspect is depicted graphically in figure 2.2.

The degree of formalization is relatively low in a document repository that can be simple files

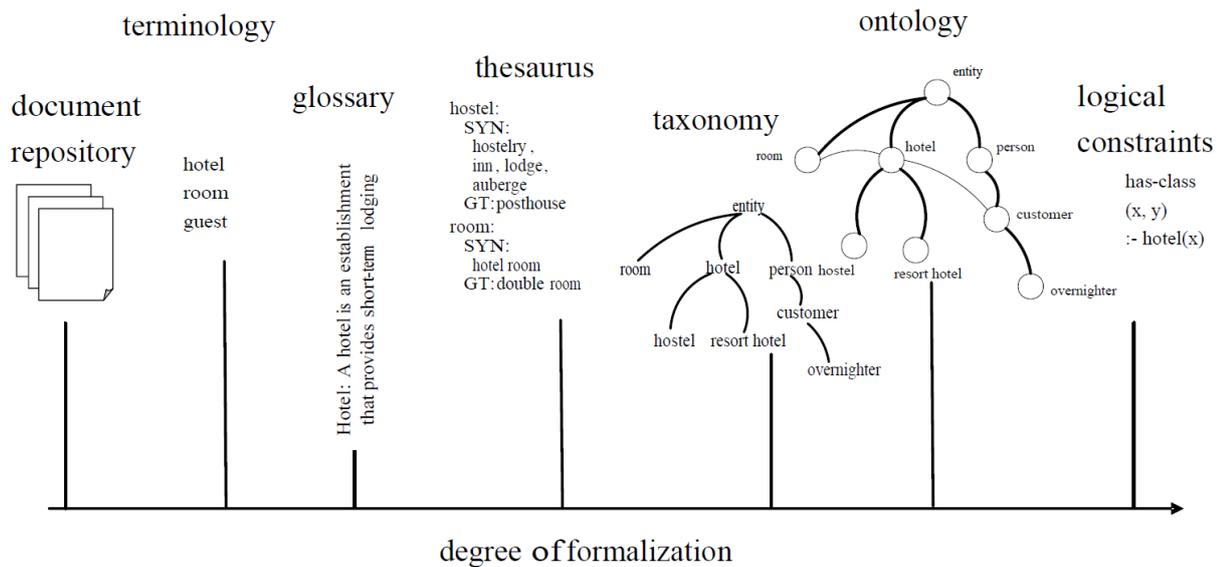


Figure 2.2.: The different degrees of formalization in KR methods: from unstructured textual content to ontology and logical rules. Source: Navigli et al. (2008: 72)

organized in a file-system-based folder structure. The degree of formalization increases if information is expressed in a defined terminology; this way data from different sources but modeled according to the given terminology can be applied by programs that are designed to use this kind of data. The same applies to glossaries and thesauri that are special more formalized cases of terminologies. An even higher degree of formalization can be found in taxonomies and, of course, in ontologies that model not only the pure vocabulary, but also relations between the defined terms. If, additionally, logical constraints are applied to formulate the relations between the terms, that is, for example, possible in Ontology Web Language (OWL) markup, we have the highest degree of formalization feasibly implementable as of the state of the art today.

Knowledge can be classified as *explicit* and *tacit*. Explicit knowledge is formal and structured and can be codified to be shared, while tacit knowledge is experiential, consisting of lessons learned while executing tasks/projects and insights gained from continuous problem resolution (North et al. 2014: 23). While explicit knowledge can be modeled relatively straightforward, applying the mentioned techniques, tacit knowledge, for example can be modeled only to certain degree because it is to a degree subjective, and cannot be objectively modeled. An example of subjective tacit knowledge is for example the art of painting: Even if a master painter describes the process of painting a picture in deep detail, an untrained individual will not be able to achieve the same result as the master painter following his instructions.

Knowledge representation (KR) is a semantically unprecise concept about what can be considered as KR. For example, document-based metadata annotation, as described in the section above, can be considered as KR, as well as an integrated Data Base (DB), or even better, a KB, that allows direct query and work with the actual data or knowledge items. This second aspect

of KR is more related to the understanding of the concept in the AI community.

To distinguish between the document and the knowledge approach for the creation of research databases, table 2.2 presents a good overview.

	Document focus	Knowledge item focus
1.	Find out what the core knowledge interests are	
2.	Find out which documents deal with given interest	find out which knowledge items deal with these interests
3.	Build a KB infrastructure	Find out which knowledge process to create and manage knowledge items
4.	Reorganize process to distribute knowledge	Build a KB infrastructure

Table 2.2.: Document vs. knowledge approach, after (Staab et al. 2001).

KR serves to maintain knowledge for use in information systems. *Knowledge modeling* is the process of organizing information, thus *knowledge representation* is organized and formalized information. For example, computer programming is a process of representing knowledge (Welty 1995). Here, data is manipulated and organized through formalized procedure (algorithms) that contain knowledge on what the data represents and how to manipulate the data to answer a certain question. Data formalized in an ontology or semantic net on the formal basis of *description logics* (Hitzler et al. 2010) is also regarded a representation of knowledge; that is because these languages contain mechanisms to formulate knowledge through defining a variety of relationships between data properties. This ability to assign and define relations between data properties is known as the degree or level of expressivity. More expressive modeling language can express a wider variety of statements about the model. Modeling languages of the Semantic Web — Resource Description Framework (RDF), RDFS, and OWL — differ in their levels of expressivity (Allemang et al. 2011: 25). A collection of, in this sense, formalized knowledge is referred to as a KB. Thus we can define:

A KR is formalized knowledge (Davis et al. 1993). A KB is a formalized collection of KR items.

The borders between the IS, KR, and KB are quite fuzzy, because the knowledge can be formalized in all of these three parts (algorithms, queries, data structures, semantic annotations, etc.) in different percentages of the whole system.

Seen from a problem-solving perspective, knowledge representation can be defined as follows:

Knowledge is the information about a domain that can be used to solve problems in that domain. To solve many problems requires much knowledge, and this knowledge must be represented in the computer. As part of designing a program to solve problems, we must define how the knowledge will be represented. A representation scheme is the form of the knowledge that is used in an agent. A representation of some piece of knowledge is the internal representation of the knowledge. A representation

scheme specifies the form of the knowledge. A knowledge base is the representation of all of the knowledge that is stored by an agent (Poole et al. 2010).

A successful representation of knowledge must be in a form that is understandable by humans and must cause the system using the knowledge to use it as if it *knows* it (Welty 1995).

In this context, it is of interest, that Bartelme (2005: 249) speculates that if knowledge is sufficiently well modeled — in data, rules, algorithms, semantic nets, etc.— and thus to a certain extent understandable by AI systems, there will not only be advantages but also disadvantages as a result. Bartelme (2005) further identifies another *externalisation of knowledge* phase in this context. In human history two phases of this *externalisations of knowledge* already took place, and triggered substantial changes in human self-conception. The first was the development of language in the palaeolithic, and the development of script or writing in the neolithic. AI, fueled by formalized knowledge will be the third of these disrupting phases. According to Bartelme (2005), the problem is the loss of control over externalized information and knowledge. The emergence of speech was followed by the *babylonian confusion* and the emergence of writing was followed by the *information explosion*. Here, the author wants to state that he does not share these concerns about the above depicted developments. These kinds of developments helped humanity to adapt and develop new capabilities that guarantee the success of the human species.

As one available way to facilitate knowledge representation in the above depicted sense, the semantic data model and its implementing SWT stack, as introduced in section 2.2.4, serve as a capable technology.

2.2.3. Standards and interoperability

In essence, a standard is an agreed way of implementing something. It is about consensus and a common way of interpreting and applying something. Formally put in the IT context:

A standard is a document that provides requirements, specifications, guidelines, or characteristics that can be used consistently to ensure that materials, products, processes, and services are fit for their purpose (ISO 2015).

The term standard lacks a very good reputation among some scientists and artists because standards are seen as rules that are suspected to limit the creativity and freedom of individual expressiveness and thus science and art. Some examples of often-heard expressions in this direction are: "Standards don't bother me," "standards are for scientific low-flyers and pedants," "I want to materialize my ideas, that's what counts in the end" (Birkenbihl 2006: 73).

In consensus there often lies a trade-off. To achieve interoperability, it is required to apply agreed-upon standardized rules that may limit the potential of the application. Another important pitfall is the vendor lock-in. If a vendor is so powerful that he can create quasi-standards, users of the system are locked into the vendor's software and hardware ecosystem. Well known cases of this kind of standard are Microsoft in the 1990s and the first decade of the new millennium, or

the computing giant International Business Machines Corporation (IBM) in the 1970s and 1980s (Birkenbihl 2006: 75). Thanks to the work of the Internet Engineering Task Force (IETF) and the W3C, all basic techniques of the Internet and the WWW remain open until now. And it is certain that the openness we have today is under massive attack from proprietary vendor monopolies and intergovernmental policy, see for example, (Berners-Lee 2000) or (Warnke 2011) for detailed and interesting insights on this topic.

Another aspect to facilitate interoperability is known as integration, in the scope of this work in particular *information-* or *data integration*. Research on information integration has focused on particular aspects of integration, such as schema mapping or replication, individually.

Data integration is the task of combining data residing at different sources, and providing the user with a unified view of these data (Lenzerini 2002).

Standards as introduced above, are a format to codify information in a well-defined way. This fosters understanding and, thus, reuse of datasets. Interoperability, thus, is more than standardized information in a syntactic form, as it is provided through metadata standards. Also, semantic interoperability has to be implemented to enable true interoperability. The semantics of metadata standards are often not implemented facilitating semantic technology that are able to make use of the semantics. Thus, this kind of interoperability must be inferred from the user, by integrating the data into his or her local system to work correctly. Consequently, enhancing metadata standards in the context of SWT-based applications adds further use and value to the data and offers a higher level of interoperability.

Following this section, we will have a high-level overview of some standards that this work is based on.

Standards of the Web

Three simple parts characterize the technique of the Web (see also section 2.3.2): a protocol to transmit data, the Hyper Text Transfer Protocol (HTTP); a method to locate resources, such as Uniform Resource Locator (URL); and a common format to display information in form of documents, the HyperText Markup Language (HTML).

- **URL:** The URL standard (Berners-Lee et al. 1994) is based on the Domain Name Service (DNS) (Mockapetris 1987), that allows referencing of IP addresses by a hierarchic schema of *domain names*. This hierarchy is administered by a central agency, the Internet Corporation for Assigned Names and Numbers (ICANN). The single resources, which can be documents of any kind, are then addressed by a path on the server.
- **HTTP:** The HTTP standard (Fielding et al. 1999) is responsible for the data transmission of the WWW.
- **HTML:** The HTML standard (W3C 2015a), in its current version 5, is the formalism to structure and display information in documents. HTML is interpreted and visualized via Web browsers.

Standards for RDM

State of the art in the RDM community is the application of metadata standards in the form as described in section 2.2.1. Thus, documents are annotated with metadata in a standardized format. This allows the federation and integration of the metadata itself very well, but, on the data level, interoperability is not really enabled, because it is hard to automatically access and compute the contents of the data documents. In the following, the most common standards for RDM are listed:

- **DublinCore**: The most used metadata standard on the WWW, well known from the HTML <meta>-tags, its main purpose is resource description (Dublin Core Metadata Initiative 2012).
- **DataCite**: Metadata schema for Digital Object Identifier (DOI) data publication, main purpose is DOI resource description (DataCite Metadata Working Group 2011).
- **ISO 26324**: Information and documentation — digital object identifier system. The standard that defines the DOI system (ISO 2012).
- **ISO 19115**: Geographic Information Metadata (ISO19115-1 2014).
- **INSPIRE**: EU initiative for SDI & Geographic Information (European Commission 2009).
- **DCAT**: Description of data resources in context of a catalog.

Standards regarding SWT will be introduced in the next section.

2.2.4. Semantic Web Technology

More than a decade after Tim Berners-Lee coined the term *Semantic Web* (Berners-Lee et al. 2001), which describes a sort of framework or a technology stack for bringing the concepts of the Web (Berners-Lee 2000) and of metadata, knowledge representation, and modeling together, many true and false knowledge, assumptions, and stories are spread about it. Many people do not really grasp what it is about and file it as too complex and unnecessary, which results in a lack of understanding of the possibilities, that come with the application of SWT. Berners-Lee et al. (2001) has the following argument on the topic:

Virtually all datasets suffer from some form of siloing, be it through access restrictions, schema differences, or offline storage. While substantial theoretical obstacles may remain, semantic technologies have frequently been vaunted as a technical solution for merging distributed, heterogeneous data (Berners-Lee et al. 2001).

The best and also the official description of SWT, is given by the W3C:

In addition to the classic “Web of documents” W3C is helping to build a technology stack to support a “Web of data,” the sort of data you find in databases. The ultimate goal of the Web of data is to enable computers to do more useful work and to develop

systems that can support trusted interactions over the network. The term “Semantic Web” refers to W3C’s vision of the Web of linked data. Semantic Web technologies enable people to create data stores on the Web, build vocabularies, and write rules for handling data. Linked data are empowered by technologies such as RDF, SPARQL Protocol And RDF Query Language (SPARQL), OWL, and Simple Knowledge Organization System (SKOS) (W3C 2015b).

It is important to see the difference between the *Web of documents* and the *Web of data*. The Web of documents is the Web as we know it, that is, independent Web applications based on data silos that are, in most cases, not interoperable with each other, while the Web of data is queryable and integratable between Web applications.

Metadata, semantics, integration, and analysis for research data has been practiced for several decades now and has, as explained above, well established concepts and technology. The problem until now is that those concepts are mostly implemented in closed systems, also known as *data silos*. In most cases, it is hard to integrate data from Silo A with data from Silo B, even if the data are from the same domain. If the data are not from the same domain, it is very hard to define a data integration procedure.

The basic idea of the Semantic Web is to describe the meaning (i.e. *semantics*) of Web content in a way that can be interpreted by computers (Hitzler et al. 2010). To implement this, it is necessary to cast knowledge into a machine-processable form. The resulting descriptions are often called *ontologies* and the machine-readable formats on which they are based are called *ontology languages*. Ontologies, as understood in this work, have their origin in the AI domain and are the central building blocks of the Semantic Web (Blumauer et al. 2006: 12). An ontology is an explicit specification of a conceptualization (Gruber 1993).

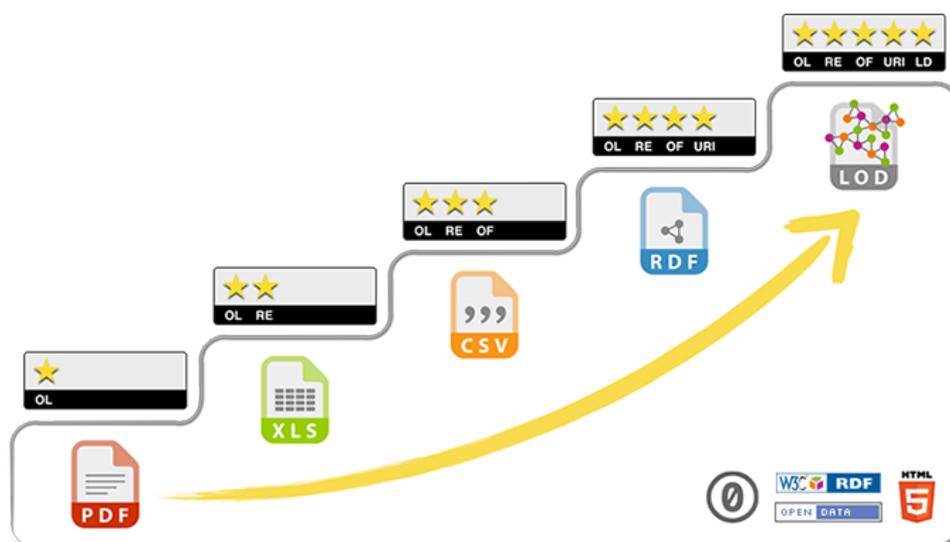


Figure 2.3.: Five-star deployment scheme for Open Data. Source: <http://5stardata.info/>, CC-0.

Tim Berners-Lee suggested a five-star deployment scheme for Open Data, a scheme that clas-

sifies methods of data publication by their openness, see figure 2.3. One star is granted if the data is published under an open license, Creative Commons (CC) for example, see section 3.5. If the data is available as structured data (e.g. Spreadsheet, instead an image of a graph), it gains two stars. Three stars are gained if the data is available in a non-proprietary format (e.g. Character-Separated Values (CSV) instead of Excel). If Uniform Resource Identifier (URI)s are used to denote things, so that people can link it, the dataset will gain four stars. Finally, if the dataset schema is linked to other data providers' schemas, in the sense of linked data (as described in the next sections), the dataset earns five stars.

In the following, a high-level overview of the main Semantic Web concepts and technologies, as well as an overview of actual SWT based-real-world applications of well known institutions and companies are given.

Semantic data model

In the given context, we will have a short excursus about the semantic data model, because the semantic data model is a special case of KR. There are two major aspects to semantic data models. The first aspect illustrates that a semantic model defines how the data modeled relates to the real world. The second aspect, the aspect that is of interest in the context of this work, includes the capability to express information in a way that enables information exchange and integration, without the need of a meta-model (or metadata annotation) for the dataset.

This is achieved through the structure of a semantic net. The basis of a semantic net are links representing relations between entities, enabling the modeling of relations between factual items. If you think about a net in terms of a graph containing edges and nodes, the relations or links are the edges and the items or facts are the nodes. A relation is modeled as a *triple* in the <subject> <predicate> <object> form. For example <car> <hasColor> <blue>, where a relation between an object <car> is related to a subject <blue> through the relation <hasColor>, meaning, that the object car has the color of subject blue.

The most prominent implementation of the triple-based semantic data model is RDF, an official W3C standard (Klyne et al. 2004). The standard is regarded as the fundamental building block of the *Semantic Web*. The Semantic Web community implicitly adopted Description Logics (DL) as a core technology for the ontology layer (Alamri et al. 2015). If triples are extended or modeled using DL, the outcome is an Ontology that can also be seen as a KB.

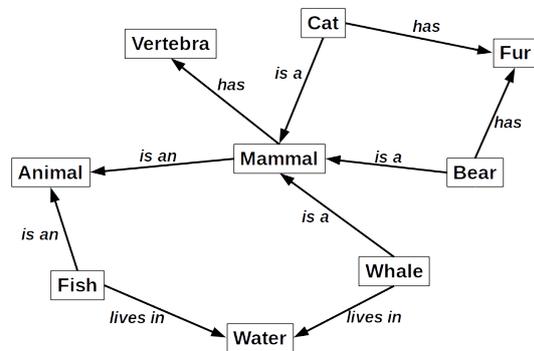


Figure 2.4.: Example of a semantic network.

Source: Own work.

Linked Data

The Web has evolved from a global information space of linked documents to one where both documents and data are linked. Underpinning this evolution is a set of best practices for publishing and connecting structured data on the Web known as *Linked Data* (Bizer et al. 2009).

The Semantic Web is a Web of data — of dates and titles and part numbers and chemical properties and any other data one might conceive of. RDF provides the foundation for publishing and linking your data. Various technologies allow you to embed data in documents (RDF in Attributes (RDFa), Gleaning Resource Descriptions from Dialects of Languages (GRDDL)) or expose what you have in Structured Query Language (SQL) databases, or make it available as RDF files (W3C 2015b).

*"The Semantic Web isn't just about putting data on the web. It is about making links, so that a person or machine can explore the web of data. With linked data, when you have some of it, you can find other, related, data"*¹. Unlike Web 2.0 mash-ups, which work against a fixed set of data sources, Linked Data applications operate on top of an unbound, global data space. This enables them to deliver more complete answers as new data sources appear on the Web (Bizer et al. 2009). Linked Data relies on two technologies that are fundamental to the architecture of the web: URIs and HTTP. These two technologies are supplemented by a technology, that is critical to the Semantic Web, or Web of Data, the RDF. RDF provides a generic graph-based data structure that is able to link and structure data that describes things in the world (Bizer et al. 2009). However, the most valuable potential of Linked Data is its ability not only to facilitate data linkage, or integration, on the facts basis, it also allows the linkage of semantics, or the creation of mappings, by establishing links between different data providers' repositories, and, thus, data models.

Vocabularies

Vocabulary is the semantic Web term for schema. A metadata standard as described above is a kind of vocabulary. An Ontology is also a vocabulary, but modeled in a stricter way, applying Description Logic techniques, for example, facilitated through OWL.

Vocabularies define the concepts and relationships (also referred to as "terms") used to describe and represent an area of concern (W3C 2015c).

At times, it may be important or valuable to organize data. Using OWL (to build vocabularies, or "ontologies") and SKOS (for designing knowledge organization systems), it is possible to enrich data with additional meaning, which allows more people (and more machines) to do more with the data (W3C 2015b).

An AP combining several existing schemas is also a vocabulary, for example, as presented in Curdt (2014b). In a broader sense, a vocabulary is the product or result of a data-modeling process.

¹<http://www.w3.org/DesignIssues/LinkedData.html>

Now, the interesting part of Semantic Web that is building on RDF, vocabularies is that it has mechanisms to formulate schema and vocabulary mappings. For example, you have two datasets that will be integrated. In dataset A the creator of an item is called "creator," and in dataset B the creator is called "author," RDF has the tools to formally describe the fact that the relationship described as "author" is the same as "creator". In a simple metadata specification formulated without use of RDF this kind of formalism is not available.

Query

In this context, the term *Query* covers technologies and procedures to algorithmically retrieve information from the Web of data. The algorithms are formulated in a Query language.

Query languages go hand-in-hand with databases. If the Semantic Web is viewed as a global database, then it is easy to understand why one would need a query language for that data. SPARQL is the query language for the Semantic Web (W3C 2015b).

In SPARQL, a query is formulated as a pattern that matches RDF triples in given endpoints and RDF stores. Query results can be returned in different formats (depending on the SPARQL server) and federated in complex mash-up sites or search engines that include data stemming from the Semantic Web.

Inference

Near the top of the Semantic Web stack one finds inference — reasoning over data through rules. W3C work on rules, primarily through Rule Interchange Format (RIF) and OWL, is focused on translating between rule languages and exchanging rules among different systems. (W3C 2015b).

Inference is a more advanced topic, that is not really implemented (yet) in the context of the presented infrastructure, but it would be possible to develop applications making use of the concept in combination with the CRC806-Database RDF data store, see chapter 8, later on.

A little example can shed a bit more light on the concept. Assuming we have a dataset that declares "Zaffaraya isA Cave", we can infer according to the ontology of the database, that declares, for example, "Cave isA Site", that Zaffaraya is an archaeological Site containing additional information such as geographic coordinates, an altitude, etc.. A program applying inference techniques could then add the information "Zaffaraya isA Site" to the dataset, if that would not have been case before.

2.2.5. Knowledge Management

Knowledge Management (KM) refers to a multi-disciplined approach to achieving organizational objectives by making the best use of knowledge. KM focuses on processes such as acquiring, creating, and sharing knowledge and the cultural and technical foundations that support them.

KM allows knowledge to be created, codified, stored, and distributed automatically (Chalmeta et al. 2008).

In the given context, *knowledge* is understood as defined above in section 2.1.3. Shortly summarized, knowledge is linked (contextualized) data and information, in this sense information of a higher (more organized) order.

An example for knowledge management is the annotation of published research articles with additional information, such that it can be queried along these annotations. For example, we have a particular bibliographic record about an article concerning the phenomenon of the "green Sahara" (see table 2.3). By annotation of spatial and temporal contexts of the articles content, we are able to find this resource by filtering for these spatial and temporal annotations in the KB. The contextualized bibliographic record, as described by table 2.3, evolved from an information item, the bibliographic record, into a knowledge item by annotation (and thus linking) of further information items (spatial, temporal, region, interval) about the resource described by the bibliographic record. This example makes clear that the annotation and linking of information is key to the creation and management of knowledge.

Table 2.3.: Example knowledge item, through linking of context information.

Information Item	Content	Context
Citation	S. Kröpelin, D. Verschuren, A.-M. Lézine, H. Eggermont, C. Cocquyt, P. Francus, J.-P. Cazet, M. Fagot, B. Rumes, J. M. Russell, F. Darius, D. J. Conley, M. Schuster, H. von Suchodoletz, and D. R. Engstrom (2008): Climate-Driven Ecosystem Succession in the Sahara: The Past 6000 Years. <i>Science</i> 320 (5877), 765-768, DOI:10.1126/science.1154913.	Bibliographic Citation
Spatial	westlimit=-14.7; southlimit=13.15; eastlimit=35.01; northlimit=32.59	DCMI BBox Encoding Scheme
Temporal	-8000 / -4000	ISO 8601
Region	name=Sahara	DCMI BBox Encoding Scheme
Interval	Mid-Holocene	CRC806-Database Vocab.

Technically, KM is the process of developing and maintaining one or many KB to support research or problem solving of any kind. The emphasis here is more on managing the organizational and technical infrastructure aspect of a KB.

Remarkably, in most publications about KM — for example (North et al. 2014; Chalmeta et al. 2008; Staab et al. 2001) —, it is seen as a business enabler and crucial for the advancements of organizations, companies, and firms. Thus, most publications regard KM as a tool for fostering economy, but, in this work, we will understand KM as a tool for and an enabler of scientific

research.

Knowledgebase

As defined above, a KB is a formal collection of KR items, or in less formal terms, a data base for the collection and management of knowledge. As we know from section 2.1.3, knowledge is information linked and related to data and to information about the data.

A KB is an executable knowledge map (Chalmeta et al. 2008), where the knowledge entities are mapped to defined contexts, to be applied and analyzed through query interfaces. The technology of choice in this work is what is summarized under the term *Semantic Web technology*, as described in detail in section 2.2.4. The Semantic Web is based on a concept known as Description Logics (DL) (Davies et al. 2006). Description logics are useful and efficient for knowledge representation, and reasoning about structured knowledge, fitting into the structural provision of RDF, RDFS and OWL technologies (Alamri et al. 2015). Typically, a DL based knowledge base comprises two parts:

- The terminological part that describes conceptualization, that is, a set of concepts and properties for these concepts, and captures the concept hierarchies (i.e. relations between concepts).
- The assertional part that captures the facts in an application domain.

In the example depicted in table 2.3, the schemas or vocabularies listed in the context column represent the terminological part, and the statements listed in the content column represent the assertional part. As the informed reader may already have noticed, this can be simply formulated in triple notation as well. Now, in the triple notation, the data (assertion) and the information (context, schema, vocabulary) are linked via triples to an entity to describe and persist knowledge. As already mentioned, the Semantic Web technology stack (RDF, RDFS, and OWL) allows to describe, store, query and manage knowledge. Thus, knowledge formulated in RDF notation is an instance of a KB.

Knowledge management in practice

Chalmeta et al. (2008) defines an Information System Development Methodology (ISDM) that is the basis of almost any Knowledge Management System (KMS). This methodology is divided into five phases:

1. analysis and identification of the target knowledge
2. extraction of the target knowledge
3. classification and representation
4. processing and storage
5. utilization and continuous improvement

In practice, as of today, most KB and KMS are not based on SWT. A KB or KMS strictly based on SWT, would consist of at least an RDF-based data storage, that is, a triple store, and a SPARQL endpoint as query interface, see figure 2.5 for example.

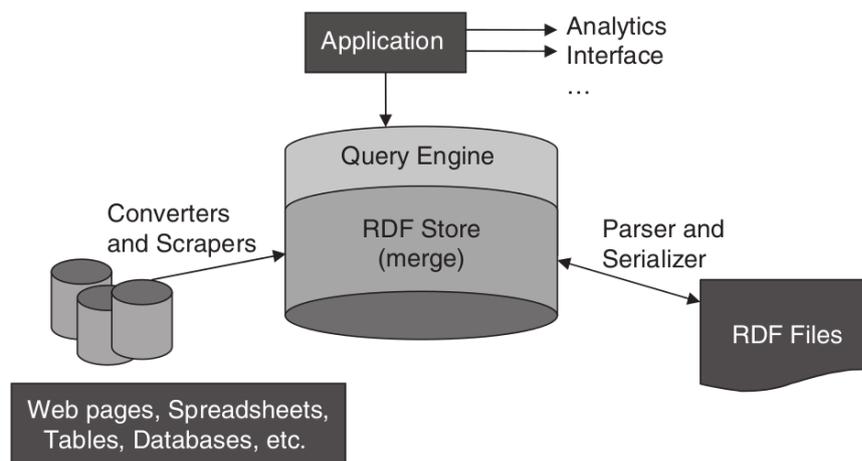


Figure 2.5.: Example architecture of a KMS strictly based on SWT. Source: Allemang et al. (2011: 57)

What we see instead as implementations of KMSs are complex and heterogeneous system architectures, either integrating several solutions for each building block of the system, that is, DB backend, catalog and query interface, UI and frontend, or completely from scratch custom developed applications.

A software system that allows the building of a KMS in one heterogeneous installation is, for example, SMW, see section 5.2.7 and the next section on semantic wikis. SMW allows the execution of all five phases of a KMS, as defined above. Some successful academic example KMS projects on the basis of SMW are (Alquier et al. 2009) and (Huvila 2012).

Semantic Wikis Collaborative knowledge management is often facilitated through wikis (Schaffert et al. 2008). The Wikipedia project is the most prominent example of a wiki. A main disadvantage of conventional wikis was the collaborative editing of structured information. Custom wikis do not have mechanisms to reuse or even query for structured information in their contents. The usual way of structuring information in wikis is through categorization/tagging and maintenance of lists and directories. Semantic wikis solve this shortcoming by adding functionality to allow entering, editing, and querying of *structured* data in wiki platforms (Vrandecic et al. 2006). In this research, we use the Semantic Mediawiki (Krötzsch et al. 2006) software, an extension to the famous MediaWiki software that is developed as the platform Wikipedia runs on. Semantic Mediawiki is a powerful, feature rich, and mature Open Source project maintained by a vital developer community. It offers many interfaces for data entry, and data discovery through the wiki based frontend or through API interfaces using standard and common formats. Semantic Mediawiki is applied in related applications and thematic domains. One example for a related

application would be the use of SMW for developing an archaeological collaborative research database by Huvila (2012).

2.3. On Networks, the Internet and interoperability

Why are networks introduced in this amount of detail in the following of this section? Because they are the very basis for each e-Science infrastructure, for RDM applications, and for KMS in the computer age, as of today.

One observation that is of importance especially to the field of geography is apparent. The transcendence of geographic distance has come to seem an inherent part of networked computer technology (Abbate 2000). This observation has strong ties to the actual economic and cultural globalization that is happening right now, and the Internet has a significant role in it by delivering a worldwide border- and space-transcending communication network. Concluding, links and nodes weaving a Web of knowledge is what the invention of computer networks brought to humanity, and it triggered certainly a massive revolution of communication and how the world interacts globally.

Beside being the technological and theoretical basis of the WWW, the essential technological foundation of the infrastructure presented here, networks are also the basis of information and knowledge, as described in previous sections (2.1.1, 2.1.2, 2.1.3). To understand networks in a wider sense, some basic concepts of networks will be introduced in section 2.3.1.

An overview of the development of the Internet and its technological basics will be given in section 2.3.2, to help understand the technological foundations of Web-based systems.

Interoperability is a key goal of good e-Science infrastructure, to enable sharing and reuse of the data and knowledge available through the system. Thus, some basic concepts of interoperability in IT systems is given in section 2.3.

And because SWT is an enabler of interoperability and knowledge management, and the basis of the field semantic e-Science, an introduction to SWT is given in section 2.2.4.

2.3.1. Networks

A thing is distinguished from another thing, they are related to each other, but are different and, because they are different things and not one, they can be related, or linked, or connected. These things can be named *nodes* and the relations, links, or connections can be named *edges* (Warnke 2011: 102). In this subsection we look at networks more from a theoretical, than from a pure technological point of view. We look at the elements of a network and at how they are organized. As we have seen in the sections before, information, knowledge, and wisdom emerge from relating (linking) or organizing bits of data and information to each other and are thus also networks. Because human social relations, the structure of a railway or a road infrastructure, and —of course— the Internet can be modeled as networks, it is possible to draw significant correlations between the structure of the Internet, a railway or a street network, and a networked society (Barabási et al. 2003).

It is an interesting side note that the Open Street Map (OSM) topology model is based on nodes and edges and can be interpreted as a graph. That is also the reason why the OSM data model is very suitable for implementing routing applications.

Network theory

If there is a direction of development in society or in nature and in general, then it is the direction toward higher structural complexity (Warnke 2011: 99). Structural complexity can be described and modeled through relations between entities (Warnke 2011: 99). It is useful to describe phenomena of nature and society through networks because it allows one to derive properties and conclude findings based on them that are not as visible without the network theory perspective (Warnke 2011: 101). On the basis of network theory, it is possible to make precise assertions about quantitative measures of a network that can lead to more insight or new findings. Large computer networks also have significant similarities to social networks of human society (Warnke 2011: 102). All this theory is of great value if you model your data in a network-based structure, for a example, a semantic net, like it is applied in this work.

As the basis for further theory we start with a definition of the term *network*:

A **network** is a configuration of nodes and edges, that can be mathematically modeled as a graph and possesses mechanisms to organize its structure.

Because networks can be modeled and described as graphs², network theory can be considered a subset of graph theory.

The main property that describes the structure of a network is its topology. Baran (1964) identified three major network types in his foundational theoretical work on a distributed network, which later became the Internet. These types are depicted in figure 2.6. The centralized (A) topology is also known as "star" network. The decentralized (B) network can also be described as a "tree" network. The distributed (C) network topology is also known as "mesh" network.

The topology of a network defines some important properties of networks, such as redundancy and resilience. The more edges or paths that exist between two nodes, the less prone to dysfunction is the network. Redundancy and resilience are, in this sense, more or less the same, with one important difference. Redundancy is the measure of the available paths or routes between two nodes, resilience is the overall redundancy.

In the following, we will define some measures, that help to quantitatively describe networks and graphs. Figure 2.7, shows on the right side a graph with nodes labeled by the nodes degree, and the degree distribution of that graph in the diagram on the left side.

An important measure of networks or graphs is the *degree* of a node (or vertex) v :

The **degree of a node** k is the number of edges incident to the node.

The *degree distribution* of a network, is defined as follows:

²A graph is a representation of a set of objects where some pairs of objects are connected by links (Trudeau 2013).

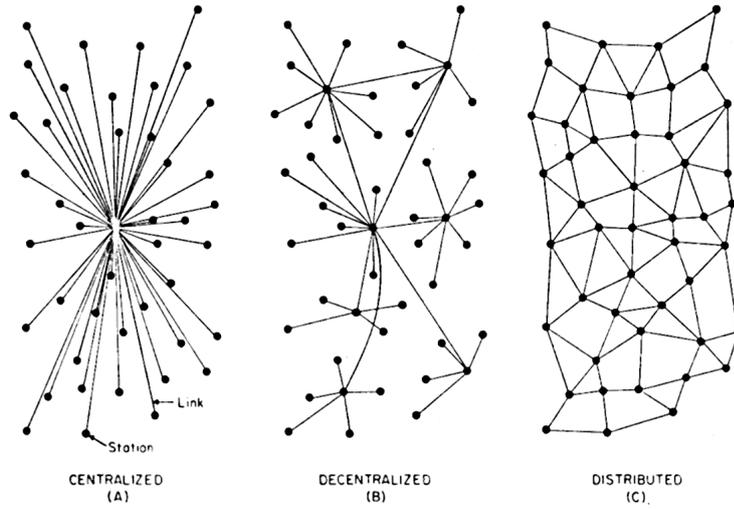


Figure 2.6.: Major network topologies by Baran (1964).

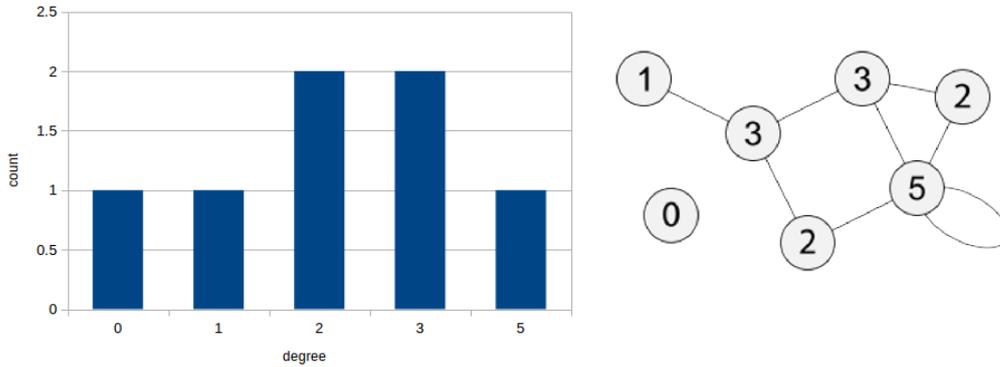


Figure 2.7.: A graph with labeled nodes by degree and its degree distribution.

The **degree distribution** is described through the probability distribution of the nodes degrees over the network. It is given as the fraction of nodes with degree k in the graph.

The degree distribution is very important in studying both real networks, such as the Internet and social networks, and theoretical networks. The distributions can take several forms as known for set distributions from statistics. Further important measures are the connectedness and the diameter.

The degree of connectivity or **connectedness** of a graph is the arithmetic mean degree of its nodes: $c(G) = \frac{\sum k(v)}{|v|}$.

The **diameter** of a graph is the number of nodes divided by the degree of connectivity: $d(G) = \frac{|v|}{c(G)}$.

In figure 2.8 A, the connectedness is uniform to the degree of $k = 2$ (where k equals the number

of links), you would need half as many links as there are nodes to reach the farthest point of a circular network. The diameter of the network then would be $N/2$ (where N equals the number of nodes). If every node is linked to the node just beyond the one it is immediately connected with (see figure 2.8 B), which is a degree of connectivity equal to four, then it is possible to skip a neighboring node and you only need half as many 'hops', $N/4$ (Warnke 2013).

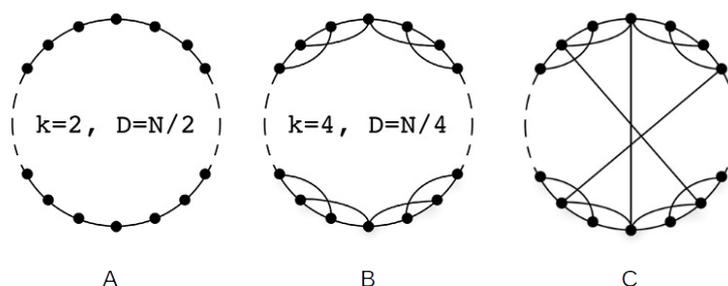


Figure 2.8.: Connectedness and Diameter of a network or graph. Source: (Barabási et al. 2003: 51).

A network structure that consists of a collection of nodes that are not uniformly connected, but with a small number of strongly connected and a large number of weakly connected objects (as depicted in figure 2.8 C), enables a considerably small diameter with very many nodes without an overall extremely high degree of connectivity (Warnke 2013). These strongly connected nodes are also referred to as *hubs*; only a few are needed to create a network with a small diameter and high cohesion.

Scale-free If we have a normal "Bell-curved" distribution pattern of degrees in the network, we speak about a *random network*. If the distribution is skewed, or looks like a logarithmic or power-law curve, we speak about a *scale-free network*. See figure 2.9 for the difference between random versus scale-free networks. The characteristic of random networks is that most nodes have a medium node degree and the degrees of all nodes are distributed around the average. In scale-free networks, which are much more numerous in reality and —especially in computer networks, most nodes have only few links but, by contrast, there exist some nodes which are extremely linked.

On the left side of figure 2.9 we have a highway network and on the right side major airline routes. Highway intersections do not have an unlimited number of exits; airports, on the other hand, differ in the number of starts and landings. Intersections have a typical number of access points, while the number of starts and landings can vary greatly; there are many small and very few big airports (Warnke 2013). This kind of topology is called scale-free because in these networks, that have few hubs with many connections and many nodes with few connections, a meaningful medium degree of connectivity is missing.

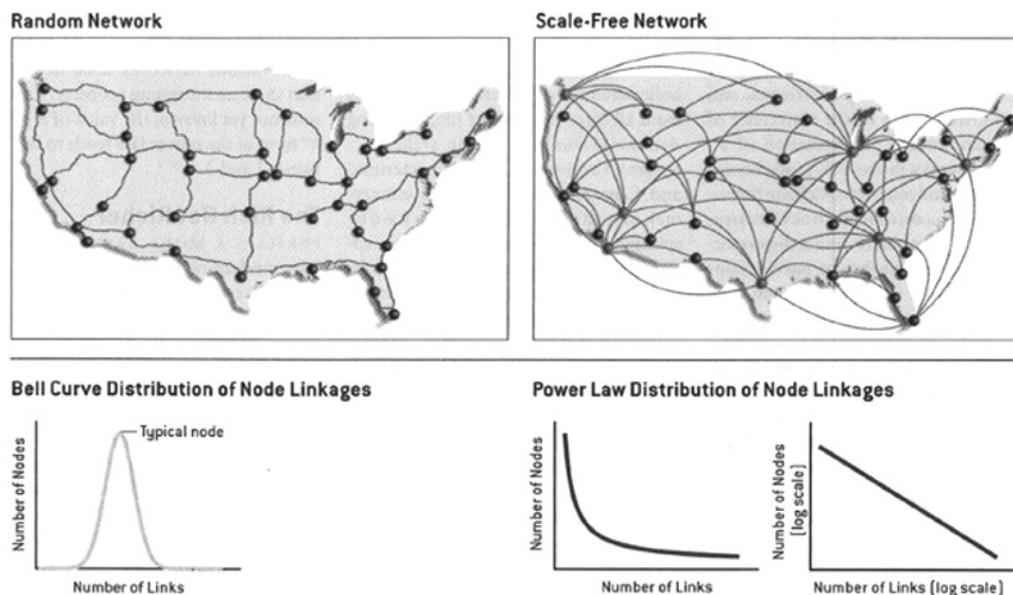


Figure 2.9.: Random vs scale-free network. Source: (Barabási et al. 2003).

2.3.2. Internet and World Wide Web

Today's Internet is arguably the largest engineered system ever created by mankind, with hundreds of millions of connected computers, communication links, and switches and with billions of users who connect via laptops, tablets, and smartphones; and with an array of new Internet-connected devices such as sensors, Web cams, game consoles, picture frames, and even washing machines (Kurose et al. 2010). The Internet starts to develop its full potential as a communication space, and this is mainly because of the distributed and dynamic creation of structured content (Galinski 2006: 47). Structured content means data stored in data bases, based on metadata, generic data models, and standards. The compliance with a certain standard and the accuracy with which the data is described such a standard has an obvious effect on the quality of the data, which is, to a significant extent, based on trust of the users (Galinski 2006: 47).

The Internet, or better its predecessor the ARPANET, was developed between 1969 and 1983 in a military context by the U.S. Department of Defense's Advanced Research Projects Agency (ARPA) to deliver decentralized communication infrastructure in the case of a nuclear war (Abbate 2000: 10). One of the most important principles for the development of the ARPANET/Internet was that it has to stay functional in the case of war, thus the infrastructure needed to be decentralized and redundant (Warnke 2011: 21). The ARPA, today known as Defense Advanced Research Projects Agency (DARPA), is an agency that funds research projects in civil universities and academia for military purposes. The research and development that created the Internet was conducted by civil scientists and research projects. The controversy over whether the Internet was developed by civilians (e.g. in academia) or by the military can be concluded by asserting, that academia was funded by the U.S. military to invent a distributed, redundant network based on the idea of Baran (1964), which then evolved into the Internet (Warnke 2011). Thus, the

Internet is not a recent phenomenon; it represents decades of continuous development (Abbate 2000).

A main characteristic of a distributed and redundant network like the Internet, is that there is more than one network path (edge) between two nodes (Warnke 2011: 22). The more paths that are available, the higher the redundancy and the more nodes that stay connected in case of any corruption of the network (Baran 1964).

The technological implementation of this idea is known as the *Internet protocol suite*, consisting of two important protocols: the Transmission Control Protocol (TCP) and Internet Protocol (IP), together well known as TCP/IP. The Internet protocol suite model architecture (Braden 1989) divides methods into a layered system of protocols (see fig. ??). There are four of these abstraction layers, as summarized from (Kurose et al. 2010: 49-53):

- **Application layer:** Includes applications such as HTTP, Simple Mail Transfer Protocol (SMTP), File Transfer Protocol (FTP) or the DNS protocol. The applications, or processes, make use of the services provided by the underlying, lower layers, especially the Transport Layer which provides reliable or unreliable pipes to other processes.
- **Transport layer:** Facilitates host-to-host communications and transports application-layer messages between application endpoints. The layer consists of protocols such as TCP or User Datagram Protocol (UDP).
- **Network or Internet layer:** Manages the exchange of network-layer packets known as *datagrams* by abstracting the network topology of the underlying network connections. The primary protocol of this layer is the IP that defines and manages IP addresses. The Internet's network layer also contains routing protocols that determine the routes that datagrams take between sources and destinations.
- **Link layer:** Includes the protocols used to describe the local network topology and the interfaces needed to effect transmission of datagrams to next-neighbor hosts. Examples of link layer protocols include Ethernet and WiFi.

The original purpose of inventing the Web, at Conseil Européen pour la Recherche Nucléaire (CERN) was to share scientific data (Berners-Lee 2000). As such, it was one of the first e-Science infrastructures, that evolved into a basic technology almost everyone uses today. The original funding proposal (Berners-Lee 1989) for developing the foundation of the WWW at CERN was titled: "Information Management: A proposal." In this proposal, Tim Berners-Lee combined several, then available innovative ideas into a distributed hypertext system, and provided building blocks such as a simple underlying HTTP, URI, and an easy-to-use markup language, HTML, to create human-readable, interlinked documents accessible across the Internet. The idea of allowing persistent publication of information on your server, which would be publicly available via an open protocol, combined with the possibility of linking this information arbitrarily, encouraged people to publish enormous amounts of data making the Web the biggest data collection available (Fensel et al. 2007). Today, the development of the WWW is governed by the W3C, that organizes the standardization of key technologies of the WWW. The W3C was founded in 1994

at the MIT Laboratory for Computer Science by Tim Berners-Lee.

2.4. Research Data Management

This section introduces the concept of RDM, by first defining it and then explaining the theoretical, practical, and some of the legal context of RDM in research settings. This introduction to the topic is followed by an investigation of the concept of *Research Data* because it is the main subject of RDM. Followed by an introduction to the *Research Data Lifecycle*, giving a structure to the RDM concept. An important aspect of RDM is the publishing of research data. Consequently, the most important concepts for publishing and citing research data are introduced as well. Finally, a short overview of RDM in Germany and especially in CRC's was given.

The proper handling and management of data is vital for the successful conduct of science. Allowing the reproduction of research by making the data on which the hypotheses are built accessible and available over long term.

To be clear what we talking about when referring to RDM, we first define the concept as follows:

Research Data Management (RDM) deals with methods, best practices, and implementations of infrastructure to archive, publish, and access research data (Brunt 2009). Pryor (2012: 7) defines RDM as: "[...] an active process by which digital resources remain discoverable, accessible, and intelligible over the longer term, a process that invests data and datasets with the potential to acquire values as assets [...] wider use".

Additionally, RDM deals with some overarching and organizational tasks such as research data policy and agreements, copyright, ethics, educational politics and funding (Engelhardt 2013).

These above definitions make clear that the concept of RDM is quite wide and covers a lot of ground. A more narrow and precise definition of RDM is due for each project, bespoke based on the demands of the funders and from the project goals and its participants.

Since around 2010, a Data Management Plan (DMP) is requested by most of the research funding agencies as part of a project proposal, see (DFG 2014) or (NSF 2011) for examples. In a DMP, the researchers who propose a project need to describe how they plan to manage the research data that is collected and produced by the research project. This includes processes and methods for documentation, publication, and archiving of the data (Donnelly 2011). However, the extent, comprehensiveness, and detail of this DMP's are dependent on the funding agency's demands. The DMPs are nowadays demanded by the funding agencies because in practice, only a small minority of researchers took care of RDM in their projects and day to day research work, as prominently described by Nelson (2009) in the journal "Nature."

Meanwhile, RDM is a well established part of most research projects. The topic is now internationally recognized and many national and international institutions and policy makers have emphasized the importance of scientific RDM for publicly funded projects (Curdt 2014b: 11).

From the comprehensive overview of international and national RDM initiatives prepared by (Curdt 2014b: 11 ff.), it is quite notable that almost all of the institutions and agencies cited and mentioned advertise or even demand Open Access policies for the publication of the research data. Although, in Germany at least, freedom of art and science is guaranteed by the constitution (Grundgesetz), Article 5, paragraph 3:

Kunst und Wissenschaft, Forschung und Lehre sind frei. Die Freiheit der Lehre entbindet nicht von der Treue zur Verfassung. (*Translation: Art and Science, Research and Education are free. The freedom of education does not prevail from the obedience to the constitution.*)

Winkler-Nees (2013), the program director of the "Scientific Instrumentation and Information Technology" (INF projects) branch of the DFG, correctly asserts that this constitutional right also facilitates the right not to publish research results. Anyway, there exists reasonable argumentation that this right is to be waived if the reproducibility of the according research depends on the underlying primary data for validation of the results. However, the emphasis of the guideline for good scientific practice (DFG 1998) was not on data publishing and sharing, but on *securing and storing* primary research data. The original wording of the statement regarding research data is as follows:

"Primary data as the basis for publications shall be securely stored for 10 years in a durable form in the institution of their origin" (DFG 1998).

Apparently, there is no formal demand or even mention of a process on how to access such primary data upon request. According to Klump et al. (2013), the announcement of the proposals was a consequence of several scandals of scientific misconduct. The primary storage of the research data was originally not a purpose (Curdt 2014b: 15).

In 2009, the DFG issued new "*Recommendations for Secure Storage and Availability of Digital Primary Research Data*" (DFG 2009), in which the need and importance of RDM and the definition of primary research data was addressed.

These DFG recommendations were influenced by the Berlin declaration on Open Access (Max-Planck-Gesellschaft 2003) where the major science funders, scientific organizations, and many universities and research institutions signed a statement for the facilitation of Open Access to knowledge in the sciences and humanities. This declaration is seen by many as the starting point of the Open Access movement. See section 2.5.3 for more details on the Open Access (OA) concept, and see section 3.1.2 for a description and analysis of the DFG demands on RDM, including its statements on Open Access.

2.4.1. Research data

To be able to define and reason about RDM, we need to be clear about what we understand as *research data*, first, because, simply put, it is about the management of *research data*. There are

many definitions of research data, and also in the context of RDM available. A rather general definition is:

Research data is the recorded factual material commonly accepted in the scientific community as necessary to validate research findings (UO Libraries 2015).

Notably, Oßwald et al. (2012) asserts that the term research data has to be redefined always in relation to the respective scientific discipline and the context of the research project.

The output of academic research in digital form increased overwhelmingly in the past decades and is still rapidly growing (Pryor 2012). Additionally, we have to deal with all kinds of data increasingly produced by industry and society. This observation is prominently described as the *data deluge* (Hey et al. 2009) or *information explosion* (Nelson 2009).

The following incomplete list contains some examples of possible research data objects:

- Documents (text), spreadsheets
- Photographs, films
- Geodata
- Collection of digital objects acquired and generated during the process of research
- Databases (domain specific, mostly from literature)
- Laboratory recordings, notebooks
- Questionnaires, transcripts, codebooks
- Source Code

The purpose of this short list is just to give an impression, of what the *data ground* of a research project can consist of. This ground can differ according to project domain and participating scientific disciplines. In this context it is helpful to classify the data into categories and types. This can be implemented according to different classification schemes. Some are introduced in the following.

The first and most fundamental distinction of research data is between *primary* and *secondary* research data.

Primary data is the data that is created during the conduct of research, that is observational measurements, interviews, artefacts discovered on archaeological excavations, and so on.

Secondary data is the data that is used during the conduct of research to support research questions and is from the published record of science and cultural heritage. This includes any kind of scientific publications (articles, books, datasets, etc.), governmental publications and statistical datasets, as well as all kinds of reliable, citable, and attributable sources.

The DFG (2009) demands the accessibility and storage of primary research data, by defining primary data as follows: "*Primary research data are data that result during the course of scientific research, experiments, measurements, surveys or polls. They serve as the basis for scientific publications.*"

A classification along the data provenance, use, and purpose is also common (Ludwig et al. 2013; Curdt 2014b; RIN 2008):

Observational data created from observations. Classic primary data, as explained above. For example, sensor data, survey data, sample data, aerial photographs, etc.

Experimental data created from lab equipment, often reproducible, but can be expensive. For example, radiocarbon-dated ages, grain size distribution of a sample, etc.

Simulation data generated from scientific models, where model and metadata (input data, boundary conditions) are more important than the output data. For example climate models, economic models, agent-based models, etc.

Derived or compiled data is reproducible but expensive. For example, text and data mining, compiled database, 3D models.

Reference or canonical a collection of published (peer-reviewed) datasets, the accepted published record. For example, gene sequence databases, chemical structures, administrative spatial data, published excavation artifact records.

Another well known classification of research data is along its statistical domain, into discrete, ordinal, continuous, nominal, interval, and ratio data.

Discrete data can only take certain values.

Ordinal data can be ranked and compared to each other, along an ordinal scale. An example would be school grades from 1 to 6.

Continuous data can take any value within a defined range. Example would be temperature measurement data or people's height measurements.

Nominal data are items which are differentiated by a simple naming system. For example, named temporal intervals or spatial regions.

Interval data are measurements or observations along a defined scale in which each position is equidistant from one another. For example, a rating of a product along a scale of one to five stars.

Ratio data is also data measured on a defined scale like interval data, but the values can be compared and differentiated in proportions to each other. For example, age in years, here 15 years is three times 5 years.

Digital Curation is a subset of RDM. It concerns the curational part of data management. The British Digital Curation Centre (DCC) defines the ongoing activity of “[...] maintaining, preserving, and adding value to digital research data throughout its lifecycle” as *Digital Curation*

(DCC 2015). Digital Curation ensures the availability of the data for reuse and discovery in the future (Curdt 2014b).

Traditionally, a curator is a manager or overseer in the realm of cultural heritage and the arts and mostly found in museums, libraries, and galleries. In this sense, a curator is a content specialist responsible for an institution's collection. Today, the term is also applied to interaction with digital media including compiling digital collections of images, Web links, movies, academic publications, and research data for example.

2.4.2. Research Data Lifecycle

The core element of any RDM concept is a "Data Lifecycle" (Eynden et al. 2009; Brunt 2009). According to Merriam Webster (2015): "A lifecycle is the continuous sequence of changes undergone by an organism from one primary form, as a gamete, to the development of the same form again." Lifecycle models are useful tools to depict complex processes, help to identify important components, roles, responsibilities, and demonstrate connections and relationships between parts of the process and the whole concept. Thus, it provides a framework to develop infrastructure and services supporting the research lifecycle. In some works, the concept of the research lifecycle is also referred to as RDM workflow (Addis 2015).

Table 2.4.: Overview of some State-of-the-Art Research Data Lifecycle Models.

Name	Steps	Description	Reference
DCC Curation Lifecycle Model	11	High-level overview of the stages required for successful curation and preservation.	(DCC 2015)
DDI Lifecycle v3.2	5	Lifecycle model and metadata specification for the social and behavioral sciences.	(DDI Alliance 2015)
IANUS Forschungsdaten Lebenszyklus	6	Data Lifecycle model documentation for the IANUS project.	(Trognitz 2015)
OSU Libraries Research Lifecycle	6	Combined Research, Project and Data Lifecycle model.	(Oregon State University Libraries 2015)
USGS Science Data Lifecycle Model	6	Data management steps to help ensure that USGS data are discoverable, and preserved beyond the research project.	(Faundeen et al. 2013)

The five Research Data Lifecycle models listed in table 2.4 are an almost random overview of different models depicting the same subject. The models are all more or less tied to an institutional, project or discipline setting and generally reflect the interests, perspectives, and biases of the agencies that created them. But models mask complexity, tend to overlook heterogeneity and diversity, and depict some view of the ideal.

As an example for a deeper investigation of the concept, the Oregon State University Libraries (2015) Research Data Lifecycle model, shown in figure 2.10, will be described in detail in the following. In this model, research is described from the project point of view. The research lifecycle depicts the stages and processes of a given research project. The project is divided into three major parts, the i.) project planning phase, the ii.) research data lifecycle, and the final iii.) publication of results and project wrap-up phase.

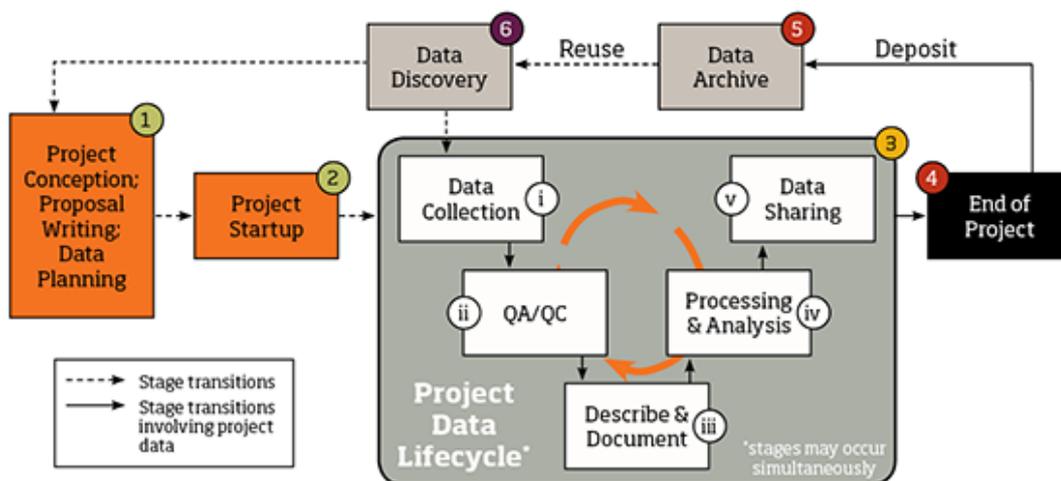


Figure 2.10.: The Research Lifecycle divided into 6 research stages and the data management part is subdivided into 5 stages. Source: (Oregon State University Libraries 2015)

The three major parts and the six stages of the research lifecycle as graphically depicted in figure 2.10 are detailed in the following list:

- **Project Planning**

1. Project ideas are developed and formulated in a project proposal. This includes in most cases already a review of existing data sets and sources, and the formulation of a DMP.
2. Upon acceptance of the project, all data responsibilities are communicated to project members. Minor revision of DMP if necessary.

- **Research Data Lifecycle**

3. Project Data related activities (all stages may occur simultaneously)
 - a) Organize files; design a categorization of the given data domain, implement a backup strategy, and access control and security
 - b) Check the incoming data for appropriateness and formal completeness
 - c) Contextualize the data, geographically, temporal, categorizations, and tagging to make the data discoverable in the data catalog interface. This stage also includes techniques of Linked Data, for improving interoperability, data integration, and reuse
 - d) Document and manage file versions to maintain the provenance of the datasets

e) Expose the datasets in open data formats and standards to improve sharing

- **Project Wrap-up**

4. Write papers and publish the results of the project
5. Deposit the according datasets in the RDM archive
6. Link the datasets from the project publications

This lifecycle was chosen as an example, because it is well conducted, detailed, and has all aspects that other models listed in table 2.4 brought to the table. Another positive attribute of this model is the inclusion of the project planning and the project wrap-up phases that are ignored in most of the other models, but are responsible for a considerable amount of work that is delivered during a research project.

2.4.3. Publishing and citing research data

To facilitate better reuse, and to make research datasets a valuable asset, in the sense of a publication that earns the researcher merit and impact, methods to formally publish and cite data are developed. The publication of research data is today mostly done in descriptive forms in text-based publications such as journal papers, books, dissertations, or project reports.

Callaghan et al. (2012) draws an explicit distinction between Published and published data: published data is at least available, while Published data is persistent, documented, and peer-reviewed. Publication refers to the scholarly literature, while publication is used in the sense of any kind of printed and distributed material (Kratz et al. 2014).

Data can be published in manifold and heterogeneous ways. However, the following three publication models are mostly applied and seem to prevail, according to several publications (Dallmeier-Thiessen 2011; Curdt 2014b; Kratz et al. 2014). These models are:

- i. data published in an independent **repository**,
- ii. data published in a **data journal**,
- iii. data as **supplementary material** of a traditional scholarly publication.

Publishing data in a data repository (i.) has the advantages that these repositories specialize in the publication of data sets. So, they almost always have long-term preservation strategies and sophisticated interfaces for annotation and description of the datasets. Data can also be deposited in a distinguished repository to supplement a traditional scholarly publication, if the repository supports persistent handles, also known as URI, to allow the citation of the dataset from a scholarly publication. From the current developments in the field, it seems that the DOI model (ISO 2012) will be established as the standard for this type of referencing and citation of resources.

Since some years, more and more data journals have emerged. The publication of datasets in the form of a data paper (ii.) has the advantages that they do not need to have any interpretations

of the data, no discussions and no conclusions. This gives the opportunity to publish data that is measured or generated, but can't be properly analyzed or otherwise used due to limited (time, financial, human, intellectual, etc.) resources. However, most data journals demand a very strict structure of the data papers. In most journals, the structure has the following sections: abstract, collection methods, and a description of the dataset (Kratz et al. 2014).

The publication of data as supplement to a traditional scholarly publication is the most familiar kind of data publication (iii.). This kind of data publication supports primarily reproducibility of the assertions claimed by the according article, but it does not necessarily support the reuse of the data (Kratz et al. 2014). An often observed phenomenon in this realm is, for example, the publication of supplemental data in the form of a table in a PDF or Microsoft Word™ document.

Proper publication of research data is a value-adding act and contributes significantly to the much desired achievement of academic impact and, thus, credit (Pryor 2012).

Citability of Web resources and Digital Object Identifiers

DOIs one method that seems to evolve as the quasi standard to cite scholarly resources on the Web. This is mainly because of the advantage that resources can change their WWW location, that is, their URL, to which the DOI can be updated to redirect to the changed URL to preserve the reference. The persistence of the reference link (URI) is the only important criteria for citability, a DOI is not necessary to achieve this, but it simplifies the maintenance of the reference. Important to note here is that a DOI identifies an object and not its location, the current location is just a mandatory metadata item to which the DOI system redirects on request. The DOI System is an implementation of the Handle System®. The Handle System®, developed by the Corporation for National Research Initiatives (CNRI) is a general purpose distributed information system designed to provide an efficient, extensible, and secured global name service for use on networks such as the Internet (NISO 2010). Further prominent examples for handles in other domains are the International Standard Book Number (ISBN) and International Standard Serial Number (ISSN) systems.

A Digital Object Identifier (DOI) is a name (not a location) for an entity on digital networks. It provides a system for persistent and actionable identification and interoperable exchange of managed information on digital networks (NISO 2010).

There are also other similar systems in existence, such as for example, *OpenURL* or Persistent Uniform Resource Locator (PURL)³. The OpenURL is a standard to coin metadata for a URL to deliver storage location independent referencing and linking (NISO 2004). The OpenURL system works similarly to the DOI system by providing a resource location brokerage or handle. Additionally, the system implements a User Rights Management (URM) system, that is able to check and maintain access rights to the resource. The idea and initiative for the OpenURL standard goes back to the *Xanadu* project by Nelson (1990), one of the first hypertext systems.

³<https://purl.org>

A PURL is a URL that is used to redirect to the location of the requested Web resource. Thus, it is also a link brokerage or handle system. The PURL concept was developed at Online Computer Library Center (OCLC) in 1995.

The basic functionality of the DOI system is to resolve the DOI and redirect to a currently valid URL that provides a landing page for the resource in question. On this landing page, a set of defined metadata elements, see table ??, are given and access to the resource is provided. The access to the resource itself can be restricted, the access to the metadata must be open (IDF 2015). In 2000, the syntax of the DOI was standardized through National Information Standards Organization (NISO) (NISO 2010). The DOI system was approved as an ISO standard in 2010 (ISO 2012; IDF 2015).

The "DOI Kernel" is a minimum metadata set with two aims: recognition and interoperability (IDF 2015). The DOI system data model consists of a data dictionary and a framework for applying it. Together, these provide tools for defining what a DOI name specifies (through use of a data dictionary) and how DOI names relate to each other. This provides semantic interoperability, enabling information that originates in one context to be used in another in ways that are as highly automated as possible (IDF 2015).

DOI names consist of a prefix of the form 10.ORGANISATION-ID and a suffix, that can be freely minted by the issuing organization. DOI names are not case sensitive. Because, a DOI is a URI, a Schema-Identifier doi: is assigned; this yields the following form for a DOI reference: doi:10.Organization-ID/suffix.

Merit and impact from published data

The main incentive for data publication is the increase of a researchers scientific impact and gain of merit that help to improve the researcher's reputation and position. With the introduction of DOIs, it is now formally possible to cite a published data set like any other traditional publication in a scholarly work. A record in the reference list would look like this:

<Author's> (<Year>): <Title>. <Publisher>, <DOI>.

The citations can be tracked through several providers. DataCite, one of the main issuers of DOI's provides detailed access statistics for each DOI through their Web portal, for example. The publisher Thompson Reuters recently launched the *Data Citation Index*, that aims to be included in the ISI Web of Knowledge citation index in the future. The academic platform from Google, *Google Scholar*, is also including data set publications in their citation statistics.

There is still discussion of whether a new identifier should be created when, for example, a dataset is updated by fixing an error. New identifiers would break the possibly already existing citations of the data resource. In contrast, the DCC (Ball et al. 2015) Dataverse⁴ and the UK Natural Environment Research Council (NERC) (Callaghan et al. 2012) insist that any change to a dataset should trigger a new identifier (Kratz et al. 2014). This seems to be a contradiction to

⁴<http://dataverse.org/>

the previously introduced principle of reference persistence, but only if the already referenced identifiers are abandoned. If the previous identifier stays existent, there is no contradiction. It is then in the interest of the data publisher to reference newer versions from the records of the previous versions.

2.4.4. RDM in Germany

In Germany, the topic of RDM is mainly handled by the Research Funders, such as DFG, Bundesministerium für Bildung und Forschung (BMBF), or the European Union (EU). The universities are just beginning with to setup their own institutional policies and according repositories and infrastructures for RDM.

The DFG report "Safeguarding Good Scientific Practice" (DFG 1998) was the first to contain a general, cross-disciplinary requirement to preserve data beyond a fixed period of time. Correspondingly, there was originally no requirement to provide research data for scholarly purposes and for reuse (Winkler-Nees 2013). The issue of dealing with research data at universities has as yet no particular nationwide coordinated response in Germany. In contrast to international approaches, German universities do not view themselves as under the obligation or even capable of initiating measures to improve the situation (Winkler-Nees 2013).

RDM in CRCs

The DFG maintains CRC's as large long-term research projects where universities and research institutions, as well as industry partners in some cases, collaborate in an interdisciplinary context on an overarching research question (DFG 2012). CRCs are designed to facilitate the specializations within universities and research institutions and should enable long-term research concepts and projects. A further goal of the DFG CRCs is the support and funding of early-stage researchers and PhD students, as well as gender equality in the supported institutions (Engelhardt 2013). A CRC has, on average, 20 sub-projects that should produce considerable amounts of research data (Effertz 2010). Because of the interdisciplinary nature of CRCs, the resulting data is comparably heterogeneous with regard to its contents as well as in data formats. This results in special demands for e-Science and RDM infrastructure in the context of CRCs.

To address these special RDM demands, since 2007 it is possible to propose an own sub-project "Informationsinfrastruktur (INF-Projekt)" (information infrastructure) in a DFG-funded CRC. Their purpose is to facilitate research data management within the CRC, which can include the approach to data management as well as the establishment of the necessary infrastructure. The concept of the INF projects is designed to encourage cooperation between researchers and their local infrastructure institutions, such as the library or the computing centre (Engelhardt 2013).

However, the main duty and goal of these INF projects is clear. It is the facilitation of long-term secure storage, meaning at least 10 years after the project ends, as proposed in the DFG "Proposal for good scientific practice" (DFG 1998) and accessibility to research data produced

in the CRC (DFG 2012). Additionally, the development of tools and infrastructure to support and facilitate communication and research within the project based on data is encouraged by the DFG as well (DFG 2012).

An important part of the DFG research data management policy is to support own RDM projects for CRCs. The DFG believes that this approach helps to find the best RDM solution for the research context in the given CRC (Engelhardt 2013).

2.5. e-Science

In this section, the concept of e-Science will be introduced and defined including its sub- or related concepts of Semantic e-Science, Open Science including Open Access and the five further open principles, as well as Linked Science. The connection between these concepts will be shown, as well as its application within the presented work was outlined. These concepts are central to the design and implementation of the presented CRC806-Database semantic e-Science infrastructure.

As the title suggests, this thesis places itself within the context of Semantic e-Science (Hey et al. 2005; Fox et al. 2009; Ma et al. 2015). The term e-Science was coined by Jim Gary, an U.S. American computer scientist described as "where IT meets scientists" (Hey et al. 2009). In the book "The Fourth Paradigm," Hey et al. (2009) founded the basis for this new field of research, or maybe even research discipline. In this context, the fourth paradigm is understood as the next or current scientific paradigm in the succession from the first paradigm, the *empirical sciences*, describing natural phenomena from observations, which is common to humans for at least the last thousand years. The second paradigm is understood as the *theoretical science*, where models, mathematics, and formalizations revolutionized science, at least since Newton. This was followed by the third paradigm, the *computational science*, facilitating simulations of complex phenomena, that started during World War II. Now, we have on top of that the fourth paradigm, understood as *data intensive science*.

Originally, Taylor (1999) introduced the term and concept more than a decade ago as follows: "*e-Science is about global collaboration in key areas of science, and the next generation of infrastructure that will enable it*". Hey et al. (2009) formally defined the concept in the book "The 4th Paradigm", and provided the following definition:

e-Science is the set of tools and technologies to support data federation and scientific collaboration (Hey et al. 2009).

Today, almost any science discipline is, at least partly, conducted using computers to handle and analyse large amounts of data, and can be considered as *Data-Intensive Science*, in the sense that Jim Gary defined it in Hey et al. (2009). Data-driven means in this context, that new patterns and conclusions emerge from application of computer algorithms, data models, and statistical data analysis.

The *fourth paradigm* is closely related to the phenomenon called the *data deluge* (Hey et al. 2003), also known as the *information explosion* (Beath et al. 2012), describing the exponential increase in the production of scientific data (Tolle et al. 2011) that emerges from the ever-growing number of data generated by data-producing scientific applications such as sensors and sensor networks (Devaraju et al. 2015), Location Based Services (LBS) (Baaser 2010), computer model simulations (Taylor et al. 2012), environmental monitoring (Tilly et al. 2014; Hoffmeister et al. 2014), and modelling (Brocks et al. 2014), as well as recently emerging low-cost and ubiquitous remote sensing applications (Bendig et al. 2014; Aasen et al. 2014), social networks and the Internet of Things (IoT) (Atzori et al. 2010), and generally all kinds of data-generating computer applications used and applied during, and for the conduct of scientific research.

However, there are also critical voices concerning the fourth paradigm, for example Kasrtens (2012) has an interesting critique to the assumption, that there would be a whole new paradigm, and that making meaning, as he calls it, from the complexity of nature, is really not so new. He argues, "I would suggest, though, that this paradigm comes into play whenever the rate of accumulation of data greatly outstrips the rate of accumulation of interpretation, i.e. the rate at which the scientific community can assimilate data into an interpretive framework" (Kasrtens 2012). His argumentation stems from an observation that sea floor spreading was proven in 1965 by synthesis of magnetic anomalies (Pitman et al. 1966) that came from previously published works. "That expansion wasn't done by acquiring new data, but by mining the existing archive with newly insightful eyes—a Fourth Paradigm inquiry 45 years before the term was coined" as Kasrtens (2012) argues.

Considering this, the difference now is that there is a lot more data (digitally) available for almost any topic and there will be exponentially more in the times to come, that this scientific synthesis work on existent and available data will be more and more complex and also more and more necessary to find new conclusions. This change in the scientific environment demands new and adapted methods and these computational electronically enhanced methods are what forms e-Science.

In conclusion, in some sense the term *e-Science* describes a scientific paradigm (Bohle 2013), the fourth paradigm (Hey et al. 2009); the *e* stands for both *electronic* and *enhanced* Science. It refers to the science that is conducted through collaborations enabled by the Internet (Taylor et al. 2014). However, Taylor (1999) also emphasized that there is much more about e-Science, than just putting research results on the Web. He formulated it as follows:

The WWW gave us access to information on Web pages written in HTML anywhere on the Internet. A much more powerful infrastructure is needed to support e-Science. Besides information stored in Web pages, scientists will need easy access to expensive remote facilities, to computing resources, and to information stored in dedicated databases (Taylor 1999).

As such, e-Science is the application of computer technology to the conduct of modern scientific research, including the preparation, experimentation, data collection, results dissemination,

and long-term storage and accessibility of all materials generated through the scientific process (Bohle 2013).

In many domains and research settings today, e-Science systems evolve around the development of new methods to support scientists in conducting scientific research, by facilitating computational resources to access and analyse vast amounts of data accessible over the Internet. However, discoveries of value are not made by simply providing computational tools, a *Cyber Infrastructure* (CI), or by the application of algorithms, or performing a set of instructions to produce a result. Instead there needs to be an original, creative aspect to the activity, that by its nature cannot be automated. This has led to various research that attempts to define the properties that e-Science platforms should provide in order to support a new paradigm of doing science, and new rules to fulfill the requirements of preserving and making computational data results available in a manner such that they are reproducible in traceable logical steps, as an intrinsic requirement for the maintenance of modern scientific integrity that allows an application of "Boyle's tradition in the computational age" (Bohle 2013).

e-Science increasingly takes place not in subject domain silos but across disciplines. It facilitates new opportunities for interdisciplinary research and innovation. Enabling this inter/cross-disciplinary work substantially increases the interoperability problems at computational, semantic, and also organizational and professional levels (European Commission 2006).

2.5.1. Semantic e-Science

The term *Semantic e-Science* extends e-Science with the application of semantic methodologies and technologies (Ma et al. 2015), in this work also referred to as SWT, as well as related knowledge-based approaches (Fox et al. 2009). SWT, in general, facilitates and helps to understand the meaning (semantics) of data and, thus, helps to formulate information and even knowledge (see section 2.1). Fox et al. (2009) introduced the field of *Semantic e-Science* in his contribution to Hey et al. (2009) book "The fourth paradigm," as follows:

Those involved in extant efforts (noted earlier, such as solar-terrestrial physics, ecology, ocean and marine sciences, healthcare, and life sciences) have made the case for interoperability that moves away from reliance on agreements at the data element, or syntactic, level toward a higher scientific, or semantic, level. Results from such research projects have demonstrated these types of data integration capabilities in interdisciplinary and cross-instrument measurement use. Now that syntax-only interoperability is no longer state-of-the-art, the next logical step is to use the semantics to begin to enable a similar level of semantic support at the data-as-a-service level (Fox et al. 2009).

Semantic e-Science concerns state-of-the-art technologies in knowledge representation, data interoperability, vocabulary and data services, as well as data processing (Ma et al. 2015). The major application and focus of Semantic e-Science is on data reuse, and is looking to answer questions such as "How do I use these data that I did not generate?", or "How do I use this data

type, which I have never seen, together with the data I use every day?” or “What should I do if I really need data from another discipline but I cannot understand its terms?” (Ma et al. 2015).

2.5.2. Virtual Research Environment

The idea of a Virtual Research Environment (VRE), which in this context includes cyberinfrastructure and e-Infrastructure, arises from and remains intrinsically linked with, the development of e-Science (Fraser 2005).

A virtual research environment (VRE) is an online system helping researchers collaborate. Features can include collaboration support, like Web forums and wikis, document hosting, and tools depending on the discipline, such as data analysis, visualisation, or simulation management (Carusi et al. 2010).

From a technological perspective, virtual environments are based primarily on software services and communication networks. Virtual research environments are essential components of state-of-the-art research infrastructures (Allianz Initiative 2011).

The VRE helps to broaden the popular definition of e-science from grid-based distributed computing for scientists with huge amounts of data to the development of online tools, content, and middleware within a coherent framework for all disciplines and all types of research (Fraser 2005).

2.5.3. Open Science

Open Science describes the on-going transitions in the way research is performed, researchers collaborate, knowledge is shared, and science is organized (European Commission 2015). It is enabled by digital technologies, and driven by:

- the enormous growth of data,
- the globalisation and enlargement of the scientific community to new actors (e.g. citizen science), and
- the need to address societal challenges.

Open Science is about making scientific research process and most importantly its output (data) accessible to all levels of an inquiring society, it is based on the following six open principles (Whyte et al. 2011):

Open Educational Resources Free and open resources for teaching and learning. Such as MIT open course ware or Massive Open Online Course (MOOC).

Open Access Making scientific publications accessible by the public. See section 2.5.3 for more details.

Open Peer Review Transparent and comprehensible and quality assurance, by openly documented review and discussion of proposed research findings.

Open Methodology Detailed documentation of the application of methods including the whole process, for example software toolchains or experiment steps.

Open Source Publishing the source code, of tools applied for the conduct of scientific analysis, for review and feedback by the community. See section 2.5.3.

Open Data Making research data openly available, and reusable by applying open licenses. See section 2.5.3 and 3.5.

The main idea behind these six principles is to open up the scientific process. This involves not only opening up the final results of research, though this would be a good first step, because this is sadly not the case for now. In addition, however it aims also to shed light on the applied methods, tools, and data, that led to the research results. A main goal of open science is to facilitate reproducibility of research.

Granting access to publications and data may be a step toward open science, but it's not enough to ensure reproducibility. Making computer code available is also necessary — but the emphasis must be on the quality of the programming (Hey et al. 2015).

The idea of a more open and accessible science is not really new, and absolutely makes sense, but policies and copyright law, for example, in the practical conduct of science led to the current restricted-access situation. Open Science began in the 1600s with the advent of the academic journal when the societal demand for access to scientific knowledge reached a point at which it became necessary for groups of scientists to share resources with each other so that they could collectively do their work (David 2004).

Open Access

Most scientific knowledge is created in a publicly funded context and is already paid for by the taxpayers (Sitek et al. 2014). Because scientific knowledge is mostly published in scientific journals, OA developed chiefly in the journal publishing sector.

Figure 2.11 shows the OA process as proposed by (Bartling et al. 2014; Sitek et al. 2014). The depicted process makes clear, that there are several avenues to Open Access publishing. The most open solutions are via preprint servers, such as arXiv, or obviously via a dedicated OA journal. The OA journal implements a peer-review process, as is well known from traditional journals. This kind of OA publication is then labeled "OA Gold." The preprint servers do not review the papers' contents, rather only the formalities. Preprint servers are common in natural sciences such as physics, mathematics and chemistry, to gain feedback from the community before the paper is submitted to a peer-reviewed journal for official publication. Today, most traditional journals also offer an Open Access option for publication, for which option, the authors have to pay a fee, depending on the journal pricing policy. However, this paid Open Access option, called "OA Green" (see fig. 2.11), does not, for most part, grant all the copyrights, for example, for reuse and sharing.

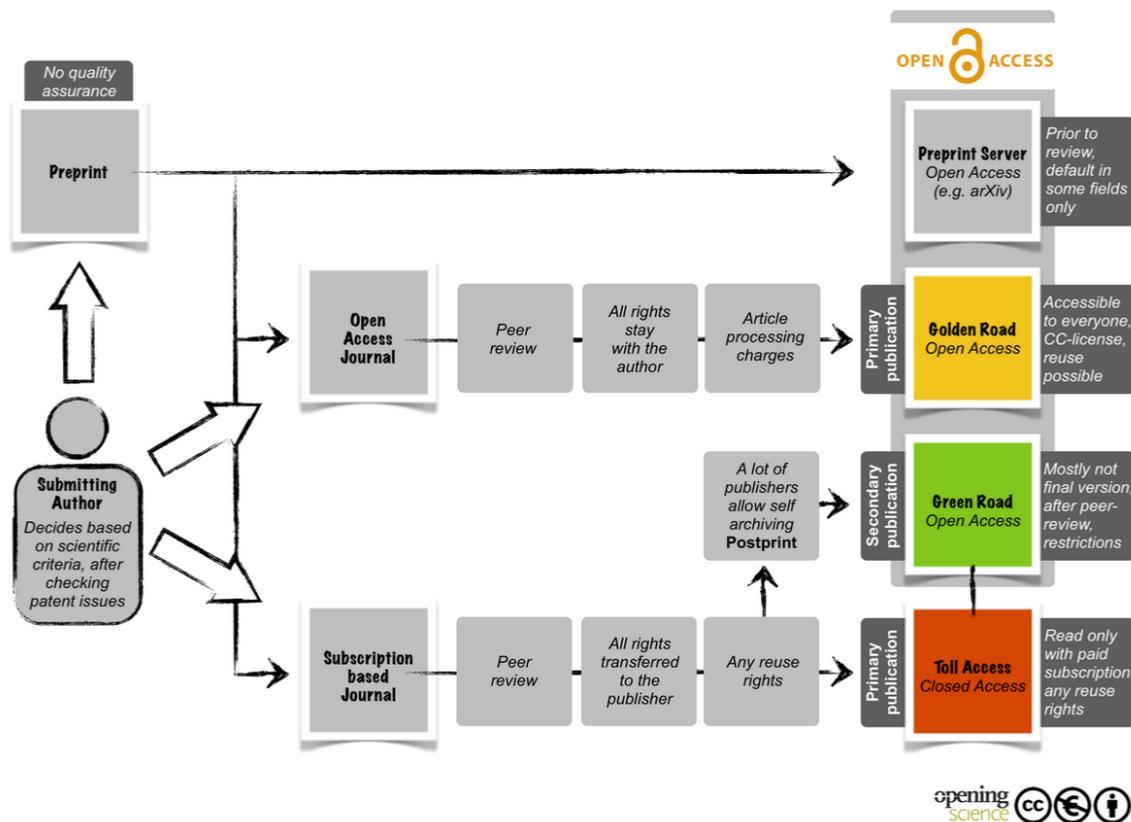


Figure 2.11.: Overview of the Open Access process. Source (Bartling et al. 2014; Sitek et al. 2014).

OA also has a positive influence on the visibility and outreach of research results. Obviously, a broader audience access to the publications increases the probability of citations for that publication. The idea of OA is today broadly accepted and even promoted by government bodies, such as the German Bundesregierung or the European Commission, as well as from science funders, science organizations and universities. Some of those institutions signed the Berlin Declaration on Open Access (Max-Planck-Gesellschaft 2003) to state their support of OA. One of the central paragraphs of the declaration is the following:

In order to realize the vision of a global and accessible representation of knowledge, the future Web has to be sustainable, interactive, and transparent. Content and software tools must be openly accessible and compatible (Max-Planck-Gesellschaft 2003).

Open Source

The term Open Source describes software, which source code is publicly accessible, has assigned an open license that allows to reuse and adapt the source code, and reuse the software for own applications. The official definition of the Open Source concept is provided by the Open Source Initiative (OSI) (OSI 2007). According to this definition, the distribution terms of open-source

software must comply with the following criteria:

Free Redistribution The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.

Source Code The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost preferably, downloading via the Internet without charge. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a preprocessor or translator are not allowed.

Derived Works The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.

Integrity of The Author's Source Code The license may restrict source-code from being distributed in modified form only if the license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time. The license must explicitly permit distribution of software built from modified source code. The license may require derived works to carry a different name or version number from the original software.

No Discrimination Against Persons or Groups The license must not discriminate against any person or group of persons.

No Discrimination Against Fields of Endeavor The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, it may not restrict the program from being used in a business, or from being used for genetic research.

Distribution of License The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.

License Must Not Be Specific to a Product The rights attached to the program must not depend on the program's being part of a particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the original software distribution.

License Must Not Restrict Other Software The license must not place restrictions on other software that is distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.

License Must Be Technology-Neutral No provision of the license may be predicated on any individual technology or style of interface.

Open Data

As introduced by the *five star open data* concept in section 2.2.4, Open Data has in this context of Open Science the aspect of reusability of the data, in terms of data format and in terms of license and copyright.

The Open Knowledge Foundation (OKFN) provides an official definition of what Open Data is, it is called the Open Definition (OKFN 2016). This Open Definition sets out principles that define “openness” in relation to data and content. It makes precise the meaning of “open” in the terms “open data” and “open content” (OKFN 2016). The official summary of the Open Definition (OKFN 2016), that entails 23 paragraphs and sub-paragraphs, is the following:

“Open means anyone can freely access, use, modify, and share for any purpose (subject, at most, to requirements that preserve provenance and openness)” (OKFN 2016).

The official most succinct essence of the Open Definition is given as:

“Open data and content can be freely used, modified, and shared by anyone for any purpose”

To facilitate the free reuse of data, it has to be made available in an open format, that can be processed without proprietary software. At best, this format is even an open standard, but this is not mandatory to count as open data.

The aspect of copyright and license is also of importance. The license must allow free reuse and modification of the data, without monetary compensations. Attribution can be imposed, which also fits the good scientific practice of citing a data source, if it was used for a study or application.

2.5.4. Linked Science

Obviously, the process of scientific knowledge sharing and publishing process needs to be improved. One of the main problems is how reproducibility of scientific research can be facilitated by implementing the Open Methodology principle (see section 2.5.3). In this context Kauppinen et al. (2011) proposed the Linked Open Science approach. The approach was originally developed to solve the challenges of an *executable paper*. The main problem to solve is the imprecise description of methods and the lack of access to the data applied in a research study. As a result it takes too much effort and time to reproduce scientific results and to add new knowledge on top of that (Kauppinen et al. 2012).

Linked Science is defined as a combination of Linked Data, Semantic Web and Web standards, open source and Web-based online environments, Cloud Computing, and a machine-understandable technical and legal infrastructure (Kauppinen et al. 2012).

When compared to the definition of Semantic e-Science (see section 2.5.1), is very similar in scope, but Linked Science is focused more on the practical facilitation of reproducible publication, the executable paper, where Semantic e-Science is broader about enabling collaboration on, accessibility to, and discoverability of scientific knowledge.

In Kauppinen et al. (2011) the Linked Open Science approached is summarized in four concise points:

1. publication of scientific data, metadata, results, and provenance information using Linked Data principles,
2. open source and Web-based environments for executing, validating and exploring research,
3. Cloud Computing for efficient and distributed computing, and
4. Creative Commons for the legal infrastructure.

All these four points are implemented in the CRC806-Database e-Science infrastructure presented here.

2.6. Geographical Information Systems

In this section about GIS, a brief introduction to the here relevant aspects of Geographic Information Systems (GIS) are given. Based on a definition of GIS and its components, the concepts of GIScience, SDI, Web-based GIS (WebGIS), as well as OGC standard-based geospatial Web services, known as OpenGIS Web Services (OWS) are introduced. Finally, a point for the importance of openness in geospatial science is made in this context, and the Open Source Geospatial Foundation (OSGeo) is briefly introduced.

Geographic Information Systems (GIS) are relatively complex systems, which assertion is supported by the fact, that there is no simple, short, and concise definition of the concept available that covers all that GIS is about. The shortest sufficient definitions are at least 250 characters and they often are much longer. The short definitions are then also in most cases quite awkward in wording and sense making. That is, why the concept is more broadly introduced and defined in this section. In theory, a GIS can be defined as follows:

GIS are IS to capture, edit, organize, analyze, and visualize geographic data.

In practice, an instance of a GIS combines software with hardware, data, the user, etc. to solve a problem, support a decision, and help to plan. Thus, Longley et al. (2005) defined that a GIS consists of six parts, see figure 2.12. Interestingly, in this definition the network is the central part of a GIS.

Network The central and integral part of GIS today is the network. All functions of a GIS are facilitated, supported, or at least influenced by the Internet.

Software A GIS is apparently based on software, that can be big software suites from a single vendor solving (almost) all needs, or a set of integrated smaller software tools, each for a particular purpose of the given tool chain (workflow).

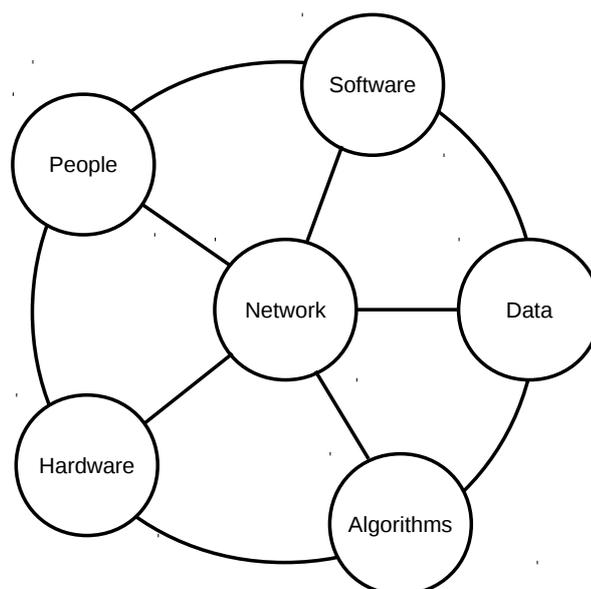


Figure 2.12.: Six parts of a GIS. Changed after: (Longley et al. 2005: 24)

Data Geographic and according attribute data are the basis for any GIS operations, as defined above.

Procedures to manage and organize the GIS are significant to the conduct of successful GIS operations.

Hardware Hardware is, of course, very important too; it ranges from mobile devices to large-scale computing clusters, and from local out-of-the-box solutions to cloud based distributed server infrastructure.

People a GIS is useless without the people who design, program, and maintain it, supply it with data, and interpret its results.

According to several sources (Longley et al. 2005; Goodchild 1992; Bartelme 2005), the first GIS was the Canada Geographic Information System (CGIS), designed and developed by Tomlinson et al. (1976) and his team in the mid-1960s as a computerized map measuring system. Many technical developments in GIS and also the Global Positioning System (GPS) originated in the Cold War and were invented to solve military needs, such as missile targeting, automated map making, and navigation (Longley et al. 2005). The modern history of GIS dates from the early 1980s, when the price of sufficiently powerful computers fell below a critical threshold (Longley et al. 2005).

The Internet is increasingly integrated into many aspects of GIS use, and the days of standalone GIS are mostly over (Longley et al. 2005).

Today, at least in the developed first world, every person has almost daily contact with GIS software solutions, such as car navigation systems, weather maps, smartphone maps and GPS applications, as well as LBS applications that use the position of a device for presenting accord-

ing informations, such as restaurant guides, or travel guides.

2.6.1. GIScience

The term GIScience was coined by Goodchild (1992). In this work, the author argued that academic questions about GIS approaches, designs, data, and methods are important, and that their systematic study constituted a science in its own right (Longley et al. 2005).

GIScience deals with questions of representation, models, accuracy, visualizations, and analysis of geographic data and problems.

Disciplines that are concerned with GIScience are, foremost, geography, cartography, remote sensing, geodesy, surveying, photogrammetry computer visualization and image processing, and of course computer science and information sciences.

2.6.2. SDI and WebGIS

Since the Spatial Data Infrastructure (SDI) concept was first introduced in the early 1990s, its definition has changed a bit (OGC 2015a). Originally SDI is defined as:

SDI is a collection of technologies, policies, and institutional arrangements that facilitate the availability of and access to spatial data (GSDI 2009).

Budhatoki et al. (2007: 11) claims, that there is no consistent definition of the SDI concept. However, the technical distributed and network (i.e. Internet) and economic and social collaboration aspect is what distinguishes a SDI from a traditional GIS. Where in GIS the emphasis is on the tools and algorithms for spatial analysis and map production, in SDI the focus is mainly on data retrieval, distribution, and access over a network, as well as standardized interfaces for data interoperability.

Another important part, and of most interest to this study, is the GIS data management and spatial data handling and organization aspect of the SDI concept (Bareth et al. 2010; Curdt et al. 2015; Willmes et al. 2014e). There are many examples available for applying SDI concepts for this purpose (Curdt et al. 2010; Curdt et al. 2011; Curdt et al. 2012; Curdt 2014b; Bareth 2009; Willmes et al. 2007; Willmes et al. 2008; Willmes et al. 2010).

The OGC SDI technologies, known as the OWS standards, are based on well-defined data models and schemas (OGC 2015a). Thus, data managed implementing the OWS specifications (see next section), is well-defined. In this work, an Open Source SDI implementation is applied (see sections: 3.3.2, 5.2.3 and chapter 7) for taking care of spatial data management according to well defined metadata standards and interoperability concerns.

The concept of WebGIS describes Web based GIS applications. In a more narrow and technical definition, it is a GIS application that uses OWS for accessing and visualizing data. The use of the WWW to give access to maps dates from 1993 (Longley et al. 2005). For displaying the geospatial datasets of the CRC806-Database presented here through the Web application in a Web browser

environment, an OpenLayers (OpenLayers Contributors 2015) based WebGIS was implemented, see sections 7.3 and 10.1.

Geospatial Web Services

One important feature of SOA interoperability is standardization of service interfaces. There are several Organizations, Institutions, and private-sector companies that have a stake in standardization of geospatial Web services.

However, the OGC is the de facto standard for standardization in the geospatial Web realm. Formally, the OGC is an international industry consortium of >500 companies, government agencies, and universities participating in a consensus process to develop publicly available interface standards. The OGC and W3C cooperate in an endeavor called "*Spatial Data on the Web Working Group*", to clarify and formalize the relevant standards landscape for spatial on the Web (OGC 2015b). At the time of writing this work, the OGC maintains 47 standards, of which only the few most relevant specifications for the infrastructure presented here will be introduced.

Web Map Service (WMS) The Web Map Service (WMS) (de la Beaujardiere 2006) is a standard specification for providing georeferenced map images, or visualizations of geospatial data, via HTTP protocol in an SOA context. The specification has been and is developed and maintained by the OGC since 1999. The WMS is presumably the most popular and widest distributed standard of the OGC. The standard is supported by almost any desktop GIS available and is facilitated in almost any WebGIS implementation for geospatial data visualizations.

Web Feature Service (WFS) The Web Feature Service (WFS) standard (Vretanos 2010) defines an interface for the exchange of geographic feature (vector) data. The interface allows for a number of different formats as the answer to WFS request, thus it servers in the INSPIRE context as the feature download service. Originally, it was specified for delivering Geography Markup Language (GML) Simple Feature Profile (Vretanos 2005) only, but later, other formats, such as JSON, ESRI Shapefile, Keyhole Markup Language (KML), etc. were added.

The WFS-T (WFS-Transactional) extension adds functionality for editing and deleting the data base behind a WFS, based on user roles.

WFS is supported significantly less in desktop GIS packages, but the major software packages from esri or QGIS Desktop support WFS, most of the smaller GIS and popular consumer-grade software products such as Goggle Earth do not support the standard either. Also, in WebGIS realm, for example, not all Web mapping clients support WFS. While OpenLayers (OpenLayers Contributors 2015) does support WFS, Leaflet (Agafonkin 2015) does not.

Web Coverage Service (WCS) The Web Coverage Service (WCS) standard defines an interface for multidimensional coverage data exchange over the Internet (Baumann 2012). The geographic coverage data are suitable as input for scientific models. WCS format encodings allow the delivery of coverages in various data formats, such as GML, GeoTIFF, HDF-EOS, or NITF.

The WCS-T extension (t stands for transactional) allows for client-side editing and updating of the service data, based on user roles.

The WCS standard is supported by two major desktop GIS packages, esri ArcGIS and QGIS. It is not widely used in WebGIS context because it is not a service for visualizations. In the 3D WebGIS realm, the standard is partly supported as a container for Digital Elevation Model (DEM)s, for example in deegree WPVS (Willmes et al. 2010) or in the Cesium JavaScript framework (Analytical Graphics, Inc. 2015).

Catalog Service Web (CSW) The Catalog Service for the Web (CSW) interface is a standard for exposing a catalog of geospatial metadata on the Internet (Nebert et al. 2007). It defines interfaces for data lookup, data exchange, and also transactional (update, edit) access to records. Typically, the records include Dublin Core, ISO 19139 or FGDC metadata, encoded in UTF-8 characters. Each record must contain certain core fields including: Title, Format, Type (e.g. Dataset, DatasetCollection or Service), BoundingBox (a rectangle of interest, expressed in latitude and longitude), Coordinate Reference System, and Association (a link to another metadata record).

2.6.3. OpenSource Software for Geospatial

Open Source Software (OSS) for geospatial applications, gain in distribution and acceptance in the private sector as well as in the public sector and in the education sector.

All scientific research is based on absolutely transparent reproducibility, which is not given if there is no possibility to look into the source code of the software that is applied to conduct the analysis on which the research findings are based. Therefore, proprietary software which does not disclose the source code can also not be used for valid scientific research (Christl 2014).

The OSGeo is a non-profit organization to facilitate an umbrella for the Free and Open Source Software for Geospatial (FOSS4G) community. The foundation facilitates mainly communication through several mailing lists and other platforms for the community; additionally, it hosts the yearly international FOSS4G Conferences, as well as regional (FOSS4G-EU, FOSS4G-Asia, etc) and local events, such as the German FOSSGIS conference. OSGeo is also involved in the academic and educational realm; it runs an Open Access Journal, the OSGeo Journal⁵, as well as the Geo4All⁶ initiative for organizing Open Source Labs and Education at Universities. Openness is a key factor in standards development. Education thrives on Open Access, and Open Data is key to innovative work, evolving businesses, e-government, and inclusion (Christl 2014). The systems developed for and presented in this thesis, are almost exclusively implemented using OSS.

⁵<https://journal.osgeo.org/>, accessed: 2016-01-06.

⁶<http://www.geoforall.org/>, accessed: 2016-01-06.

Part II.

Design, Methods and Technology

3. Demands and Design

In this chapter, the overall design and system architecture of the CRC806-Database will be explained. First, the demands of the project and its members, as well as the requirements of the funding agency are considered. Based on this demands the basic infrastructure features and components are identified and designed. The development of the CRC806-Database is and was an *organic process*, meaning that the infrastructure grew and will further grow with time. From the first prototype of the current infrastructure and its subsystems, many iterative development steps took place, and will go on until the end of the project funding. This organic development process is mostly on the part of the collaborative knowledgebase and its data model (see sections 3.3.3 and 5.2.8), but the RDM infrastructure also grew organically, from the aspect of features and functionality, as well as from the aspects of robustness, security, stability and its organization. Some may argue, that organically grown sounds like no planning. A good balance between top-down beforehand planing and defined structure, and the bottom-up flexibility of adding functionality and features along the way is of key importance to successfully develop and implement IT infrastructure (see also section 12.1.1 for a discussion of this aspect). But the most important reason for this approach is, that the feedback from the scientists working with the infrastructure always brings up demands for new features or improvements. Collaboration with the project participants is vital for success of an RDM infrastructure (Curdt et al. 2015), it is also explained through the overall applied prototyping development approach (see section 4.5).

3.1. Demands for e-Science infrastructure

This section introduces the different demands for the development of an e-Science infrastructure for the CRC 806 project. There are three domains of demands identified. First, the funders demands, then the demands from within the CRC 806 and general demands for facilitating collaborative working on databases and data infrastructures. Each of the demands are addressed in the following of this section, with links to the according section containing more detail on how these demands are implemented.

The identified demands are the constraints on which the overall infrastructure, as described in this thesis, is designed and implemented.

As defined in section (2.5), RDM is a subset of e-Science. In this section, the overall demands for semantic e-Science including RDM infrastructure in the context of the CRC 806 are described.

The necessity for managing of research data in a professional way is beyond discussion (Bareth et al. 2013; Curdt 2014b). As described above, in times of the scientific data deluge, science

2.0, open access the 4th paradigm and similar developments (see section 2.5), it is crucial to find solutions for the problem of making research data available, and even more importantly accessible and reusable, to anyone interested in them. The problem is not only technical, it is to a significant extent social and even cultural. Addressing this demands, and building good solutions has many facets and layers. To address the demands, it is helpful to sort out and precisely define each demand, as given in the following descriptions:

Data storage The demand for secure long term data storage and preservation is a key demand from the project funder (DFG). The guidelines demand storage and accessibility of the data for at least 10 years after project funding ended.

Data exchange The facilitation of data exchange, through possibilities of publication, e.g. persistence and citability, as well issues of interoperability (see also the data reuse demand).

Data context Annotation and formal description of the data is crucial to contextualize the data, to facilitate data exchange and reuse. The spatio-temporal annotation is of key importance to this aspect.

Data reuse Enabling reuse of the provided data includes techniques for interoperability and accessibility, as well as context and also copyright and license issues.

Usability It is important, to provide interfaces, that are usable by applying mostly common sense and without too specialized training. Also providing standardized interfaces, like OGC Services and REST API interfaces, help to increase the usability and accessibility of the infrastructure.

More general societal demands can and should also be considered, when setting up a publicly funded e-Science infrastructure, the most relevant are summarized in the following list:

Technological demands are for a user friendly interfaces, that include clear organization of the contents, standardized access interfaces for fostering interoperability.

Social and Cultural demands are, accessibility of content, abiding to good scientific practice, reliability of the system, as well as security of the data, and the possibility for the data publishing user to opt for, or against, access constraints.

Policy and Law demands, are for clear rules for reuse, e.g. licensing and copyright. It should be possible for the data publishing user to opt for, or against, Open Data and Open Access publication.

Further demands come from the science funders, as described in the next section (3.1.2), from the project partners (see section 3.1.3), and demands specific to the infrastructure design (see 3.1.4).

3.1.1. RDM specific demands

RDM is a comparably broad field, with many diverse valid approaches for implementation. Demands for a RDM infrastructure are mostly originated from according policies and specific

project or institutional demands as described further in this chapter according to the CRC806-Database implementation. The table 3.1 depicts 21 specific demands for RDM infrastructure, including an according suggestion for its implementation. This list also gives a good overview about the overall demands for a state-of-the-art RDM infrastructure.

Table 3.1.: Specific demands for RDM infrastructure implementations.

#	Infrastructure demand	Implementation
1	Awareness of research regulatory environment	Code of good research practice
2	Access to searchable data discovery services	Data catalog
3	Ability to describe analogue holdings and link these to publications and other outputs	
4	Data shared with national and funder specific research outputs registries	
5	Data catalogue to enable searching via external data registries	Licensing, Copyright, Law and Policy
6	Data access policies and procedures in place	
7	Procedures for managing copyright and IPR	
8	Advice on statements to be included publications and metadata records	
9	Identifying data assets for retention, risks, scoping access requirements	
10	Support for defining and monitoring retention periods	Active data storage
11	Access to sufficient data storage	
12	IR/Data repository collects statistics relating to access requests	Data repository
13	Sustainable data repository for storing, searching and accessing locally held data	
14	Secure working area for sensitive data	Secure data access
15	Allocation of a persistent identifier for research data	DOI minting service or related support
16	Departmental / Institutional policies relating to RDM and data management planning	Departmental / Institutional RDM policy or aspirational statement
17	Awareness of OA policy requirements	Open access policy
18	Awareness of funders' policies	RDM support service
19	Support and guidance for data management planning	
20	Support for data description and citation	
21	Training, guidance and support for staff and students	

How and if the demands listed in table 3.1 are implemented in the CRC806-Database is concludingly discussed in section 12.2.

3.1.2. Demands by DFG for INF-Projects

As introduced above, the CRC 806 is a DFG funded project, and the DFG, like most funders, has a policies and certain demands according to RDM, for projects funded by them.

Since 2007, the DFG offers the possibility for CRCs to propose a sub-project to take care of implementing the DFG RDM demands. According to the DFG proposal guideline bulletin 60.06, it is even expected that a CRC proposes a project section focusing on sustainable data storage and management, and archiving in cooperation with libraries or computing centres (Bareth et al. 2013: 365). A CRC INF-Project (see section 1.4) has on average about one full Scientific Staff (PostDoc) and/or one or two PhD students (50-75% Scientific Staff), plus some resources for Student Assistants (SHK) and funding for Hardware and Software as well as for purchasing commercial Data in some cases, depending on the particular project proposal. This approach results in many smaller and diverse RDM projects and resulting infrastructure implementations, creating a comparably diverse and heterogeneous RDM landscape in Germany. From the DFG point of view, the strategy of funding many smaller RDM projects, aims to foster crystallizing the best solutions for the current case (Engelhardt 2013). In the project proposal guidelines for CRCs (DFG 2012), the DFG defines the following demands for INF-Projects in the context of CRCs:

1. Development of a database to centrally store the research data produced by the CRC,
2. Development of techniques and methods of data handling (care, allotment, referencing, and linking) the data,
3. Support and facilitate the reuse of the data, through enabling interoperability with external repositories and data bases,
4. Development of a VRE, to facilitate collaboration and reuse of the data within the CRC,
5. Facilitation of "interoperable components", such as Wikis, project management software or Version Control System (VCS),
6. Adaptation and implementation of state of the art technology, for electronic publication, identity management or virtual organizations.

Furthermore, it is expected, that the INF-Projects cooperate with local infrastructure providers, such as the university computing centres and the libraries (Engelhardt 2013).

As described in detail in the further course of this study, all of the 6 above stated DFG demands, as well as the cooperation with local infrastructures were realized successfully. Demand #1, the development of a database to centrally store the research data, is implemented by the CRC806-Database data catalogue application (see chapter 6 and 10). Demand #2, development of techniques and methods to handle the data is implemented in the data catalogue, as well as in the SDI (see chapter 7 and 10) and in the KB (see chapter 8 and 11). Demand #3, facilitate the reuse of data, is realized through several measures, like the implementation of DOI's for better citation and referencing of the data (see 6.1.4), through the implementation of OWS interoperable interfaces for geodata in the SDI (see chapter 7), as well as through the open data API provided by the Comprehensive Knowledge Archive Network (CKAN) backend of the data catalogue (see 9.7.1). Demand #4, development of VRE, is mainly delivered through the KB application (see chapter 8), but also through the whole CRC806-Database infrastructure, that can, to some extent, also be defined as a VRE. Demand #5, facilitation of "interoperable components",

is implemented through the GitLab code repository (see 5.2.5), that also includes a bug tracker, as well as of course through the SMW based KB, that is build upon the MediaWiki (MW) Wiki software. Demand #6, technology for data publication and identity management, is implemented through the RDM infrastructure, in form of the members directory, and through the publications DB (see 6.2). These implementations of the demands are then concludingly discussed in section 12.2.2.

3.1.3. Demands defined in the project proposal

The summarizing paragraph of the first funding Z2 INF-project proposal (Bareth 2009) of the CRC 806 describes well, what the here presented infrastructure is about:

Data storage and exchange in interdisciplinary research projects, that focus on spatially distributed field data collection in an organized framework are key issues. The overall and sustainable success of such projects depends on the well organized data management and data exchange of/for all projects. Therefore, the aim of this project is to solve the main problems of data storage and exchange within an interdisciplinary project by using a complex spatial database that allows the management of heterogeneous spatial and attribute data. Besides geographical and attribute data, e.g. multi-scaled field maps, this database includes metadata, literature, file management, project staff and publication, as well as picture and video data and provides as a basis for the external and internal presentations of partial results and conclusions. A special focus is the sustainable use of all gathered data within the proposed CRC also after the project is finished (Bareth et al. 2009).

This paragraph primarily sets the demands of the project proposal. Which is, providing the infrastructure for centrally storing and managing the data as the main duty of the Z2 Project. The Z2 Project "Data Management and Data Services" is defined in the 2nd Project proposal (Bareth et al. 2013) of 2013 as follows:

As data storage and exchange is a key issue in interdisciplinary research projects, data management and presentation will be part of the centralized assignments. Besides geographical and attribute data, the database will include metadata, literature, file management, project staff and publications, as well as picture and video data. Furthermore, this includes presentation of spatial and thematic project data on standardized maps and the generation of on demand map sets for field investigations. GIS and remote sensing will be essential tools (Bareth et al. 2013).

An overall refined definition of the purpose of the data management infrastructure of the CRC 806 is given in the introductory part of the CRC 806 proposal (Schuck et al. 2013) as follows:

The main purpose of an Information Infrastructure Project in a Collaborative Research Centre is the management of relevant data collected by the CRC with the aim of

enabling systematic and long-term use of such data. Within the CRC 806 project Z2, headed by Georg Bareth and Olaf Bubenzler, is the main and centralized Information Infrastructure Project (Schuck et al. 2013: 24).

In the first funding proposal (Bareth et al. 2009) for the project Z2: "Data Management and Data Services" were defined as follows:

- overall aim is to implement and maintain a project data base.
- Interfaces to well-established DBs e.g.. PANGEA and NESPOS shall be established
- Integration CalPal tool
- A WebGIS shall be established

These demands, that were formulated in the project proposals are almost all implemented with highest priority. Though, the interfaces to NESPOS (Bradtmöller et al. 2010) and PANGEA (Grobe et al. 2015), were not implemented as originally proposed. Also the cooperation with CalPal (Weninger et al. 2010) was not implemented as it was planned. The manifold reasons for not exactly realizing the interfaces as proposed in the project proposals are addressed in detail in section 12.7.

A major concern, and as such also a major demand for the RDM infrastructure, was the possibility to implement detailed access restriction mechanisms. It was demanded, that data resources could be restricted to only a defined group or even individuals. See sections 6.1, 6.1.5, 6.3 and 9.2.4 for details of the implementation of fine grained access rights and user rights management.

3.1.4. Demands for collaborative Knowledgebase

Besides the demands by the project funders for collaborative infrastructures like VRE's and KB's, as described in section 3.1.2, further demands for a collaborative KB are prevalent, though they are not formally defined by the funders or the project partners.

Some researchers of the project were looking for a database application to create an overview of their actual research data, from their working domain and from the literature as well as from data sources published online. Until now, all researchers, or in some cases projects, maintain their working data in local collections. These collections are mostly consisting of spreadsheets (MS Excel) or simple relational (MS Access) database applications. The exchange of data always is a problem, because any spreadsheet and database is maintaining its own data schema. To solve this problem, a web based application to collaboratively collect, share, query and analyse very heterogeneous data, was requested by project members.

An other fact, that makes the endeavor to create an integrated KB for the CRC 806 ambitious, is the heterogeneity of the data domains of discourse. And of course, the heterogeneity within the domains and its sub-domains. Thankfully the spatial annotation of an archaeological or geoscientific artefact is sufficiently clear, in the case of temporal annotation it is much less clear. And if you look at the integration layer of cultural or environmental classifications and annotations, we find ourself in mere discourse. Thus, the development of an integrated data model can be seen as the seek for the smallest valid denominator.

Thus, the KB should be primarily an application to collaboratively work on and share data, information and knowledge that is not officially published through the CRC806-Database RDM infrastructure. These data and information items are mostly externally published resources and internally produced minor data sets, that are not yet published combined with the published data to build an integrated KB. Through the applied wiki approach any project participant can edit and extend the KB, which implements the VRE demand by the DFG.

3.2. Research Data Management

In this section, the design of the RDM implementation within the CRC 806 will be described. The design of the infrastructure is based on the demands as outlined in the section 3.1 above. First, we will clarify with what kinds, types and domains of data the CRC 806 is mainly concerned. Followed by the definition of methods applied to the research data in the course of archiving, publishing and curation of the CRC 806 data. This implementation and definition of methods applied to the research data will be described along a custom developed Research Data Lifecycle model, see section 3.2.3.

3.2.1. CRC 806 data domains

Because the CRC 806 is an interdisciplinary project, the data basis is very heterogeneous. The data domains diversify along four classes. The first class is constructed by the scientific domains or disciplines, with their own subject areas, classifications, schemas and vocabularies. The second class is the data format and type, e.g. spatial data, tables and spreadsheets, images, textual informations, etc.. The third class are the data types, like primary and secondary datasets that are created by the CRC 806 and its members, and data that is integrated to facilitate the work in the project. Finally, a classification of information types constructs the fourth organizational axis.

Scientific disciplines

The CRC 806 data from the disciplines of Archaeology, Anthropology, Geosciences, and Geography, additionally data and literature from neighboring domains are concerned to some extent within the project. All these different scientific domains have own research methodologies and conceptual designs and approaches. This consequently results in different subjects of concern and how to conduct research. These factors lead to very different data formats, data types and information types produced and demanded by each of the disciplines.

Data format

Data formats range from bibliographic data, over textual informations, like reports, transcripts, or presentations, to media data like video, photo and audio, to data formats for handling statistical and measured numerical informations, like spreadsheets and structured file formats, to

complete databases, that can contain all of the before mentioned formats. An important data format of the CRC 806 are spatial or geodata, because of its intrinsic information integrating factor, see section 2.6.

Data types

Data types, in this context, are understood as the distinction between primary and secondary data. Primary data are empirical research data, that are directly measured. This can be measurements of sediment properties in drill cores, archaeological artefacts and objects, a satellite image or an interview recording. Secondary data are data that is derived, by conducting analyses, computations and derivations of any kind from the primary or other secondary data.

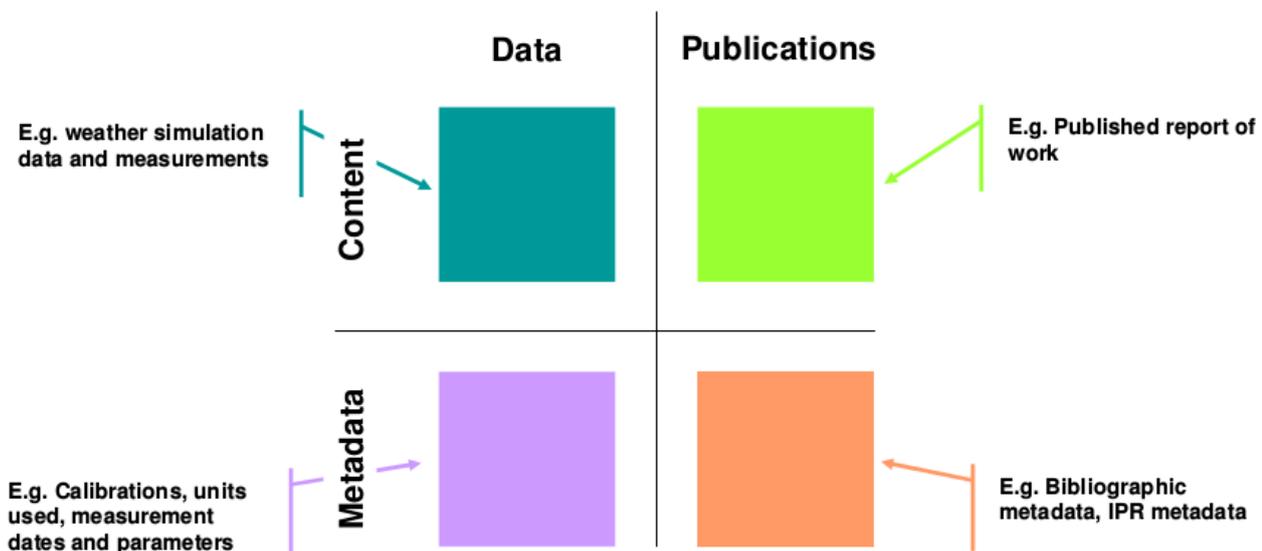


Figure 3.1.: Characterization of Information Types in repositories. Source: (European Commission 2006: 16)

Information types

It is helpful to define and characterize information types held in repositories, see figure 3.1. The information in repositories is usually differentiated into different, well-defined items; synonyms for “item” are record or entry (European Commission 2006). A basic distinction is between publications and data items. Publications are those, whose content is published in some sort. This can be published articles, reports, pre-prints, theses, patents, books and similar. Data items are all other types of content, such as datasets and databases, images, videos, simulation, computation and analysis results, and so on. The second distinction is between content and metadata, see section 2.2.1 for a description of the concept of metadata, content then, is all other information stored in a repository.

It is clear, that this characterization is very high level, and that there are a lot more distinctions between information types in a repository are possible. These other distinctions include, classification and organizational schemes, thesauri, ontologies, gazetteers, indexes, catalogs and more.

3.2.2. Research Data Lifecycle

In this section, the ideal Research Data Lifecycle (RDL) of the CRC 806 is described. Ideal, because it is clear, that this model is a simplification of reality and thus can not entail all details and cover all probable cases.

As introduced in section 2.4.2, the RDL is the center piece of any RDM design. The CRC806-Database RDL is given in figure 3.2, consisting of six major stages; creating, processing, analyzing, preserving, publication and re-use of data, as described in more detail in the following of this section.

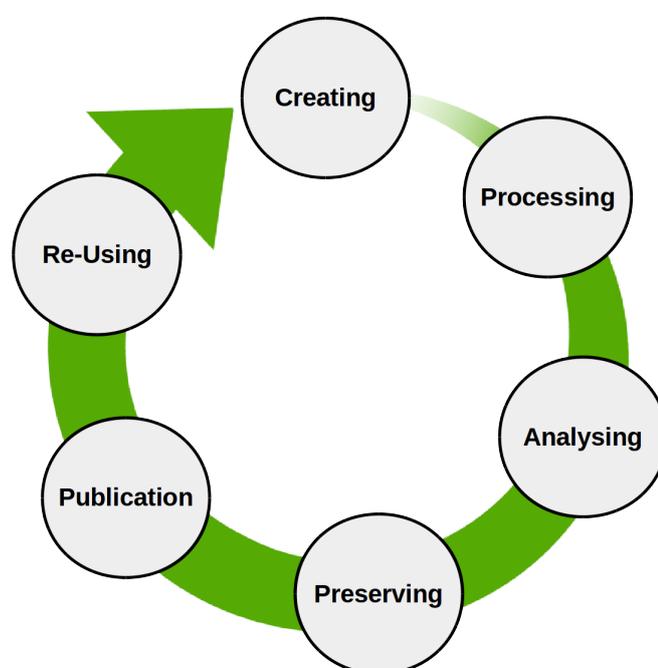


Figure 3.2.: CRC806-Database Research Data Lifecycle. Source: own work.

Creating data

The RDL is based on the creation of new data. This is the most complex and least predictable step in the whole research lifecycle. Some measures can be taken to facilitate the best possible outcome of the research. This starts in the design phase of the research and includes a plan for data management, including thoughts on data formats, data storage, backup, etc. In some cases of collaboration with external partners, a formal consent for data sharing, before any data is created is a good idea. Another very important task prior to creating new data is research about

existing data that could be reused for the given research. This is followed by an according data collection (field work, excavations, experiment, observe, measure, simulate) and a temporary creation of a data base holding the primary data. Finally, metadata should be recorded and created straight away with capturing the primary data.

Processing data

The second step in the RDL is data processing. This includes format transitions, digitalization, entering data into a data base from notes or non digital observations and publications, transcription, translating, and so on. In the geospatial domain, this can include some complex task like interpolation, converting of vector data into raster data or the other way around from raster into vector format. These tasks, can produce large amounts of secondary data, that needs to be managed, stored and backed up. In some case also measures for data security in terms of access restrictions and encryption are conducted.

Analyszing data

The next step is the analysis of the data, that includes interpretation and derivation of the data, conduction of statistics, classifications and further computations. In the geospatial domain this is the key task including comprehensive GIS analysis. The production of research outputs, like maps, graphs and data tables are also part of this step. And the preparation of the data for the following steps of preservation and publication is also situated in this stage, including transformation of the results into open data formats for facilitating re-use of the data.

Preserving data

Data preservation is a key task to allow for successful data management. This task includes to make sure, that the data is provided in a standardized – at best also open – format. In the case of the CRC806-Database, the backup and secure long term storage and archive is facilitated by the Regionales RechenZentrum Köln (RRZK), as described in section 5.1. Another important task of this step is to create or finalize the metadata description of the preserved datasets.

Data publication

Publication of data is facilitated through the CRC806-Database, by the possibility to assign DOI for datasets and resources, as described in section 2.4.3. In the frame of e-Science and the Open Science approach, as described in section 2.5.3, this step of the CRC806-Database Research Data Lifecycle is paramount to the overall aim of the infrastructure. Important sub-tasks that need to be considered while publishing data are the access rights and its control, as described in the sections 6.1, 6.1.5, 6.3 and 9.2.4. As well as the assignment of a license to define the copyright, as described in section 3.5.

Another possibility is to publish the data in third party repositories, like PANGEA or NESPOS, or as supplemental material of a traditional journal publication.

Re-using data

Re-use of the data is directly dependent on the previous step of data publication. Most important are the open license, an open data format and of course the open access to the data, as well as the possibility for proper citation. All these demands are facilitated by the CRC806-Database infrastructure. This enables possibilities for collaborations and follow-up research, research reviews, as well as teaching and learning based on the previously published research. It increases the citations of the according publications and thus also increases the impact of the research, the involved scientists and the overall CRC 806 project.

3.2.3. Repository Lifecycle

As introduced in section 2.4.2, the research data life cycle is an established model to describe the process of RDM during a research project. The CRC806-Database RDM system implements the whole life cycle of research data as described in that section. Additionally, the CRC806-Database implements what I propose as a project Repository Life Cycle (RLC), as graphically depicted in figure 3.3.

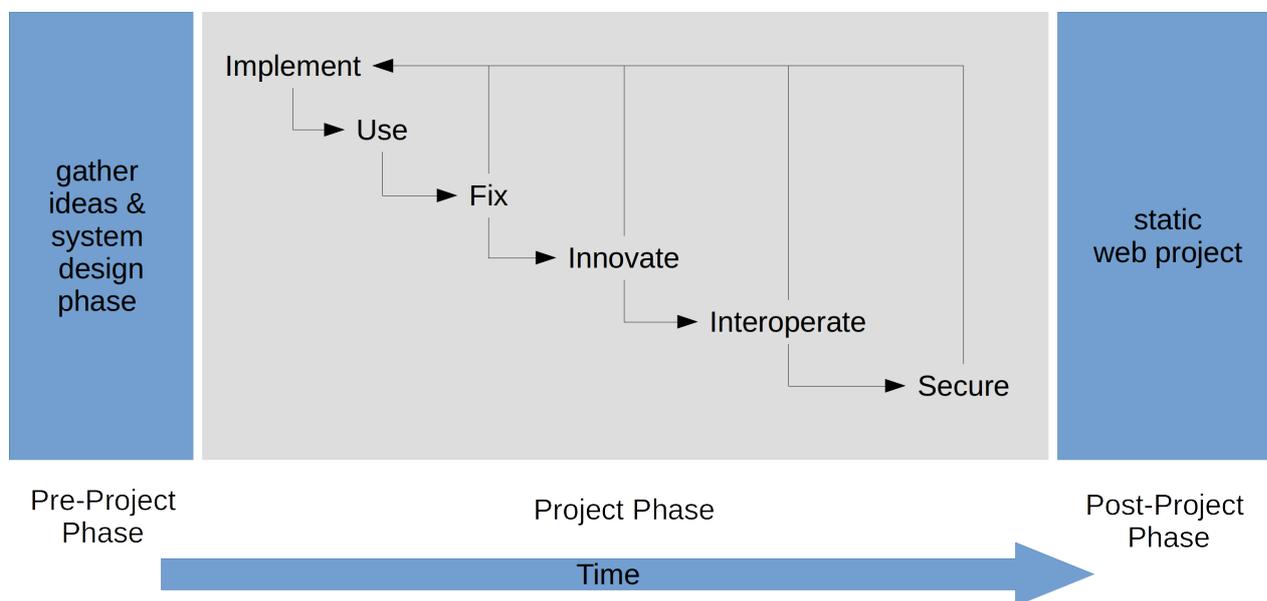


Figure 3.3.: Project Repository Lifecycle Model. Source: Own work.

Research projects are in most of the cases funded for a limited time, and if the project implements an own RDM repository, as it is the case for the CRC806-Database and for DFG funded

CRC's in general (see section 2.4.4), some common patterns emerge, that are described in the following.

Pre project phase

In the pre project phase, gathering of ideas, research about related work and infrastructures and project proposal writing are the main tasks. This phase is about designing the initial layout and aims of the infrastructure, that is then formulated in the project proposal.

Project phase

In the first phase of the project, the system is implemented, as developed in the project proposal. During this initial development, or at the latest, after a first version of the repository is finished, the system will be tested and used, during this process, almost certainly errors and shortcomings will be detected, that needs to be fixed. This will start a feedback cycle, similar to the prototyping approach (see section 4.5), until most errors are identified and fixed. At the latest then, innovation, respectively requests and ideas for additional functionality will emerge. To develop these additional features will restart the initial development cycle. At the latest, when it seems that the repository is feature complete, it should be taken care of implementing and improving interoperability, ideally according to standards. Finally, the security of the system should be hardened. This includes, testing the whole system for vulnerabilities and fixing them.

Post project phase

Because the project will not be maintained by the project funded staff, after the project has ended, strategies for keeping the system online and working need to be developed and implemented. This can include many different measures, that are very dependent on the given infrastructure implementation and the technology used. What always holds is, that the system needs to be reduced in complexity, this means to get rid of most dynamic functionality. The here presented system, for example we be transformed into a static website, without server side scripting and no Relational database management system (RDBMS), no user login, etc., just plain HTML. This solves about 99% of all possible vulnerabilities of a web based system. Access to not publicly archived data, will be facilitated, by an email form, to contact the dataset owner for access to the data.

3.3. System architecture

In this section, an overview of the system architecture for the CRC806-Database e-Science infrastructure is given. The infrastructure is organized into three main systems, the CRC806-RDM, the CRC806-SDI and the CRC806-KB infrastructure. While the CRC806-RDM infrastructure is designed with a focus on implementing the policy and funders demands for RDM in CRCs, the CRC806-SDI is designed to provide solutions for geospatial data management and handling, and

the CRC806-KB infrastructure is designed to enhance and facilitate the collaborative research within the project.

3.3.1. CRC806-RDM

The system architecture of the RDM infrastructure consists of three main components:

- Data catalog and repository
- News and Members
- Publication database

The three components are implemented as custom developed Typo3 extensions, as further described in chapter 6. The *Extbase & Fluid* based Typo3 extensions implements the CKAN Action API functionality to integrate CKAN into the CRC806-Database web application frontend. Furthermore, it handles the file management for resources on a redundantly backed-up file system, provided by the RRZK (Willmes et al. 2014e).

The user has the possibility to access the research data through the CKAN API (see figure 3.4). In addition, a CSW interface is offered to browse and access the OGC compliant geospatial web services (WMS, WFS, WCS) of the SDI, as described in section 3.3.2. This enables the researchers and interested public to access the project database data in an automated/scripted way. In the following of this section, the three main components of the CRC806-RDM system, as given above, are described accordingly.

Data catalog and repository

The data repository is designed to offer tools and capabilities to handle data in all stages of the Research Data Lifecycle model, as described in section 2.4.2. The Research Data Lifecycle model aims to model solutions for the demands of the funders and the project partners as discussed above. The main demands are to secure long term storage and availability of project data, at least 10 years after project termination, as well to facilitate online publication of datasets and research results. This is where the above proposed Repository Lifecycle model takes effect, see section 3.2.3. To secure the long term access and preservation of the data deposited and published via the repository, redundantly secured and backed up file systems, provided by the RRZK are employed, see section 6.1.1. The metadata catalog, as well as the search and discovery backend and application of the data repository are facilitated by a CKAN instance, see section 6.1. The UI is implemented in an Typo3 *Extbase & Fluid* extension, see section 6.1.5.

News and Members

The two applications of *News* and *Members*, are primarily of administrative nature. The News applications was developed as a Typo3 extension to provide a custom application, to handle and

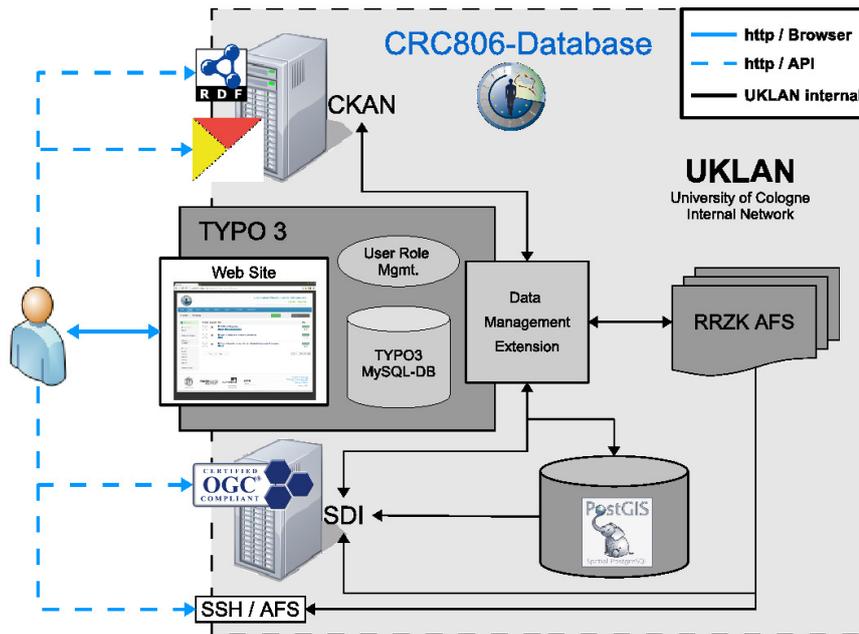


Figure 3.4.: System architecture of the CRC 806 Research Data Management infrastructure. Source: Willmes et al. (2014e).

show news posts, to have lists of the currently newest data sets, geo-datasets and publications prominently displayed on the frontpage, as well as the display of visitor statistics by including the PIWIK (PIWIK Contributors 2014) map widget. The Members directory is based on the Typo3 user management, but extended with custom functionality to integrate with the data catalogue, publication database and also with the maps SDI frontend.

Publication database

To be able to present the publication output of the CRC 806 project to the public, and also make it available for internal communication and archival, a publication data base was demanded. The publication database is also implemented using the CKAN instance of the data catalogue as metadata backend, with an additional metadata schema for handling bibliographic records, see section 6.2.1. A simple categorization in an internal metadata field handles the distinction between research data and publication record. The database has an own frontend called *Publications* on the main website, including sophisticated filter, search and browsing options, as well as features like BibTeX export. To be able to display the publications on the main CRC 806 website¹, a plug-in to interface the Joomla Content Management System (CMS)(Open Source Matters, Inc. 2015), that is the basis of the main website, was developed, see section 6.2.3.

¹<http://www.sfb806.de>

3.3.2. CRC806-SDI

To deliver a catalogue, repository and web based services for geospatial data, a GeoNode (GeoNode Contributors 2014) based SDI is implemented as part of the CRC806-Database (see figure 3.4). Geospatial data is useful to support almost any research question within the CRC 806, it additionally facilitates data integration along a spatial and temporal context. The SDI is based on a GeoNode backend, see section 7.1, and is integrated into the CRC806-Database web application by a custom developed Typo3 extension, see section 7.1.2. The Typo3 frontend application facilitates discover, access and visualization, including WebGIS capabilities, of the geodata provided by GeoNode. Additionally, it provides OGC compatible OWS interfaces, for integration in other applications, like Desktop GIS.

3.3.3. CRC806-KB

The CRC806-KB system aims to support and facilitate research directly in the sense of an VRE in the present context of a semantic e-Science infrastructure. A key focus are tools and interfaces to query and discover spatio-temporal patterns, not directly visible before.

The system is named as a KB, because it aims to combine a lot of information and data, to model a certain knowledge on top of the data collection. An example of the kind of knowledge maintained in the system is, that through the system it is visible, what research was conducted according a certain theme, archaeological setting and location. This knowledge is modeled in form of according queries on the data, information and knowledge collection. Another justification for the term KB is, that this system is sort of a database containing information about databases and datasets, which on their own already contain information about the data of these databases and datasets.

Developing an integrated research database for a large interdisciplinary research project is a complex, ambitious and laborious task. Nonetheless, this KB infrastructure aims to present an approach to solve this problem, see figure 3.5.

The KB has primarily an project internal scope. Meaning only project participants can edit the KB. The system allows to store all sorts of data, information and knowledge about published and unpublished resources. Data, information and knowledge is gathered by the project participants, by editing the Wiki based frontend in a collaborative, and thus sort of peer-reviewed approach. The resulting KB can then be queried through complex spatio-temporal queries, such as "show all archaeological sites, with artefacts classified as Aurignacien culture and located in northern Spain" for example. This query will yield a certain result set, that can be directly visualized on a web based map, or shown in form of a table and even exported in many different formats, such as Excel, XML or JSON for example. The system is technically based on SMW, as described in section 5.2.7. On this basis, an infrastructure, that integrates available, already published, datasets and databases of interest to the research questions of the CRC 806, allows to enter and handle manually entered data from available publications into defined forms (Schemas based) and to build up a bibliographic data base of related relevant research publications, that all can be

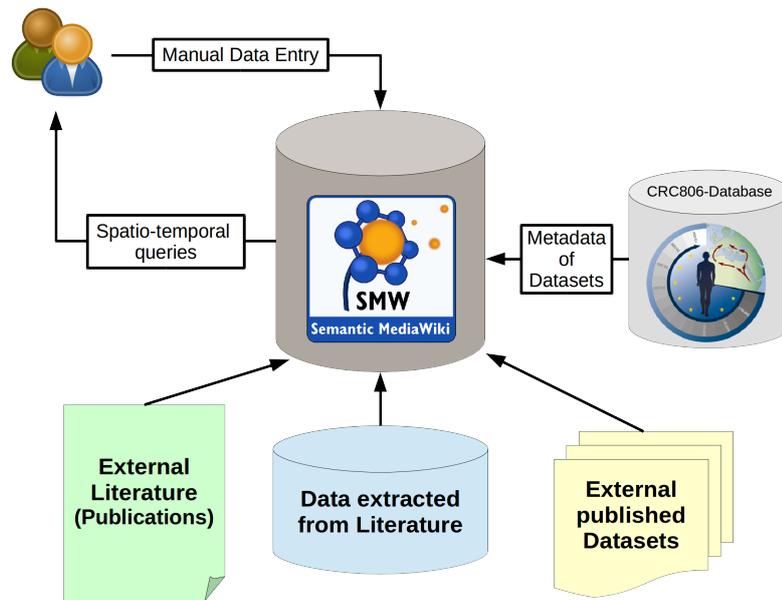


Figure 3.5.: Architecture of the CRC806-KB infrastructure. Source: Own work.

collaboratively edited, discovered and accessed through a single user friendly web application.

3.3.4. Additional VRE and collaborative infrastructure

In this sub-section all the overall services and applications, that are not directly part of the CRC806-RDM infrastructure, the CRC806-SDI or the CRC806-KB system, are summarized. Though, all three of this systems also fit into the definition as sub-systems of a VRE, as defined in section 2.5.2. But the virtual environment is also extended into the non-computerized world of services and infrastructure, as described below.

For managing collaborative work on documents, for example source code, reports or paper writing, a GitLab based VCS was setup. Additionally, *Subversion* based VCS and *trac* bug tracking and documentation wiki provided by the RRZK are used by the CRC806-Database project for managing collaborative work. The Data Management project, that is the umbrella project of the here presented infrastructure offers additionally to providing this infrastructure, GIS and Data services, to support the conduct of research in the CRC 806. This includes acquisition of data and conducting GIS analysis, as well as creating maps, or just supporting researchers to use GIS software. Another service, the Z2 Data Management project offers, are the participation in field campaigns, for delivering surveying, mapping and remote sensing services.

3.4. System integration and interoperability

The overall concepts and technology to provide interoperability and integration of resources within the CRC806-Database e-Science infrastructure and to external applications and use cases,

are described in this section.

A key aim of the CRC806-Database semantic e-Science infrastructure is to support interoperability and data integration, by implementing open standard interfaces and abiding to principles of Open Data, Open Access (OA), and Open Source Software (OSS). The internal system integration is a main focus of this work, to integrate the CRC806-RDM, CRC806-SDI and CRC806-KB infrastructures, several interfaces and tools are implemented and in development. The central part of the interoperability strategy of the CRC806-Database is the application and implementation of open standardized formats and interfaces, as described in section 3.4.3. Even if this fits to some extent also into the previous section, the application of Semantic Web Technology (SWT) within the CRC806-Database system is described in an own sub-section (3.4.4), because a main factor of SWT is the facilitation of system and data interoperability, and thus also system and data integration.

3.4.1. Semantic e-Science infrastructure resource integration

The integration of sub-systems within the CRC806-Database semantic e-Science infrastructure is of course a main interest. All information stored and available from the CRC806-RDM and the CRC806-SDI infrastructure are also integrated into and available from the CRC806-KB. Metadata records for all research datasets, publication records and geospatial datasets, are additionally stored in the SMW based CRC806-KB, see section 8.1.1. This allows to integrate the data of the RDM infrastructure in the KB, to relate records into additional context (e.g. external integrated data and publications), the contents of the RDM infrastructure are then also queryable from the KB interface. This is implemented by defining resource types in the SMW based KB infrastructure, and add according records for each resource of the RDM infrastructure. The other way around, an interface to access the KB from the CRC806-Database website, is not yet implemented, but it is possible by accessing the SMW API similar to the CKAN and GeoNode API, and planned for the future, see section 7.1.3.

3.4.2. Resource integration within the RDM infrastructure

Several approaches for internal data integration are implemented in the CRC806-RDM infrastructure. The first and most simple method, is to integrate resources by a categorization, the most obvious categorization is along the project organization, by clusters and sub projects. Another categorization would be the topic category, and of course by content type. Tags are an approach to annotate datasets with keywords, so called tags. These can be bound to controlled vocabularies or free text. The approach chosen for the CRC806-Database is free-text, to allow the creation of new tags by the user. A very useful way to integrate data is its spatial extent. In the CRC806-RDM infrastructure all resources can be annotated with a spatial extent. That allows to filter data spatially. Resources of the data catalogue are also integrated along the temporal dimension. Either by annotating the resource with a defined time period or event, or by setting a custom time span or date. This allows to filter resources by a time interval.

An extra feature, to allow the annotation of relation between any resources of the RDM infrastructure is designed as well. This related feature allows, to define a relation between for example a publication record and a geodataset and or a research data resource. On the detail pages of these resources, the related resources will be linked and shown to the user, see section 6.1.7.

3.4.3. Open standards and formats

Another method to foster system interoperability and integration is the application of open standards and formats. The CRC806-Database infrastructure implements several such interfaces, that are described in the following of this section.

Spatial Data Infrastructure Open Web Services

The CRC806-SDI implements OGC OWS interfaces, see section 2.6.2, to support interoperability. All data layers are provided in WMS (de la Beaujardiere 2006), WFS (Vretanos 2010) (vector data only) and WCS (Baumann 2012) (raster data only), the datasets are also advertised via an harvest-able CSW (Nebert et al. 2007) endpoint. See table 3.2 for an overview of the applied OGC OWS implementations. The geodatasets can also be downloaded in well known standard formats like GeoTiff, Shapefile, SpatialLite, KML, GML, and even more, as provided through GeoNode (GeoNode Contributors 2014), see section 5.2.3, 7.1, and 7.1.2.

Table 3.2.: Table overview of applied OGC OWS.

Standard	Description	Reference
Web Map Service (WMS)	Access to georeferenced map images and visualizations of spatial data.	(de la Beaujardiere 2006)
Web Feature Service (WFS)	Access to geographic feature (vector) data.	(Vretanos 2010)
Web Coverage Service (WCS)	Access to geographic coverage (raster) data.	(Baumann 2012)
Catalogue Service Web (CSW)	Access to geospatial metadata.	(Nebert et al. 2007)

Metadata standards and vocabularies

Another method to allow interoperability and data integration, is the application of metadata standards and vocabularies like DublinCore (Dublin Core Metadata Initiative 2004), ISO 19115 (ISO19115-1 2014), and DCAT (Maali et al. 2014), see table 3.3 for a complete list of the implemented standards.

The data catalogue implements DC (Dublin Core Metadata Initiative 2004) and the DataCite (DataCite Metadata Working Group 2011) Metadata Kernel v3 as the basis of the metadata of the datasets, as well as the DCAT Schema (Maali et al. 2014) that is applied by the underlying CKAN application, see section 3.3.1 and 6.1.

For the Publications DB, see section 3.3.1, 6.2 and 6.2.3, the BibTeX schema (Patashnik 1988), as well as the Bibo Ontology (D’Arcus et al. 2009) are applied for modeling the schema. The interface allows the export of publication records in BibTeX format.

For the Maps application and the according CRC806-SDI system, the metadata is handled by the GeoNode (GeoNode Contributors 2014) application. Internally GeoNode stores the metadata in the ISO 19115 schema (ISO19115-1 2014), but allows the export of the metadata in other well known and often applied metadata schemas. That are, among others, the US Federal Geographic Data Committee metadata standard (FGDC) (FGDC 1998), DublinCore Metadata (DCC 2015), and Electronic Business Registry Information Model (ebRIM) (ebRIM 2004).

Table 3.3.: Table overview of applied metadata standards.

Standard	Implementation	Description	Reference
DublinCore	Data Catalog and SDI	Basic metadata elements to describe resources	(Dublin Core Metadata Initiative 2004)
DCAT	Data Catalog	Metadata to describe Data Catalogs and Datasets	(Maali et al. 2014)
ISO 19115	SDI	Metadata for spatial data.	(ISO19115-1 2014)
ebRIM	SDI	Electronic Business Registry Information Model.	(ebRIM 2004)
FGDC	SDI	US Federal Geographic Data Committee metadata standard.	(FGDC 1998)
BibTeX	Publication DB	Schema for Bibliographic data.	(Patashnik 1988)
Bibo Ontology	Publication DB	Ontology for the description of Bibliographic data.	(D’Arcus et al. 2009)

API endpoints

The CRC806-Database e-Science infrastructure offers several standardized, or at least publicly documented API’s, for access to the data and contents of the infrastructure. As described in section 5.2.7, the SMW based KB infrastructure offers a comprehensive API for query, access and export of data. This functionality is delivered by the MW software, which is the basis of SMW. The RDM infrastructure offers beside the CKAN and GeoNode based REpresentational State Transfer (REST) API’s, the above introduced OWS endpoints of the SDI, that also implement API endpoints, and additionally Rich Site Summary (RSS) and Atom format feeds of the data catalogue, the Publications DB, the Maps application, as well as for the News & Blog’s, for content syndication.

3.4.4. Linked Data and Semantic Web

Another key aim of the CRC806-Database design is its accordance to state-of-the art web technology, and especially to the data interoperability realm. In this real, the concepts of Linked Data

and Semantic Web, as introduced in section 2.2.4, are the current most promising approaches to the overall problem of web wide data interoperability. The concepts of Linked Data and Semantic Web are applied in several ways within the CRC806-Database system.

For the resources and datasets of the RDM infrastructure, CKAN offers an RDF export of all metadata. Additionally, the HTML of the data catalogue, the Publications DB, and the Maps application, are annotated with RDFa markup, see section 4.6. This allows search engines like Google, to index the content according to defined vocabularies, that enhances the indexing and thus results in better listing positions for according searches.

The SMW based KB also offers RDF export for all structured data of the SMW instance. It is even possible to implement a SPARQL endpoint for the SMW based KB, by configuring an according backend, but this is not implemented yet in the current system. Though, it is certainly on the list of future developments for improving the infrastructure, see ???. To deliver this, the data stored in the KB needs to be linked to metadata and data models of existing schemas and vocabularies.

3.4.5. Open Science, Open Data and Open Access

The implementation of open principles, such as Open Science, Open Data and Open Access (as introduced in section 2.5.3), are also key aims of the CRC806-Database design.

Open Data data principles, such as open licenses that allow reuse of data, and general access to the datasets, meaning being able to download the data, are implemented in the RDM infrastructure through the application of CC licenses, as described in section 3.5, and through the data access right management, as described in section 9.2.

Open Access, as introduced in section 2.5.3, is also facilitated like Open Data, mainly through the applied licenses and of course the access to the according resources. The CRC806-Database offers, for example, to host pre-prints assigned with a DOI, to facilitate open peer review.

As introduced in section 2.5.3, Open Science is understood as the facilitation of open principles in the course of the scientific research process. For now, the CRC806-Database implements 4 of the 6 open principles that constitute Open Science. These are: Open Data, Open Access, Open Peer Review and Open Source, the two remaining principles, of Open Education resources and Open Methodology are not explicitly implemented, but are implicitly also existent within the CRC806-Database.

Facilitating the proper citation of web resources, as published via the CRC806-Database, also improves interoperability and integration of the CRC806-Database in third party works and applications. To allow the citation of web resources, the DOI standard was implemented, see section 6.1.4.

3.5. Copyright and licensing

It is important to take care of proper copyright and licensing policies for publishing research results, because not making clear the terms under which the data can be (re-)used is obviously to

some extent counter-productive. The default legal position on how data may be used in any given context is hard to untangle, not least because different jurisdictions apply different standards of creativity, skill, labor and expense when judging whether copyright or similar rights pertain (Ball 2014). At least in Germany, if data does not have any license assigned, it is protected by intellectual property and copyright law and can't be reused legally. Because assigning no license, means the data is the intellectual property of its creator and protected by the German *Urheberrecht* (overlaps with Anglo-Saxon understanding of copyright, but is not the same). In this sense, and in the context of the presented infrastructure, proper licensing enables the legal interoperability of the published data.

With all these complexities and ambiguities surrounding the rights of database compilers, re-users need clear guidance from compilers on what they are allowed to do with the data (Ball 2014). It is a clear design aim for the CRC806-Database to be open and facilitate interoperability, because a culture of openness deters fraud, encourages learning from mistakes as well as from successes, and breaks down barriers to interdisciplinary (Ball 2014). To facilitate and foster data reuse some DFG funded projects develop own bespoke *data policies*. This has advantages and also disadvantages. Advantages are, that terms can be tailored to the needs of the given research project setting, disadvantage is, that these terms are probably not compatible with laws in most jurisdictions. Ball (2014) asserts, that writing a bespoke license for your data is not a trivial undertaking, and almost certainly unnecessary in the light of the standard licenses available.

For the license framework of the CRC806-Database, it was decided to apply the CC licenses, as described in the following sub-section.

3.5.1. Creative Commons

As elaborated above, immediate, unrestricted access to scientific ideas, methods, results, and conclusions, is not always compatible with the stringent rules of copyright, which apply fully and automatically to all published works, by default. Licences are a topic many researchers shy away from. And it is common behavior that property rights are unknowingly signed away (Friesike 2014). The exercise of something less than full copyright requires, oddly, some legal tinkering—which is where Creative Commons (CC), the organization, comes in (Brown 2003). CC is a non-profit organization, founded by Lawrence Lessig in 2001, with the aim to develop and maintain licenses, that allow to legally share and reuse creative works. The organization was founded on the idea that some people prefer to share their works on more generous terms than standard copyright provides (Brown 2003).

The licenses are formulated in a simple yet robust modular way, that allow creators to define which rights they reserve and which rights they waive, see table 3.4 for a description of the available CC licenses. Every CC license works around the world and lasts as long as applicable copyright lasts, because the licenses are based on copyright (Creative Commons 2015).

The CC licenses also play an important and growing role in the Open Access realm (Creative Commons 2016). The license model is applied by many Open Access publishers as their default license model.

Table 3.4.: Creative Commons License options model. Official descriptions from (Creative Commons 2015).

Code	Meaning	Description	Logo
CC BY	Attribution	This license lets others distribute, remix, tweak, and build upon your work, even commercially, as long as they credit you for the original creation. This is the most accommodating of licenses offered. Recommended for maximum dissemination and use of licensed materials.	
CC BY-ND	Attribution-NoDerivs	This license allows for redistribution, commercial and non-commercial, as long as it is passed along unchanged and in whole, with credit to you.	
CC BY-SA	Attribution-ShareAlike	This license lets others remix, tweak, and build upon your work even for commercial purposes, as long as they credit you and license their new creations under the identical terms. This license is often compared to “copyleft” free and open source software licenses. All new works based on yours will carry the same license, so any derivatives will also allow commercial use. This is the license used by Wikipedia, and is recommended for materials that would benefit from incorporating content from Wikipedia and similarly licensed projects.	
CC BY-NC	Attribution-NonCommercial	This license lets others remix, tweak, and build upon your work non-commercially, and although their new works must also acknowledge you and be non-commercial, they don't have to license their derivative works on the same terms.	
CC BY-NC-SA	Attribution-NonCommercial-ShareAlike	This license lets others remix, tweak, and build upon your work non-commercially, as long as they credit you and license their new creations under the identical terms.	
CC BY-NC-ND	Attribution-NonCommercial-NoDerivs	This license is the most restrictive of our six main licenses, only allowing others to download your works and share them with others as long as they credit you, but they can't change them in any way or use them commercially.	
CC Zero	Public Domain	You can copy, modify, distribute and perform the work, even for commercial purposes, all without asking permission.	

Within the CRC806-Database the CC licenses of version 4.0 are applied, because the versions of the licenses prior to version 4 were not specifically aimed at data (Ball 2014). In conclusion, CC licenses are chosen as the copyright framework for the data published through the CRC806-Database. Resources published through the system, can be assigned any of the seven CC licenses as listed in table 3.4.

4. Development Methods

In this chapter, the development methods including their theoretical background for creating the CRC806-Database e-Science infrastructure are introduced. Here, the development methods are understood as the techniques and its foundational concepts, that were applied to implement the infrastructure. This includes also to some extend software or framework specific approaches.

4.1. Web development

Most web development work for the CRC806-Database RDM infrastructure was done using the *Extbase & Fluid* (Rau et al. 2013) framework, to develop the Typo3 extensions that handle the RDM demands. Extbase & Fluid are based on the design patterns and principles of Domain Driven Design (DDD) (see section 4.1.1) and Model View Controller (MVC) (see section 4.1.2), that will be described in the following of this section.

4.1.1. Domain Driven Design

DDD is a software development pattern, to model and build complex object oriented software projects (Evans 2004). The Extbase framework (Rau et al. 2013), that is used to build the data management related extensions for Typo3, is implemented based on the DDD pattern. DDD is not only a method or technique, it is more a paradigm of thinking to enhance the development quality and productivity in large and complex domain specific software projects (Evans 2004).

Figure 4.1 shows, where the DDD has its place in the context of a program run, in this case a user interaction.

The premise of DDD are the following points (Lobacher 2014):

- The software projects focus is its core domain and domain logic.
- All complex designs are directly based on the domain model.
- Fostering a creative collaboration between technical and domain experts to iteratively refine a conceptual model to address problems of the domain in focus.

The idea behind DDD is nothing really new. Many software developers ascertained, that real world problems need to be modeled very precise in software to be useful to help solving a certain problem (Rau et al. 2013: 27). DDD is foremost a pragmatic take on software development and related to the prototyping approach (see section 4.5), in most cases many model refinement iterations are needed to come up with the final domain model (Rau et al. 2013: 26).

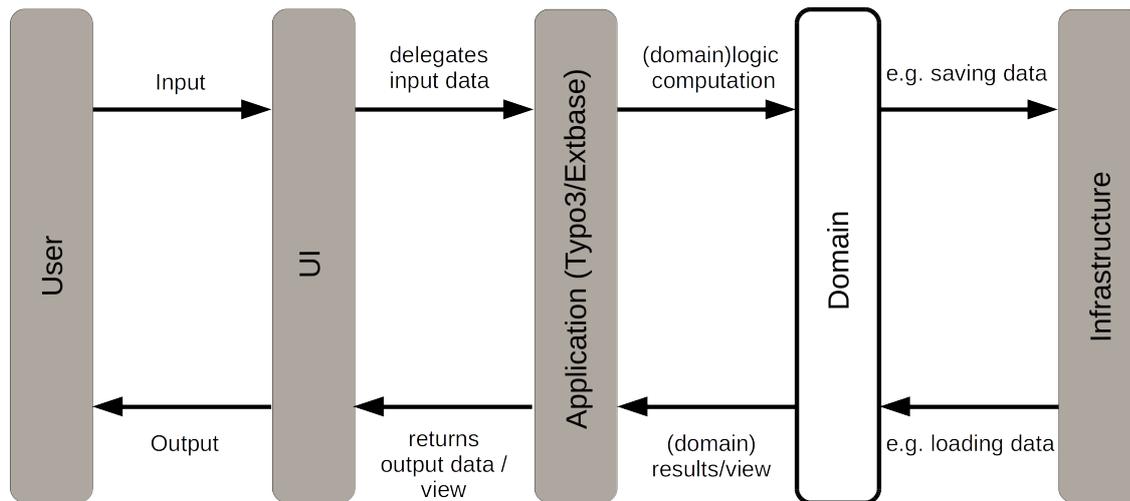


Figure 4.1.: Domain Driven Design layered view. Source: Own work, after Lobacher (2014: 74).

4.1.2. Model View Controller

MVC is a software architecture design pattern for the implementation of an UI based software application. The design pattern divides the application into three interconnected aspects (see figure 4.2). These three aspects are defined as follows:

- **Model:** Holds the domain model including the domain logic, concerns the data access and storage.
- **View:** Displays the data and UI frontend, provides functionality to the user and facilitates user interaction.
- **Controller:** Coordinates the data and input/output flow of the application between the model and the view

MVC was conceived in 1978 as the design solution to a particular problem. The top level goal was to support the user's mental model of the relevant information space and to enable the user to inspect and edit this information (Reenskaug 2003). Traditionally used for desktop graphical user interfaces (GUIs), this architecture has become extremely popular for designing web applications (Lobacher 2014).

The DDD requires a *layered architecture*, that is provided to Extbase through the application and implementation of the MVC concept (Lobacher 2014: 84). Typo3 Extbase & Fluid implements the MVC pattern (Rau et al. 2013: 37), the frameworks abstracts many aspects of the development process. For example, the framework takes care of creating and maintaining the table schema of the persistence layer, the data base backend.

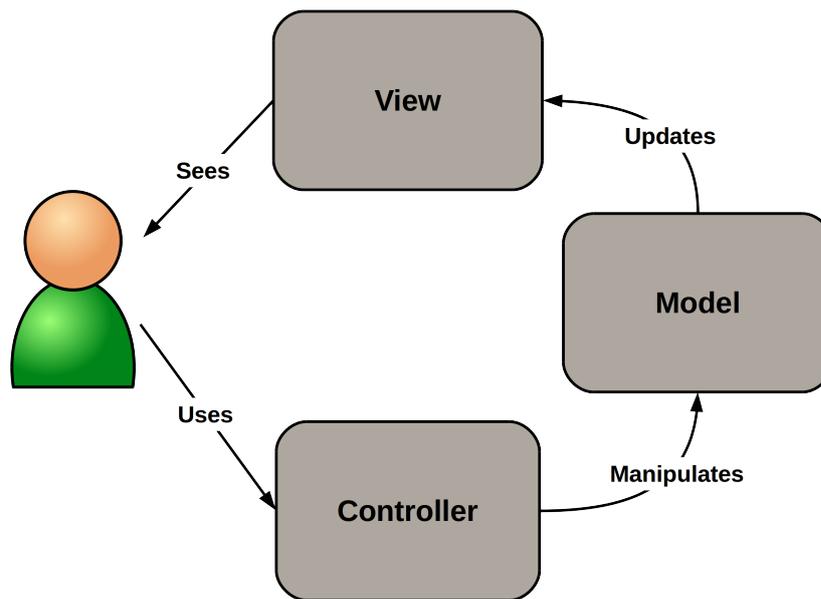


Figure 4.2.: Model View Controller (MVC) concept. Source: Own work.

4.1.3. Continuous Integration

For securing and strengthening of the application, a comprehensive testing approach based on the *continuous integration* (Duvall et al. 2007; Fowler 2006) pattern was implemented, as described in section 6.6.

Continuous Integration is a software development practice where members of a team integrate their work frequently, usually each person integrates at least daily - leading to multiple integrations per day. Each integration is verified by an automated build (including test) to detect integration errors as quickly as possible. Many teams find that this approach leads to significantly reduced integration problems and allows a team to develop cohesive software more rapidly (Fowler 2006).

The application of *continuous integration* (CI) (Duvall et al. 2007) in software development has several advantages, that help to stabilize and secure the system. But it also has some disadvantages. Implementing a CI test, compile or build pipeline takes an investment of effort, time and knowledge, additionally it needs discipline by the developers, to write the according tests, when implementing new functionalities. The advantages are as follows. The code base is tested, each time a change is committed to the VCS. This reduces the risk of deploying errors. Bugs can be better detected, while they are fresh, and the code is still in the developers mind (Duvall et al. 2007). The approach also provides more transparency on the functionality of the code, because the tests provide a kind of documentation of what is expected from the code. Frequent deployment is possible, because the code base is always tested for its intended functionality (Fowler 2006).

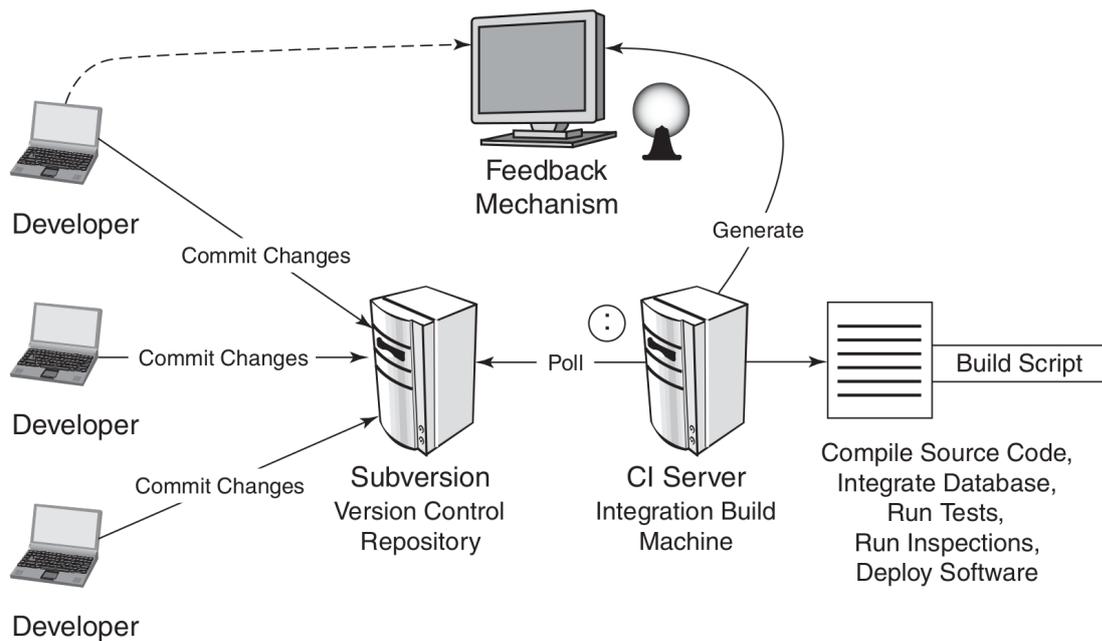


Figure 4.3.: The components of a Continuous Integration System. Source: (Duvall et al. 2007: 5).

In figure 4.3 the components of a Continuous Integration System are shown. It always involves a VCS and an integrating *Build Machine* that is responsible for facilitating the test environment, see section 6.6 for a detailed description, of how the CI system of the CRC806-Database is implemented. The steps in a CI scenario will typically go something like this (after Duvall et al. (2007), see figure 4.3):

1. First, a developer commits code to the version control repository. Meanwhile, the CI server on the integration build machine is polling this repository for changes (e.g., every few minutes).
2. Soon after a commit occurs, the CI server detects that changes have occurred in the version control repository, so the CI server retrieves the latest copy of the code from the repository and then executes a build script, which integrates the software.
3. The CI server generates feedback by e-mailing build results to specified project members.
4. The CI server continues to poll for changes in the version control repository

4.2. System administration

System administration is not directly a method in the strict sense of the term, but it is described here in this context anyway, because has a pivotal role in building, maintaining and providing IT infrastructure, like the CRC806-Database. Additionally, the craft of system administration is an often underestimated but diverse and complex skill set, well needed to be able to make informed decisions for design and implementation of IT systems.

The skill set covered by the term of system administration includes knowledge and understanding of:

- Servers and their Operating System (OS)
- Web architecture
- Computer networks
- Data bases

These topics need to be mastered in a level, that allows detection and solving of problems in the configuration and interplay of the system components, to be able to implement and integrate an infrastructure as presented in this work.

Configuration, stability/robustness, performance, backup and security of the systems demand great care and expert knowledge. The learning curves are often steep and the number of sub-systems to install and maintain are manifold, as can be grasped from section 5.1. The amount of work, to successfully maintain a system infrastructure like the CRC806-Database is considerable, at least in the development and built up phase. When the system is built and well configured, so that it runs without major problems and is optimized for security and performance, the workload of administering the system drops considerably. Although, as brief as this topic is described here, this brevity does not represent the amount of work necessary and its overall share in conducting the development of the CRC806-Database, that it would deserve compared to the other parts described in this work.

The topic of System Administration is not further detailed, because this would lead to far in this context. For some parts of the CRC806-Database infrastructure, the task of system administration is delivered by the RRZK, e.g. web-projects server administration, MySQL DB administration and maintenance of the RedHat Enterprise Linux (RHEL) based Virtual Machine (VM) servers. The infrastructure components that are maintained and those that are partly or not maintained by the RRZK, are listed in table 5.1, and will be described in more detail in that same section.

The topic of System Administration is closely related to the System Architects and Developers, an individual who combines knowledge and applies in all of this three domains is also known as a *DevOp*. DevOp (a clipped compound of "development" and "operation") is a culture, movement or practice that emphasizes the collaboration and communication of both software developers and other information-technology (IT) professionals while automating the process of software delivery and infrastructure changes (Erich et al. 2014).

4.3. Top-down vs. bottom-up development approach

Top-down and *bottom-up* are two complementary approaches to system development and implementation (Sabatier 1986). Both are strategies of information processing and knowledge ordering (Hare et al. 2006), used in a variety of fields including software, humanistic and scientific theories, and management and organization, as well as policy development and implementation (Goldfarb et al. 2003). In practice, they can be seen as a style of thinking and teaching.

4.3.1. Top-down

A top-down approach to system engineering is also known as decomposition or stepwise design (Goldfarb et al. 2003). This recursive heuristic is decomposed to the following steps; start with a big-picture view of the problem; break it into a few big sub-problems; figure out how to integrate the solutions to each sub-problem; and then repeat for each part. Shalizi (2012) formulated the following step based procedure for implementing a top-down approach:

1. **The big-picture view:** resources (mostly arguments), requirements (mostly return values), the steps which transform the one into the other.
2. **Breaking into parts:** try not to use more than 5 sub-problems, each one a well-defined and nearly-independent calculation; this leads to code which is easy to understand and to modify.
3. **Synthesis:** assume that a function can be written for each sub-problem; write code which integrates their outputs.
4. **Recursive step:** repeat for each sub-problem, until you hit something which can be solved using the built-in functions alone.

A top-down model is often specified with the assistance of "black boxes", these make it easier to manipulate (Goldfarb et al. 2003). However, black boxes may fail to elucidate elementary mechanisms or be detailed enough to realistically validate the model (Sabatier 1986). Shalizi (2012) concludes the method as follows:

Top-down design forces you to think not just about the problem, but also about the method of solution, i.e., it forces you to think algorithmically; this is why it deserves to be part of your education in the liberal arts (Shalizi 2012).

4.3.2. Bottom-up

The bottom-up approach is understood as the piecing together of smaller less complex systems into a larger more complex system (Barbuti et al. 1993), thus making the original systems sub-systems of the emergent system. In a bottom-up approach the individual base elements of the system are first specified in great detail. These elements are then linked together to form larger subsystems, which then in turn are linked, sometimes in many levels, until a complete top-level system is formed. Bottom-up is also referred to as an *organic strategy* (Kienreich et al. 2006). However, "organic strategies" may result in a tangle of elements and subsystems, developed in isolation and subject to local optimization as opposed to meeting a global purpose.

In a study about automated annotation of contents shown in photos and videos (Hare et al. 2006), a bottom-up and a top-down approach to modelling the information was compared. The authors believe, that the best approach to system implementations is to combine these two approaches (Hare et al. 2006).

In system modelling, the process oriented approach would be the top-down implementation, where the agent-based approach would be the bottom-up implementation (Kienreich et al. 2006).

4.4. Data model development

The CRC806-Database data model is built, in a bottom-up approach, from a set of sub-domain models. The sub-domains of the CRC806-Database are:

- **Data Catalog:** The data catalogue data model was developed from the CKAN Schema, that is based on DCAT and additional properties for spatial and temporal attributions.
- **Spatial Data Infrastructure:** The SDI data model is based on the GeoNode model, that is based on ISO19115 and INSPIRE.
- **Publication DB:** The data model of the publication DB is based on the BibTeX Schema (Patashnik 1988), and linked to the Bibo ontology (D'Arcus et al. 2009).
- **Knowledge Base:** The data model of the CRC806-Knowledgebase is developed by integrating existing datasets into the KB and adapting its data model according to the integrated datasets.

The data models of the data catalog, SDI and publication DB, are based on existing metadata schemas, and are thus not really developed by the author, so there is no development method to describe. For the KB this is different, here the data model is developed from scratch, applying state of the art development methods, that are described in the following section accordingly.

4.4.1. Data modeling

Models designed for human communication and interpretation have a crucial advantage over models that are crafted for use by computers; they can take advantage of the human ability to apply common sense and are able to set the given data in context to interpret signs and give them meaning (Allemang et al. 2011: 14). This advantage on the side of the modeler, to not have to model too precisely and complex, is on the other hand a disadvantage, because it opens the door for all manner of abuse, both intentional and unintentional. When a model relies on particulars of the context of its reader for interpretation of its meaning, as is the case in legislation, we say that a model is informal. That is, the model lacks a formalism whereby the meaning of terms in the model can be uniquely defined (Allemang et al. 2011: 15).

In the case of developing models, that are understandable by computers, the complexity, or amount of information or even amount of knowledge, present in the model is directly dependent on the level of modeled understanding. An other formulation would be the level of abstraction or the amount of connected and formalized information. All information has to be formalized and existent in the given system to be applied. A fundamental aspect of modelling is the management of commonality versus variability. When describing sub-sets of reality, some sets will have things in common (commonality), and some will have significant differences (variability).

The modeling activity is the activity of distilling communal knowledge out of a chaotic mess of information (Allemang et al. 2011: 24).

From a more technical perspective, we can observe that there are a number of approaches, techniques and patterns known and established to do practical data modeling. According to Blaha (2010), there are three main patterns to structural data modeling:

- **Hierarchical** - in a hierarchical data model the data is organized into a tree like structure. The relations between the data is given through the hierarchy. Hierarchical model can be managed in file systems applying a folder structure that represents the data model.
- **Relational** - a relational data model is based on first-order predicate logic. In a relational model all data is represented in tuples that are grouped into relations. Relational data models are implemented in well known products like PostgreSQL or MySQL.
- **Network** - a network data model represents a graph in which object types are nodes and relationship types are arcs. The advantage of this model is, that it is not restricted to strict formalism like hierarchies, but can also model them. The most prominent implementation of the graph data model is the RDF standard.

Furthermore, the ANSI/X3/SPARC Study Group on Data Base Management Systems (1975) standard for data modelling describes three kinds or levels of data models:

- **Conceptual** - the conceptual schema describes the high level semantics of the UoD, and defines the scope of the model. The conceptual model is the first step in organizing the data requirements. It defines entity classes of significance to the UoD, and relationship assertions between entity classes.
- **Logical** - the logical schema defines the model applied to a structural data model pattern (e.g. hierarchical, relational, or network). The logical schema and the conceptual schema are often implemented as one and the same.
- **Physical** - the physical schema describes the technical means to store and handle the data. This is bound to the technology that is applied, for example the kind of Data Base Management System (DBMS), if a relational database or a simple file system storage or a combination of both is used.

And finally we have two major methodological approaches to data modelling (see section 4.3 for background):

- **Bottom-up** - the Bottom-up approach develops the data model from existing data structures and data sets to build an overarching model to enable their use in the resulting system.
- **Top-down** - the Top-down approach defines data models in an abstract way from the requirements of the desired system.

4.5. Prototyping

Because the prototyping approach (Naumann et al. 1982) is applied for data modelling in this work, the basics of this approach are presented here. The prototyping approach is a well known

and established Systems Development Method (SDM), in which a prototype is developed as an early approximation of the final system or model, that is adapted and changed during the development process until all functionality and capabilities of the system are implemented. The advantage of the approach is its simplicity and clarity for practical implementation purposes. It is summarized by Naumann et al. (1982) as a four step iterative process between system user and developer. i.) The initial version (prototype) is defined and implemented, ii.) the prototype is used, iii.) misfits and errors are corrected, iv.) the system or model is evaluated and finalized. Step ii. and iii. are repeated iteratively until iv. is positive, see figure 4.4.

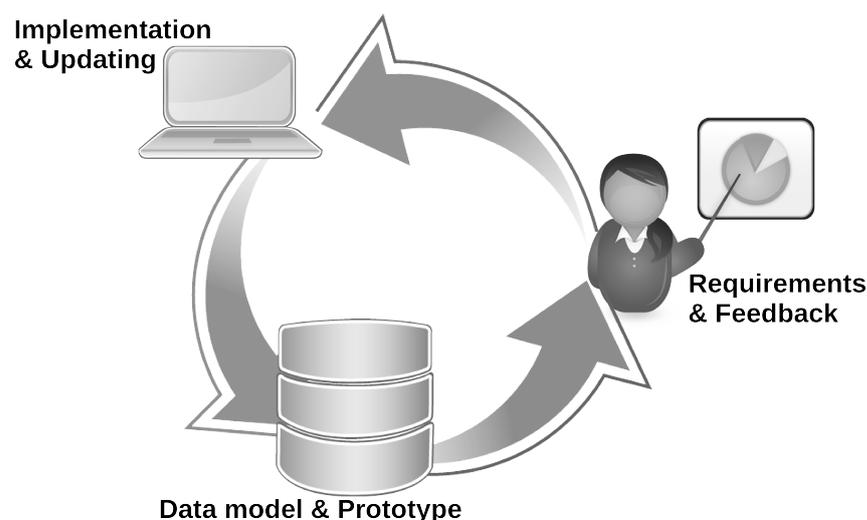


Figure 4.4.: The prototyping approach: a four step iterative process. Source: Own work, after Naumann et al. (1982).

According to the before explained properties (structure, level, methodology) of data modelling, the prototyping approach is independent of the model structure, that means it can be applied to all kinds of model structures. Prototyping is applied in the model design phase, and is thus primarily conceptual and logical, but it can also impact the physical model. And methodologically it is clearly bottom-up.

The prototyping approach also integrates the top-down and the bottom-up development approaches, as described in section 4.3 before, in a practical manner. By combining the iterative aspect of the top-down approach with the bottom-up strategy of piecing together smaller entities of less complexity into an overall complex model. This makes it the perfect approach for data modelling in the CRC806-Database context.

Prototyping also relates to the concept of Application Lifecycle Management (ALM) (Kääriäinen et al. 2009), that defines the balances between re-implementing or adapting an application for a given purpose. Here a top-down decision is taken, to control the application lifecycle, that is intrinsically bottom-up. This approach was applied for the design of the CRC806-Database Repository Lifecycle, as described in section 3.2.3.

4.6. Linked Data

In this section the implementation of Linked Data concepts in the Typo3 based web application of the CRC806-Database is presented. As described before in section 2.2.4, Linked Data facilitates interoperability, through annotation of data and its properties with well known vocabularies and metadata schemas. For the implementation of Linked Data concepts within the web application, RDFa is implemented. Thus a short overview of technical RDFa concepts is given here, for further reference within this work.

RDFa implementation

RDFa contains of a set of attributes, that allow the annotation of XML, and thus also HTML, elements. These RDFa attributes are:

about a URI used for stating what the data is about

rel, rev specifying a relationship (link) with another resource

scr, href, resource specifying the partner resource

property specifying a property for the content of an element or the partner resource

content optional attribute that overrides the content of the element when using the property attribute

datatype optional attribute that specifies the datatype of text specified for use with the property attribute

typeof optional attribute that specifies the RDF type(s) of the subject or the partner resource (the resource that the metadata is about).

In the web development practice applied in the CRC806-Database, mainly the property attribute for annotating a property with a defined vocabulary is implemented. A basic example of an RDFa annotated bibliographic record in a HTML website would look like shown now. Listing 4.1 shows the vocabulary references for namespaces.

Listing 4.1: Vocabulary definition in the HTML document header.

```
<html xmlns="http://www.w3.org/1999/xhtml"
  xmlns:xhv="http://www.w3.org/1999/xhtml/vocab/"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:bibo="http://purl.org/ontology/bibo/"
  version="XHTML+RDFa 1.0" xml:lang="en">
```

And Listing 4.2 shows an example RDFa annotated bibliographic record.

Listing 4.2: RDFa annotated bibliographic record.

```
<span typeof="bibo:Article">
<span property="bibo:authorList">
  <span property="dc:creator">Willmes, C.</span>,
  <span property="dc:creator">Brocks, S.</span>,
  <span property="dc:creator">Hoffmeister, D.</span>,
  <span property="dc:creator">H&uuml;tt, C.</span>,>
```

```
<span property="dc:creator">K&uuml;rner, D.</span>,  
<span property="dc:creator">Volland, K.</span>,  
<span property="dc:creator">Bareth, G.</span>  
</span>  
(<span property="dc:date" datatype="xsd:gYear">2012</span>):  
<span property="dc:title">Facilitating integrated spatio-temporal visualization and analysis of heterogeneous  
archaeological and palaeoenvironmental research data.</span>  
<span property="bibo:Journal">ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.</span>  
<span property="bibo:Issue">I-2</span>,  
<span property="bibo:pages">  
  <span property="bibo:pageStart">223</span>-  
  <span property="bibo:pageEnd">228</span>  
</span>. DOI:  
<span property="bibo:doi">10.5194/isprsannals-I-2-223-2012</span>.  
</span>
```

On a side note, RDFa is not only useful to implement Linked Data concepts, nowadays large search engine providers use RDFa annotation to index content of websites. This way, the search engine can extract more meaning from the content on an RDFa annotated website and will list the website probably higher on queries matching this kind of content.

5. Technology

In this chapter the technology stack applied for creating the infrastructure components are introduced. This will be described first from the server infrastructure aspect in section 5.1, detailing the hardware specifications and important configuration details, and then from the software and application aspect in section 5.2, detailing software specific implementation and configuration specifics.

5.1. Server infrastructure and technology

In this section the server and storage infrastructure of the CRC806-Database is described. All infrastructure, that is publicly accessible is hosted at and in some cases by the RRZK. The difference between 'hosted at' and 'hosted by' RRZK, denotes that 'hosted by' applications are running on servers that are maintained by the RRZK. Meaning, that the server is installed by and taken care of security updates and patches by the RRZK. Those servers would be the RRZK *Webprojekte* servers, that provide Linux, Apache, MySQL, PHP server (LAMP) web space for university websites, and the RRZK *MySQL DB* server infrastructure, that provides pooled MySQL DB's to University of Cologne (UoC) web projects. The term 'hosted at' denotes, that only the hardware (or virtualized hardware in form of an VM), is provided by the RRZK. An overview of the involved servers is given in table 5.1. The servers are described in detail in own sub-sections in the following.

RRZK data storage The long term secure data storage, as well as all servers and web applications of the CRC806-Database are based on the data storage provided by the RRZK. As shown in figure 5.1, the Storage Area Network (SAN) is the physical basis of the RRZK storage infrastructure, all other services of the RRZK are based on this storage (Foertsch n.d.; Curdt et al. 2008), thus also all of the here presented server infrastructure. The RRZK has one of its main focuses on the implementation of secure storage solutions, and is one of the first institutions in Germany, that virtualized all its storage infrastructure in a cooperation with IBM in 2004 (Foertsch n.d.). The virtualization of storage has the advantage, that the data security is almost independent of hardware failure (Clark 2005), because the data is stored in virtualized volumes, that are physically implemented on redundant disk arrays, if a physical hard disk fails, it can be replaced by a functioning one and the data is replicated by the disk array' Redundant Array of Independent Disks (RAID) implementation (Clark 2005). For the secure long term storage and preservation of data the CRC806-Database system trusts on the Andrew File System (AFS)

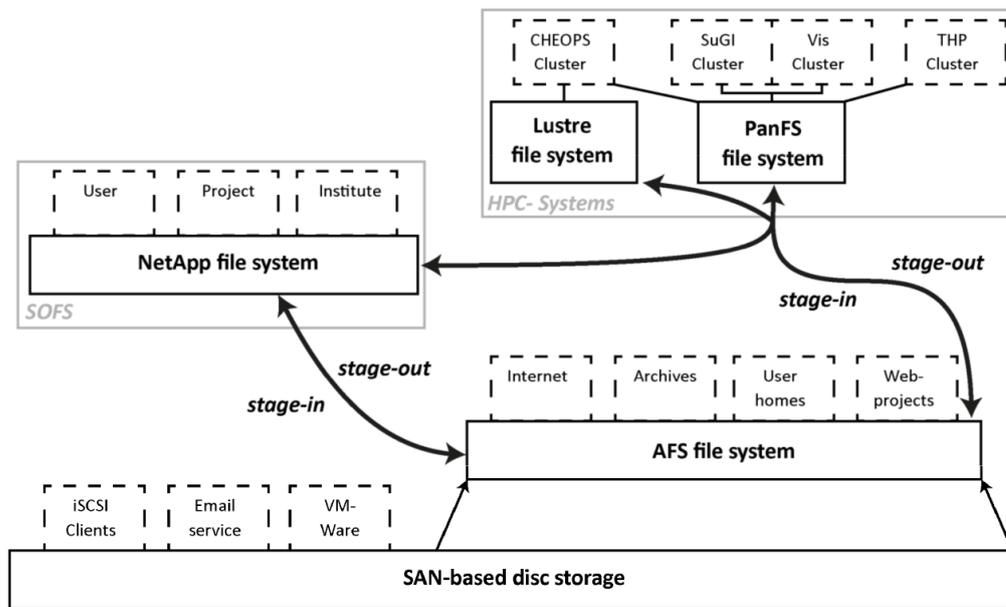


Figure 5.1.: RRZK data storage infrastructure. Source: Curdt (2014b: 71).

(RRZK 2015a), that is physically implemented on the RRZK SAN system (Curdt 2014b: 72). The AFS is a distributed file system which uses a set of trusted servers to present a homogeneous, location-transparent file name space (RRZK 2015a). The AFS serves at the RRZK, as back-end storage for all web-projects including web-space hosted at the RRZK (Curdt 2014b: 72).

RRZK webprojects The RRZK provides hosted so called *webprojects* as a service (RRZK 2015b). The webprojects include by default 200 MB webspace on a shared LAMP server, the storage space can be extended on request, a MySQL database instance and PHP serve side script executions. The RRZK maintains a daily backup for the files in the webspace as well as for the MySQL database, additionally a tape based long term storage is maintained, that allows to recover older versions of a webproject, if necessary (RRZK 2015b). For convenient access, and for management of the MySQL database, a phpMyAdmin instance is provided for the project users.

Typo3 server The RRZK provides hosted Typo3 (Typo3 Contributors 2014) instances as a service, these instances are normally centrally managed, by the RRZK webmaster team. This includes maintenance of updates and security bug fixes by the RRZK webmaster team. Because the CRC806-Database develops own Typo3 extensions based on Extbase & Fluid technology (see section 5.2.1), that is based on the most recent Typo3 v6 LTS, we needed to switch to a self maintained Typo3 instance, because at the time the RRZK provided only Typo3 v4.5 LTS, as a managed service. Although we use the RRZK managed LAMP webproject, to host the Typo3 v6 system, so that the webserver maintenance, including updates and security fixes is maintained by the RRZK.

Sever name	Application	Type	Access level	Specs
CRC806-Database Website	Typo3 instance	RRZK Webproject (pooled LAMP web-space)	UniAccount, read-write access to web space folder	200 MB Web-space
CRC806-Database MySQL	MySQL backend of the Typo3 instance	RRZK pooled MySQL instance	Webproject account, read-write access to DB instance,	Unknown
CKAN	CKAN instance	RRZK hosted VM	Root access	2 CPU, 2 GB RAM, 100 GB HD
GeoNode	GeoNode instance	RRZK hosted VM	Root access	2 CPU, 4 GB RAM, 300 GB HD
sfb806srv	CRC806-Database general purpose server	RRZK hosted RHEL instance	UniAccount with some sudo privileges	2 CPU, 1 GB RAM, 100 GB HD
sfb806db	SMW instance and general purpose webspace	RRZK Webproject (pooled LAMP web-space)	UniAccount, read-write access to web space folder	200 MB Web-space
GitLab	GitLab instance	RRZK hosted VM	Root access	2 CPU, 2GB RAM, 200 GB HD

Table 5.1.: Server infrastructure of the CRC806-Database, as of July 2015.

CKAN Server For the CRC806-Database CKAN (Open Knowledge Foundation 2014) instance (<http://ckan.crc806db.uni-koeln.de/>, see table 5.1), a RRZK VM based dedicated server is applied. The CRC806-Database project has full Root-Access to the server. The VM server instance is equipped with 2 CPUs, 2 GB RAM and 100 GB disk space. From the software side, the system consists of a Ubuntu 14.04 LTS (Ubuntu Community 2014) OS instance, that has the CKAN software installed, including the PostgreSQL (PostgreSQL Contributors 2015), PostGIS (Ramsey et al. 2014), and Solr (Solr Contributors 2015) applications, that CKAN is based on. See section 5.2.2 for a detailed technical description of the CKAN software. The server is backed-up through the VM infrastructure of the RRZK, additionally we backup the PostgreSQL data base backend of the CKAN instance and the CKAN application directory using a weekly cron-job.

GeoNode Server For the GeoNode (GeoNode Contributors 2014) server of the CRC806-Database (<http://geonode.crc806db.uni-koeln.de>, see table 5.1), another VM based dedicated server, including root access was installed. The server instance has more hardware resources compared to the CKAN instance, because the GeoNode application, and in particular its underlying GeoServer (GeoServer Contributors 2014) needs considerable resources to perform requests in decent response times. Thus, the instance has 2 CPUs, 6 GB of RAM and 350 GB of disk space. The disk space is split onto two volumes, one volume of 20 GB for just the operating system and the application software, and the other volume of 330 GB as storage for the geospatial data. The

server OS is a Ubuntu 14.04 LTS server edition (Ubuntu Community 2014), the GeoNode app is installed via the GeoNode package repository (`ppa:geonode/release`). Further detail on the setup of the GeoNode server are described in section 7.1. Details about the GeoNode application can be found in section 5.2.3.

Data server The data server of the CRC806-Database (<http://sfb806srv.uni-koeln.de>, see table 5.1), is a RRZK hosted VM based server instance. In contrast to the CKAN, GeoNode and GitLab servers, described above, this server is maintained by the RRZK. The server is also VM based and runs a RHEL v5 (Red Hat, Inc. 2015). The CRC806-Database project has access to the server, with limited administrative rights. The main responsibility of this server is the web-based upload of data files from the CRC806-Database Typo3 based web-application into the AFS based long term secure storage of the RRZK.

Furthermore, this server also hosts the MapProxy (Tonnhofer et al. 2014), PIWIK (PIWIK Contributors 2014), and MapServer (MapServer Contributors 2014) instances of the e-Science infrastructure.

GitLab Server The GitLab (GitLab Community 2015) Server (<http://gitlab.crc806db.uni-koeln.de>, see table 5.1) is also a dedicated RRZK hosted VM server, equipped with 2 CPUs, 2 GB of RAM and 200 GB of disk space. The OS is Ubuntu 14.04 LTS server edition (Ubuntu Community 2014), GitLab is installed via the GitLab repository (<https://packages.gitlab.com>) packages. Details about the GitLab software can be found in section 5.2.5.

5.2. Software infrastructure and technology

The CRC806-Database is based on a number of OSS components, an overview is given in table 5.2. The most important main software components of the CRC806-Database system will be introduced in this section. It is not possible, and also not helpful, to describe all software components that are part of the CRC806-Database, because a lot of them are either standard software, like for example the Apache webserver (Apache HTTPD Contributors 2015), or it would go too far to describe the details of a Linux distribution like Ubuntu (Ubuntu Community 2014). Applied programming languages and data formats will also not be introduced in detail, though they may be shortly introduced in the particular context of its application.

5.2.1. Typo3

Typo3 (Typo3 Contributors 2014) is a free and open source CMS framework. The Typo3 framework is written in the popular scripting language PHP, as DB backend MySQL, MariaDB, PostgreSQL or Oracle can be used (Lobacher 2014). Typo3 has a modular designed structure, meaning that most functionality is modularly configured and added through extensions. This results in a comparably thin core system, that allows to configure systems, that solve a certain problem

Software	Description	Website	Source
Typo3	CMS of the CRC806-Database webportal	http://typo3.org	(Typo3 Contributors 2014)
CKAN	Metadata store of the CRC806-Database	http://ckan.org	(Open Knowledge Foundation 2014)
GeoNode	SDI backend of the CRC806-Database	http://geonode.org	(GeoNode Contributors 2014)
GitLab	Collaboration platform and code repository.	http://gitlab.com	(GitLab Community 2015)
MapProxy	MapProxy is used to cache some geospatial services from MapServer and GeoNode/GeoServer	http://mapproxy.org	(Tonnhofer et al. 2014)
MapServer	Delivers some of the geospatial services of the SDI	http://mapserver.org	(MapServer Contributors 2014)
MediaWiki	The Wiki system of the CRC806-Database KB	http://mediawiki.org	(Wikimedia Fdn. 2015)
Semantic MediaWiki	MediaWiki extension to store structured semantic data.	https://semantic-mediawiki.org/	(SemanticMediawiki Contributors 2015)
Mobo	Mobo is a toolset for modeling SMW structure in a Schema-Driven Development (SDD) approach.	https://github.com/Fannon/mobo	(Heimler 2014)

Table 5.2.: Overview of main software components and applications of the CRC806-Database.

without a lot of overhead functionality and code (Rau et al. 2013). At the time of writing, more than 5000 Extensions are available from the Typo3 Extension Repository (TER).

Typo3 is, along with *Drupal*, *Joomla!* and *WordPress*, among the most popular content management systems worldwide, however it is more widespread in Europe, and especially in German speaking countries, compared to other regions (Lobacher 2014). This has lead to a considerable amount of German documentation, books and blogs about Typo3.

The CMS is professionally developed, by a core development team organized and partly funded by the Typo3 Association (Typo3 Association 2015), as well as by many companies, ranging from small start-ups to global players like IBM for example (Lobacher 2014).

Extbase & Fluid

Typo3 v6 comes with some useful technical features, that were not available in v4.5. The Extension framework *Extbase & Fluid* (Rau et al. 2013) is natively supported by v6, in v4.5 an extension was needed, to be able to run Extbase & Fluid based extensions. Extbase & Fluid

extensions are developed in the DDD (Evans 2004), see section 4.1.1, and MVC (Lindberg et al. 2002) pattern, see section 4.1.2, which allows to implement complex tasks in a clean and relatively fast process. The downside is the relatively steep learning curve for programming with Typo3 and Extbase & Fluid applying the DDD and MVC patterns (Rau et al. 2013). But it leads to better maintainability and cleaner source code structure, because of the more strict separation of functionality, scope, and UI.

The Typo3 Extensions of the CRC806-Database are developed using Extbase & Fluid technology.

5.2.2. CKAN

The CKAN (Open Knowledge Foundation 2014) is a free and OSS web based data management system written in Python. The aim of the project is to facilitate open data infrastructure to the public sector, i.e. Open Government, and to academia, i.e. Open Access. The project describes itself as quoted in the following:

CKAN is the world's leading open-source data portal platform. CKAN makes it easy to publish, share and work with data. It's a data management system that provides a powerful platform for cataloging, storing and accessing datasets with a rich front-end, full API (for both data and catalogue), visualization tools and more (Open Knowledge Foundation 2014).

The code-base of CKAN is maintained by the OKFN, employing several full time developers working on the project. The development of the CKAN software began in March 2006 with the first public release in July 2007 (Winn 2013). As of today CKAN is primarily used by organizations and institutions and is increasingly popular among public sector, local to international, for implementing Open Government and Open Data principles. Some of the more prominent CKAN installations are for example the open data platform of the UK government (<http://data.gov.uk>), or the public data platform of the EU (<http://publicdata.eu/>), a very good example for a CKAN based Open Data platform in a local government setting, is the Open Data portal of the German City of Cologne (<http://www.offenedaten-koeln.de/>). And just recently, the EU Joint Research Centre (JRC) launched the JRC Data Catalogue (<http://data.jrc.ec.europa.eu/>) (JRC 2016).

Because CKAN is used in many well known institutions and companies, it has very well designed functionality for all aspects of data management. The most important functionality is of course the storage and publication of data. It is possible to store data in CKAN in several ways; directly via the web interface, using CKAN's Action API, or via custom importers (Via custom spreadsheet importers, CSV,XML). Additionally, customizable harvesting mechanism which can fetch and import records from many different repository sources, including:

- Geospatial CSW Servers
- ArcGIS, Geoportal Servers
- Z39.50 databases

- Other CKAN instances

All functionality of the CKAN system is also available through a RESTful API, called "CKAN Action API". This API exposes the CKAN functionality in Remote Procedure Call (RPC) style to clients. Some of the main features of the API are:

- Get JSON-formatted lists of a site's datasets, groups or other CKAN object,
- Get a full JSON representation of a dataset, resource or other object,
- Search for packages or resources matching a query,
- Create, update and delete datasets, resources and other objects,
- Get an activity stream of recently changed datasets on a site.

The server side code of CKAN is written in Python. The web pages of the system also include some comprehensive JavaScript code as well. The backend is facilitated by PostgreSQL (PostgreSQL Contributors 2015) and the search frontend by Solr (Solr Contributors 2015). The geospatial extension (*ckan-extgeo*) uses PostGIS (Ramsey et al. 2014), the spatial extension of PostgreSQL, to facilitate spatial search for the CKAN catalog application.

5.2.3. GeoNode

GeoNode (GeoNode Contributors 2014) is an OSS web-based application and platform for developing WebGIS and for deploying SDI. On the project website¹, the community describes GeoNode as an "Open Source Geospatial Content Management System". It is designed to be extended and modified, and can be integrated into existing platforms.

As shown on figure 5.2, GeoNode is mainly based on the GeoServer (GeoServer Contributors 2014) Open Source server implementation for sharing geospatial data. Through the application of GeoServer as middleware, GeoNode is capable to deliver OGC compliant interfaces, such as WMS, WFS, WCS and CSW. GeoNode configures GeoServer via the REST API, the GeoServer contains the layer data, and GeoNode's layer model extends the metadata present in GeoServer with its own. The web application is implemented using the Django and specifically the GeoDjango (GeoDjango Contributors 2014) Python based web application framework. Additional crucial components of the GeoNode web application are:

1. pycsw (Kralidis et al. 2014), that is responsible for the OGC conform CSW geospatial metadata directory implementation;
2. the GeoExt (GeoExt Contributors 2014) and OpenLayers (OpenLayers Contributors 2015) based WebGIS visualization component;
3. the SOLR (Solr Contributors 2015) based Elastic Search engine, to deliver fast and accurate search for the GeoNode hosted data sets.

¹<http://geonode.org/>, accessed: 2016-01-25.

GeoNode Component Architecture

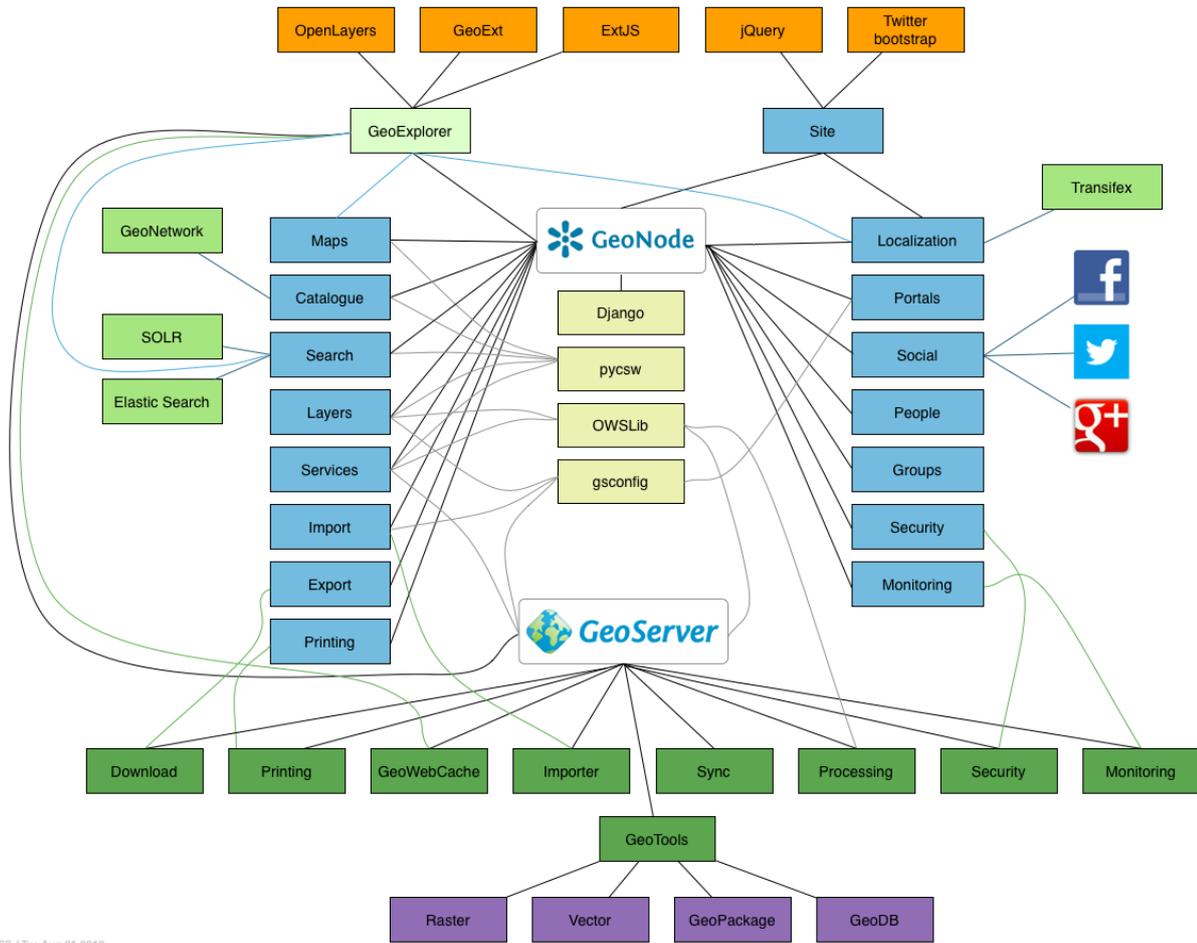


Figure 5.2.: GeoNode Component Architecture. Source: GeoNode Contributors (2014)

4. As data backend, PostgreSQL (PostgreSQL Contributors 2015) and PostGIS (Ramsey et al. 2014) are facilitated.

GeoNode serves as the backend of the CRC806-SDI, because it offers a comprehensive and intuitive web application to set up and manage geospatial web services, including access and user rights management capabilities.

5.2.4. MapProxy

MapProxy is an OSS proxy for geospatial data services, known as OWS, written in Python (Python Software Foundation 2016). It caches, accelerates and transforms data from existing map services and serves any desktop or web GIS client (Tonnhofer et al. 2014). See figure 5.3, for a schematic view of the MapProxy use case.

In the application of CRC806-Database SDI, MapProxy is mainly used to cache WMS services delivered by MapServer and to cache data from remote sources, to provide these services for

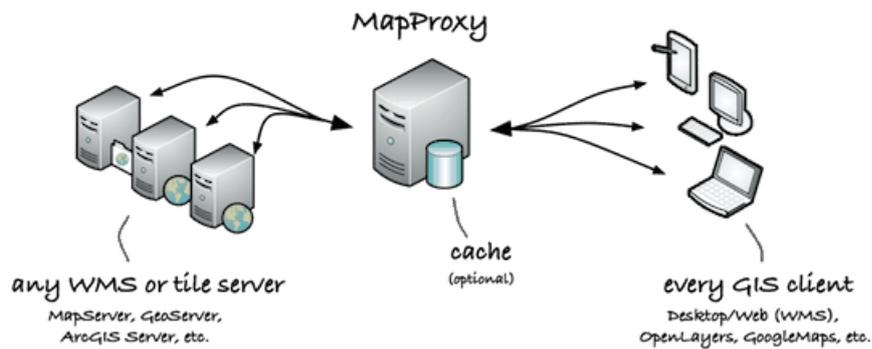


Figure 5.3.: MapProxy overview. Source: <http://mapproxy.org/>, CC-BY: Omniscale GmbH & Co. KG.

internal GIS access. The MapProxy application is particularly useful to re-project WMS that are provided only in a certain projection, for use in GIS projects applying a different projection. Additionally, some services that are popular and often used, are cached using MapProxy to reduce the load on the GeoNode server.

5.2.5. GitLab

GitLab (GitLab Community 2015) is a web application to collaborate on source code and any kind of documents in general. It is free and OSS, and licensed under the MIT license.

The software is based on Git, a distributed VCS, originally developed by Linus Torvald's, the inventor of Linux (Chacon et al. 2015). GitLab offers a web based application around Git code repositories, including features for user management, project management, ticketing and bug tracking. The web application is developed in Ruby on Rails² technology, and can be comfortably installed via Linux distribution packages, for example via `apt-get` repository on Ubuntu Linux.

Many people compare GitLab to the famous GitHub³ platform, because GitLab offers a similar interface and set of functionality. GitHub is free for open source code, but has a pricing structure for private usage and storage from \$7 for single developers all the way up to \$200 for businesses, with higher pricing if more storage is needed (Olanoff 2011). Another concern, why GitLab was chosen instead for example GitHub is privacy and control of the system. GitLab can be hosted on own infrastructure, and thus does not have to be available over the internet. Furthermore, all code is 100% percent under control of the infrastructure administration (e.g. server system administrator).

5.2.6. Mediawiki

For the CRC 806 Knowledge Base (CRC806-KB), a semantic wiki based approach was chosen. The semantic wiki, is based on the MediaWiki (MW) (Wikimedia Fdn. 2015) wiki implementation.

²<http://rubyonrails.org/>, accessed: 2015-08-18.

³<https://github.com/>, accessed: 2015-08-18.

MW is well known as the software basis of the famous Wikipedia online encyclopedia, with millions of content entries and also millions of daily users. The MW software is free and OSS, and implemented in PHP & MySQL. The MW project has a large developer community, with several full time developers, financed from large cooperations using the software, as well as small to mid-size consultancies offering MW based services and also, financed by the Wikimedia foundation and some of its local chapters (i.e. the Germany chapter, with 6 full time developers for several MW based software projects like WikiData and SMW) is stable and mature, and facilitated in many business, educational, Non-Governmental Organization (NGO) and governmental installations. MW is in use in tens of thousand of wikis around the world, it's almost certainly the world's most popular wiki software (Koren 2012).

5.2.7. Semantic Mediawiki

MW is brilliant in facilitating wiki functionality, like collaboratively editing of unstructured text, but it lacks functionality for managing structured information. This is where SMW (Semantic-Mediawiki Contributors 2015) can help out, it adds the possibility to collaboratively add and edit structured information in MW (Krötzsch et al. 2006). It defines a framework for storing data in a wiki, and querying it - which has the effect of turning a wiki into a collaboratively editable database (Koren 2012).

SMW is a free, open-source extension to MW, that enables to store and query data within the wiki's pages, and offers a full-fledged framework, in conjunction with many spin-off extensions, that can turn a MW instance into a powerful and flexible KMS. All data created within SMW can easily be published via the Semantic Web, allowing other systems to use this data seamlessly (SemanticMediawiki Contributors 2015).

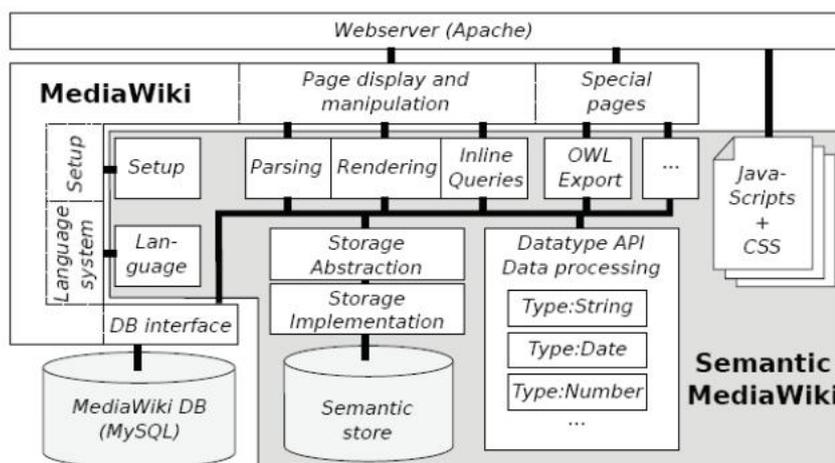


Figure 5.4.: SMW Architecture and integration into MW. Source: Krötzsch et al. (2007).

While SMW introduces structured storage of information, it can only be entered by formulating it in MW wikitext markup. To ensure a better user experience and enforce a desired structure of

the information, the Semantic Forms (SF) extension⁴ allows defining custom web forms (Heimler 2015b). SMW includes a simple query language for semantic search, called ASK, so that users can directly request certain information from the wiki (Koren 2012). Query results can be exported in several well known formats, such as CSV, XML, JSON, and more (see fig. 5.5). It is also possible to display query results directly in the wiki, using a number of provided so called *Semantic Result Formats*, like tables, data graphs or the Semantic maps Extension for displaying query results on interactive maps.

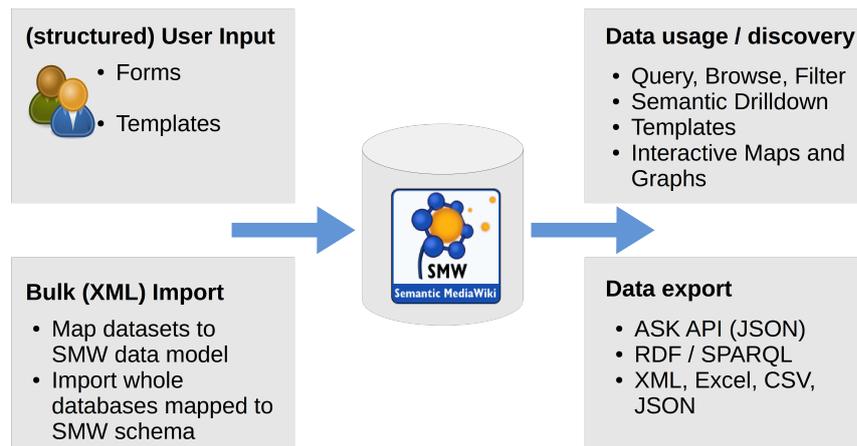


Figure 5.5.: SMW interfaces. Source: Own work.

A considerable advantage of the SMW platform, is its ability to serve its content via W3C standardised SWT, such as RDF, RDFa and SPARQL. SMW allows to configure a triple-store (e.g. Virtuoso (OpenLink Software 2015), Sesame (Sesame community 2015), or Fuseki (Apache Jena Contributors 2015)) as its data backend, that then serves as the SPARQL endpoint. Table 5.3 shows the MW components installed for the SMW setup, that is the basis for the development of the here presented KMS.

Knowledge Management in SMW

In SMW knowledge is managed based on Semantic Triples and properties, and queries upon those properties and triples. Knowledge entry into the system is facilitated using Semantic Forms, sophisticated display of knowledge stored in the system is facilitated using Semantic Result Formats. These techniques are briefly introduced in the following four sub-sections.

Semantic triples and Properties SMW's main feature is, that it enables MW to manage structured data. In SMW datum (data item) is represented as a semantic triple. Semantic Triples are the central concept of SWT (see section 2.2.4), that indicates a three-part structure: a subject, a predicate and an object (Koren 2012: 159). An example would be:

⁴https://www.mediawiki.org/wiki/Extension:Semantic_Forms, accessed: 2015-08-17.

Table 5.3.: MediaWiki and most relevant extensions setup

Name	Version	Description	Reference
MediaWiki	1.24.0	The MediaWiki core.	(Wikimedia Fdn. 2015)
Semantic MediaWiki	2.0	The SMW extension.	(SemanticMediawiki Contributors 2015)
Semantic Forms	3.0	Forms for adding and editing semantic data.	(Koren et al. 2015a)
Semantic Maps	3.1.3	Provides the ability to view and edit geographic data.	(De Dauw 2015)
Semantic Result Formats	2.0	Result formats for Semantic MediaWiki queries	(Koren et al. 2015b)
Data Transfer	0.6	Allows for importing and exporting data contained in template calls	(Koren 2015)

Germany Has capital Berlin

Where "Germany" is the subject, "Has capital" is the predicate (or relationship, or link), and "Berlin" is the object. In MW all content is stored in *wikitext* notation on wiki pages. This basic principle of MW also applies to SMW content. In SMW, the predicate is known as the "Property", and the subject is always the wiki page on which the value is stored (Koren 2012: 159). To encode the example triple in SMW would be to store the following string on the wiki page of name "Germany":

```
[[Has capital::Berlin]]
```

This syntax, allows the SMW wikitext parser to capture the semantic triple in its data base, and make it available for queries. Properties in SMW can have different types, and it depends on the type if and how the above notated triple is displayed or rendered on the wiki page. For the details on how to define properties, it is referred to the SMW documentation (SemanticMediawiki Contributors 2015), because explaining this in detail would go to far for the given scope. The take home message here is, that all SMW content is stored via wikitext markup in the wiki.

Queries If structured data is stored, it is obviously desirable to be able to query this data. In SMW queries on the structured data are facilitated from the ASK query language of SMW. The syntax of this query language is similar to the syntax of annotations in SMW. This query language can be used on the SMW special page `Special:Ask`, in SMW concepts, and in inline queries⁵.

SMW queries consists of two parts; 1.) which pages (subjects) to select, and 2.) What informations (properties) to display about those pages. All queries must state some conditions that describe what is asked for. You can select pages by name, namespace, category, and most importantly by property values. For example, the query:

⁵https://semantic-mediawiki.org/wiki/Help:Semantic_search

```
{{#ask: [[Category:Countries]]?Has capital}}
```

Would yield a list of Countries and their Capitals stored in the wiki. The first, "[[Category:Countries]]", is the filter - it defines which pages get queried; in this case, all pages in the category "Countries". The second part, after the "|", is called "printout", and selects the properties of the filtered pages (subjects) to display. In the example, all properties of "Has capital".

Semantic Forms The SF extension (Koren et al. 2015a) provide a way to edit template calls within a wiki page, where the templates are facilitated to store structured information in SMW. It thus complements SMW, by providing a structure for SMW's storage capabilities (Koren 2012: 181). The concept of SF is based on the MW templating concept. MW templates can provide structure and the definition of the display of the structured content to wiki pages. Thus, templates are useful for structuring the input of content to MW, and delivering a definition for the display of the content.

A template definition looks like the example template shown in listing 5.1:

Listing 5.1: Example Semantic MediaWiki template definition.

```
{| class="wikitable formdata"
! colspan="2"|Artefact
|-
| '''ArtefactType'''
| [[ArtefactType::{{{ArtefactType}}}|]]
|-
| '''Site'''
| [[Site::{{{Site}}}|]]
|-
| '''DatedAge'''
| {{#arraymap:{{{DatedAge}}}|;|x|[[DatedAge::x]]|,&nbsp;}}
|-
| '''TimePeriod'''
| [[TimePeriod::{{{TimePeriod}}}|]]
|-
| '''Layer'''
| [[Layer::{{{Layer}}}|]]
|-
| '''Dataset'''
| [[Dataset::{{{Dataset}}}|]]
|-
| '''Reference'''
| [[Reference::{{{Reference}}}|]]
|-
|}]
```

To store actual content on a wiki page, using this example template, the template call shown in listing 5.2 can be used:

Listing 5.2: Example Semantic MediaWiki template call, for storing structured data.

```
{{Artefact
|ArtefactType=Hominin
|Site=Cueva Ardales
```

```
|DatedAge=34000 BC
|TimePeriod=Aurignacien
|Layer=VI-2
|Dataset=NESPOS
|Reference=Kehl et al. 2012
}}
```

Semantic Result Formats The Semantic Result Formats (SRF) extension (Koren et al. 2015b) provide additional result formats for SMW inline queries, as described above, to display query results in additional formats and visualizations (Koren 2012: 221). The version of SRF that is used in the here presented installation, includes 41 semantic result formats, that are available to visualize and export query results. These result format cover almost any use case, there are result formats for calendars, timelines, charts, graphs and mathematical functions. On the extensions website⁶, all result formats are listed and documented, the formats are organized in seven categories; misc, math, export, time, charts, tables, and graphs.

Semantic Maps A special SRF is the *Semantic Maps* extension (De Dauw 2015), it allows to show query results, containing properties of special SMW type Geographic Coordinates. In Semantic Maps it is possible to use multiple mapping services. These include Google Maps (with Google Earth support), Yahoo! Maps, OpenLayers and OpenStreetMap (De Dauw 2015). In listing 5.3, an example inline query, that produces a Semantic Map, showing all records of category *Site*, by its property *Coordinates*. The property *Coordinates* needs to be of type Geographic Coordinates, that is the special type defined by the SemanticMaps extension.

Listing 5.3: Example inline query, yielding a map as result format.

```
{{#ask:
[[Category:Site]]
| ?Coordinates
| height=600
| format=map
}}
```

The query shown in listing 5.3 yields the map shown in figure 5.6, from the data stored in a CRC806 Knowledge Base staging SMW instance.

5.2.8. Mobo

Developing and maintaining a data model in SMW can get very complex and cumbersome, because any change of the model must be applied in several pages (e.g. property pages, template pages, form pages) of the wiki. To manage this complexity Heimler (2014) introduced the Mobo⁷ toolkit at the SMWCon fall 2014⁸ in Vienna. Mobo is a toolset that helps to build SMW structure

⁶https://semantic-mediawiki.org/wiki/Semantic_Result_Formats, accessed: 2015-08-22

⁷<https://github.com/Fannon/mobo>, accessed: 2015-08-16.

⁸https://semantic-mediawiki.org/wiki/SMWCon_Fall_2014, accessed: 2015-08-16.

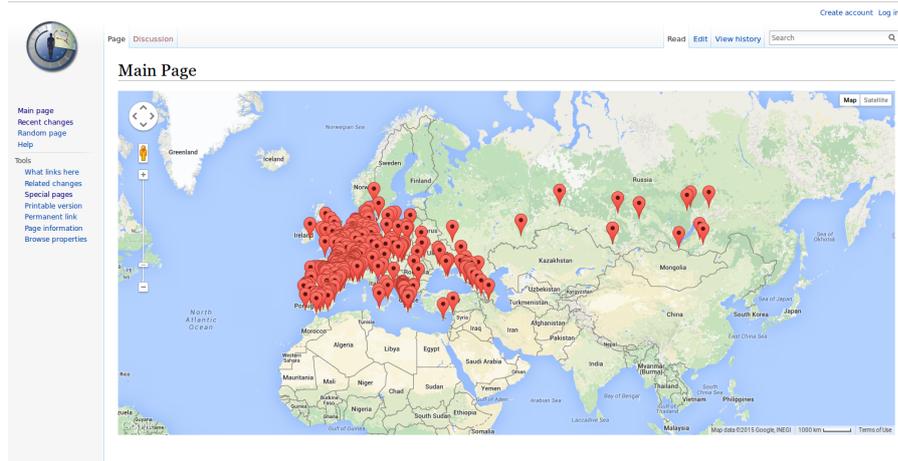


Figure 5.6.: Example SemanticMap.

in an automated SDD (a simplified Model-Driven Engineering (MDE)) approach. SDD uses annotated data schemas, which specify the expected data structures, as models to generate system artifacts (code, documentation, tests, etc.) automatically (Heimler 2015b). The model is formulated and developed in Yet Another Markup Language (YAML) or JSON, using the object oriented JSON Schema (Galiegue et al. 2013). The SDD approach simplifies the SMW data model development, because the building blocks are more generic, and thus more simple to reuse. The model can therefore be very "Don't repeat yourself (DRY)" (Heimler 2015b). The target wiki must have the Semantic MediaWiki (SemanticMediawiki Contributors 2015) and Semantic Forms (Koren et al. 2015a) extension installed. It is highly recommended to install the *ParserFunctions* Extension, since mobo's default templates make use of it. It is possible to adjust/use templates that work without it, instead (Heimler 2015b).

The main feature of Mobo is the simplified and improved model development workflow. Semantic MediaWikis can be developed rapidly and modular, leading to a more agile development process. Mobo can run in an interactive mode, automatically validating and uploading the development model in real-time (Heimler 2015a).

Mobo is written in JavaScript and is executed as a NodeJS⁹ application, the UI offers a CLI (see figure 5.7) to build the model and run the updates of the configured SMW instance. The changes are applied to the SMW instances in wikitext content that is written to the wiki pages, using a MediaWiki Bot. The tool itself is not integrated into the MediaWiki software in any way, it runs completely independent on the developers computer (Heimler 2014). The software also features a web application, that runs on localhost, for inspecting the development model in its various stages visually, and can also be used to batch-import wiki pages or data automatically (Heimler 2015b). Mobo is free and OSS, and licensed under the MIT license.

Another very useful feature of using Mobo is, that the model schema can be versioned and managed in a VCS. The fact, that a model can be developed separated from the SMW instance, makes it possible to easily deploy the model to several instances, without much additional work.

⁹<https://nodejs.org/>, accessed: 2015-08-18.

```

christian@U772: ~/git/data/model/mobo/sfb806db
File Edit View Search Terminal Help
christian@U772:~/git/data/model/mobo/sfb806db$ mobo -h
ABOUT MOBO
-----
Mobo is a command line toolset that helps building Semantic MediaWiki structure in an agile, model driven
engineering (MDE) way.

For documentation please head over to: https://github.com/Fannon/mobo

CONSOLE COMMANDS
-----
--version          (-v)   Display version
--help            (-h)   Display help text
--settings        (-s)   Displays current settings
                    Includes inherited and calculated settings

--update          (-u)   Updates the project mobo_templates to the latest mobo templates
                    This might be necessary if new features are introduced.
                    Creates a Backup from your current templates first.

--init            (i)    Creates a new raw project in the current directory
--example hardware          Installs the "hardware" sample project
--example hardware-yaml    Installs the "hardware" sample project, using YAML instead of JSON
--example shapes           Installs the "shapes" sample project
--example shapes-yaml      Installs the "shapes" sample project, using YAML instead of JSON

--force           (-f)   Forces the upload of the complete model
--run-through     (-r)   Skips watching the filesystem and serving the webapp
                    mobo will exit after completion. This might be
                    useful if mobo is triggered through other skripts.

--skip-upload     Skips uploading and deleting on the external wiki

--import <director>      Imports all files from /import/<director> to the wiki

--nuke content          Nukes all maint content wiki pages
--nuke structure        Nukes all structural wiki pages (templates, categories, ...)
--nuke custom-namespaces Nukes all content from custom namespaces
--nuke <namespaceNumber> Nukes the namespace of the given number

DEVELOPER CONSOLE COMMANDS
-----
--update-schemas  Generates / Updates the JSON Schema documentation (SCHEMA.md files)
christian@U772:~/git/data/model/mobo/sfb806db$

```

Figure 5.7.: The Mobo CLI interface.

The feature of Mobo to be able to validate the model for its syntax, structure and semantics is also valuable. The syntax is validate in the parsing step of the schema files, by the JSON and YAML libraries, that Mobo includes. The structure is checked, by validating the model against the meta-schema, additionally it is possible to check for semantic errors, by implementing according logics (Heimler 2015b). Further external JSON or YAML linters¹⁰ and validators can be facilitated to evaluate the model. Since wikitext misses validation capabilities, the ability of the generator to validate the development model is a big benefit over using wikitext directly (Heimler 2015b).

¹⁰Tools that flag suspicious usage in software written in any computer language.

Part III.

Implementation

6. CRC806-RDM Infrastructure

After introducing the research questions and overall setting, including related work and theoretical background for this work in Part I. In Part II the demands by all stakeholders of the project were analyzed, as well as the design, the architecture, the technology and the methods chosen for the development of the infrastructure were worked out.

The integration of the different technological components, as introduced and described in chapter 5, into the CRC806-Database semantic e-Science infrastructure is given here in the implementation part.

The CRC806-RDM implementation, as part of the CRC806-Database e-Science infrastructure, consists of a data catalog (see section 6.1), a data archive (see section 6.1.1), and a publication database (see section 6.2).

The Typo3 based web application¹, that implements the frontend, integrates the CRC806-RDM and CRC806-SDI infrastructures, it consists of i.) a data repository for the handling of research data of the project, ii.) a publication database for handling the publications of the project in a central place, and iii.) a SDI for handling of spatial data. Additionally, an application for aggregating news and disseminating updates of the data base, in form of the news extension, as well as user management system, implemented in the members section, was realized. This is schematically depicted in figure 6.1.

The architecture (as shown in fig. 6.1), organizes the system into *layers* (vertical, y-axis) and *tiers* (horizontal, x-axis). The layers are divided into Frontend, Middleware, Backend and API. Tiers are structured by the main sections of the website and its applications; News, Members, Data, Publications DB, and Maps. The technologies to implement the applications according to their layer and tier are then accordingly placed. In this way, for example, the diagram shows that the News application is implemented as a Typo3 Extension, using Extbase & Fluid technology for the extension middleware and MySQL as backend data base technology, it further has RSS and ATOM feeds, for automated API based access.

In 2014 the CRC806-Database underwent a major update and redesign. From 2010 to 2014 the first version of the CRC806-Database (Willmes et al. 2014e) was based on Typo3 version 4.5 (Typo3 Contributors 2014). Due to the update of the underlying Typo3 CMS from version 4.5 to version 6, that was demanded by the RRZK because the v4.5 systems would reach end of maintenance soon, the opportunity was taken to completely redevelop the Typo3 frontend, to solve some minor and major issues of the infrastructure (Willmes et al. 2015). For the new version of the CRC806-Database, the system was moved from the RRZK maintained Typo3 instance to a

¹<http://crc806db.uni-koeln.de/>

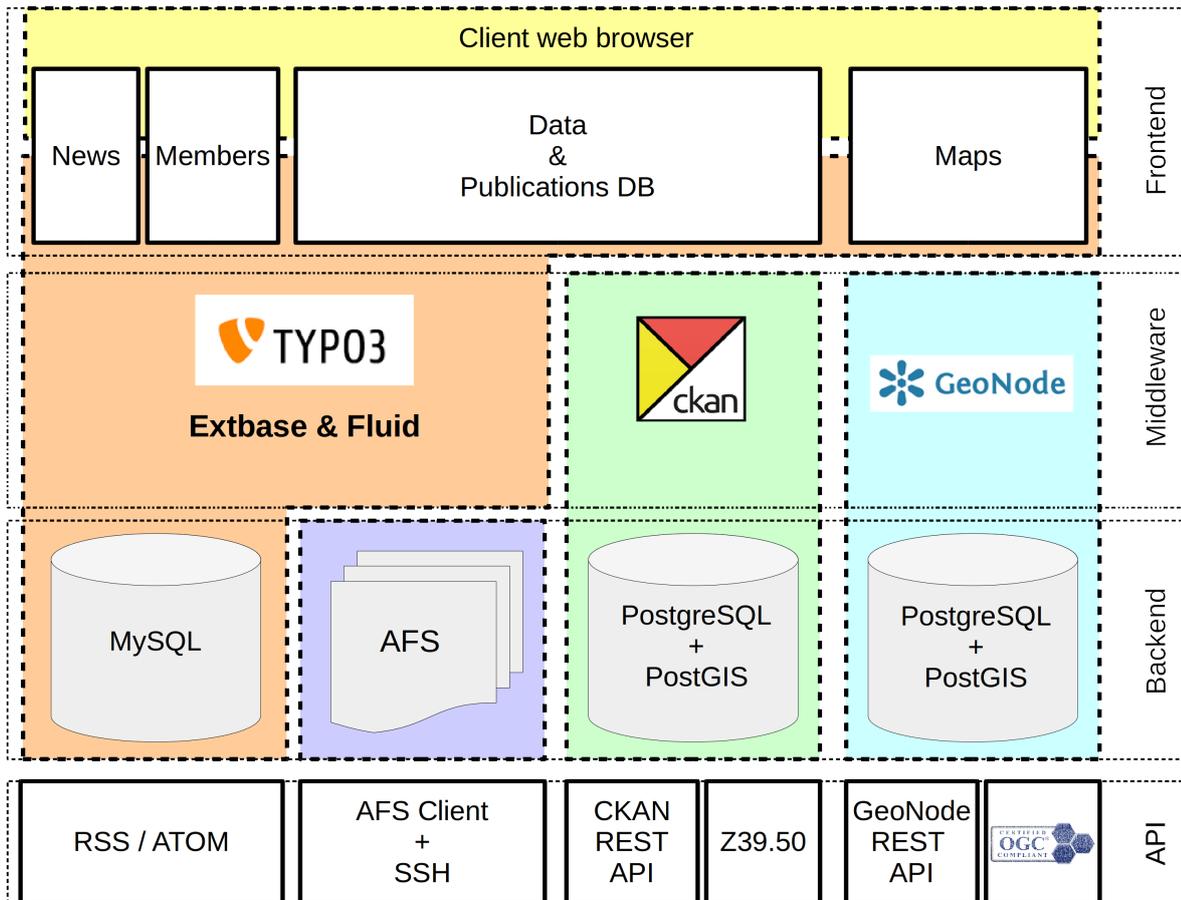


Figure 6.1.: Layered architecture diagram of the CRC806-RDM infrastructure. Source: Own work.

self maintained, but RRZK web hosted Typo3 instance (see section 5.1). The switch from Typo3 version 4.5 to version 6 had several advantages. The first advantage was that the system was no more dependent on the RRZK for updating the Typo3 core, that caused some problems in technical compatibility and increased maintenance and development workload, due to our custom developed extensions, that failed automated checks by the RRZK administration, and caused a system shutdown twice. Because of these incidents, we decided to maintain the Typo3 base our self, but chose to install it into an RRZK hosted web project. With this solution we have better control on the update cycles and the dependencies between the extensions and the core Typo3 system. This delivers the second advantage, that on the self maintained Typo3 it is an internal project decision if an upgrade to an upcoming new major version of Typo3 has to be taken or not, for example if compatibility between versions is broken, like it was between version v4.5 and v6 of Typo3. This is a crucial point for the long term availability of the system (see section 12.4), that was not possible in the former RRZK hosted Typo3, where updates needed to be maintained regularly.

Until this update, the functionality of the data catalog was based on AngularJS (Google Inc. 2014), a JavaScript MVC implementation. The AngularJS technology in conjunction with CKAN

and Typo3 proved to have some serious flaws, so it was decided to improve and thus redesign the frontend. For the re-implementation, it was decided to facilitate the then newly introduced Extension framework *Extbase & Fluid* (Lobacher 2014; Rau et al. 2013). This framework is a modern MVC implementation for Typo3 (Lobacher 2014), see table 6.1 for an overview of the Typo3 based extensions implemented for the CRC806-Database, on the basis of this technology.

As already indicated, Typo3 v6 brings some new useful technical features, that were not available in v4.5. The Extension framework *Extbase & Fluid* (Rau et al. 2013) is now natively supported by v6, in v4.5 an additional extension was needed, to be able to run *Extbase & Fluid* based extensions. *Extbase & Fluid* extensions are developed in the DDD (Evans 2004) and MVC (Lindberg et al. 2002) pattern, which allows to implement complex tasks in a clean and relatively fast process. The downside is the relatively steep learning curve for programming with Typo3 and *Extbase & Fluid* applying the DDD and MVC patterns (Rau et al. 2013; Lobacher 2014). An overview of, in the course of this project, custom developed Typo3 Extensions is given in table 6.1.

Table 6.1.: Overview of Typo3 Extbase & Fluid Extensions.

Name	Description	Details
Data	All functionalities of the data catalog	Sec. 6.1, p. 142
Publications	The bibliography database concerning functionalities and interfaces.	Sec. 6.2, p. 157
Maps	The GeoNode integration and SDI related functionality and interfaces.	Sec. 7.3, p. 178
News	The Blog & News functionality, as well as the latest news, datasets and maps streams and RSS feeds.	Sec. 6.4, p. 165
Members	All functionalities according user and access rights management.	Sec. 6.3, p. 163
Search	Functionalities for the integrated search interface.	Sec. 6.5, p. 166

During the 2014 major update, it was also taken care of preserving all URL's of resources (datasets and web pages), published in the CRC806-Database (Willmes et al. 2015). As shown in the RDM architecture diagram, figure 6.1, the RDM infrastructure of the CRC806-Database e-Science infrastructure consists of four layers (Frontend, Middleware, Backend and API), and of three to six (depending on the layer) technology tiers in each application layer. The implementation of the system will be described, structured by application layer, in the following of this chapter. The backend includes the storage and database systems, that are i.) PostgreSQL that is used by CKAN and GeoNode, and ii.) MySQL that is used by Typo3, as well as iii.) the file system based storage backend for long term preservation of research data. The middleware layer includes the application logic of the CRC806-Database system, these are the model and controller parts of the Typo3 based extensions, the whole Typo3 CMS, as well as the CKAN and the GeoNode applications. This section is structured into subsections for the different application parts, data catalog

(section 6.1), including the CKAN based data and publications application middleware, user management (section 6.3), and the news and blogs application (section 6.4). The frontend of the RDM infrastructure is one central website, the CRC806-Database (<http://crc806db.uni-koeln.de>), integrating the above described backends of CKAN and GeoNode in a customized Typo3 based web application. To provide one single point of entry, e.g. one consistent web application, for all interfaces of the RDM system was a central goal for the design and architecture of the infrastructure.

6.1. Data catalog

The data catalog is the center piece of the CRC806-RDM infrastructure. The core of the data management demands and principles, as introduced in chapter 6, are implemented in this application. The implementation is described along the backend (section 6.1.1), middleware (section 6.1.2) and frontend (section 6.1.5), organized as shown in figure 6.1. Where the backend handles the storage and persistence of files and metadata, the middleware contains almost all of the application logic and algorithms for handling and organizing user interaction, data and information, and the frontend is responsible for the UI and the visualization of data and information.

6.1.1. Backend

The backend of the data catalog has two main building blocks. First, the CKAN instance that handles the metadata storage, search, and browsing connected through the REST API, and second, the AFS based file system based persistence backend for storing and archiving the data files, that are annotated as resources to the datasets.

CKAN metadata storage

As introduced in section 5.2.2, CKAN is an OSS web application to build data repositories. CKAN's strengths are the handling of metadata and the API based search and browse interfaces to the repository. The applications backend is based on PostgreSQL (PostgreSQL Contributors 2015), which is used as the persistence layer and backend by the CKAN instance to handle and store the metadata and the user management. The relational database schema of CKAN consists of nearly 50 tables (Open Knowledge Foundation 2014). For spatial indexing, search and further geospatial capabilities, CKAN also uses PostGIS (Ramsey et al. 2014), the spatial extension for PostgreSQL (PostgreSQL Contributors 2015). Though, only metadata are stored in the DB, it allows for GIS based queries on the spatial metadata, and map based visualization of the annotated spatial features. The data file resources itself are not stored by CKAN, they are persisted in a file system based backend, as described in the following section on the AFS based storage.

To integrate the CKAN application as a backend, the system implements an API interface called the *CKAN Action API* (Open Knowledge Foundation 2014). CKAN's Action API exposes all of CKAN's core features and functionality to API clients. This includes all of a CKAN website's core

functionality (everything you can do with the web interface and more) can be used by external code that calls the CKAN API (Open Knowledge Foundation 2016). The CKAN Action API is structured into three sub-APIs;

- i.) Model API,
- ii.) Search API,
- iii.) Util API.

The Model API also contains three sub-APIs; for i.) Model Resources, ii.) for Model Methods, and for iii.) Model Formats. The Search API, that is based on the Apache Solr (Solr Contributors 2015) technology, provides functionality for searching, browsing and filtering the CKAN database. It also contains three sub-APIs; i.) Search Resources, ii.) Search Methods, and iii.) Search Formats. This API structure stems from the overall CKAN model, that has resources available at published locations, they are represented with a variety of data formats, and each resource location supports a number of methods.

The Util API provides various utility APIs, e.g. the auto-completion API which is used by front-end JavaScript to suggest terms for auto-completing the user input (Open Knowledge Foundation 2016). All Util APIs are read-only. The response format is JSON. JavaScript calls are able to use the JSONP formatting too. CKANs API is interfaced from the Typo3 web application in a custom developed *Extbase Service* implementing the CKANs domain model in the MVC conceptual view, as further described in section 6.1.2.

AFS based long term data storage

The long term data storage is implemented and provided by the RRZK, as described in section 5.1. The data is stored in a file system folder structure, with folders for each cluster (A, B, C, D, E, F and Z, see section 1.2) and sub-folders for each of the clusters sub-projects. Theoretically, it is possible to expose the file backend of the long term data storage via AFS or Secure Shell (SSH)/SSH File Transfer Protocol (SFTP) access, to offer long term access via more low level access methods, additionally to the web based application. Practically, a University of Cologne user account and access to the Universität zu Köln LAN (UKLAN) (from external via Virtual Private Network (VPN)) is needed, to enable access to a server facing the AFS backend via SSH or SFTP. Consequently, this kind of access is only provided on request and after review of the use case.

On the server side, once the AFS file system is mounted, it behaves like any other storage device under Linux, and can be accessed through standard file system operations. The Linux file system mount is facilitated by an AFS token, that is automatically renewed every 24 hours by a server side cron job. The quota of individual folders is limited to 8 GB, for maintenance reasons of the RRZK SAN file system management, thus in each project folder, enumerated sub-folders are created, starting at 000, 001, 002, etc., see listing 6.1. The overall storage capacity of their file backend is not limited in theory, additional 8 GB volumes can be requested as needed from the RRZK.

The data files, uploaded through the data catalog application, see section 6.1, are stored automatically depending on the uploaders project affiliation into the according project folder. This is facilitated by a custom developed interface, that handles upload requests for the file backend. The interface is developed in PHP and installed on the `sf806srv` server (see section 5.1, which has access to the CRC 806 AFS based file system storage). This script has a simple algorithm that, gets some parameters additional to the file handle for the upload, as input data. The parameters are the cluster and the sub-project, the file is related to. The script tests the first subfolder in the according sub-project folder, where enough space is left for the currently uploaded file. If the space left in the current upload directory is less than 1 GB, the CRC806-Database administrators will be notified, to request an additional folder at the RRZK.

Listing 6.1: Folder structure of the AFS based long term storage.

```
.
|-- A/
|-- B/
|-- C/
|-- D/
|-- E/
|-- F/
'-- Z/
    |-- Z1/
    |-- Z2/
    | |-- 000/
    | |-- 001/
    | '-- 002/
    |   |-- FileA
    |   |-- FileB
    |   '-- FileC
    |-- Z3/
    '-- Z4/
```

The RRZK takes care of the file systems security and data backup. This is at first maintained due to a redundant RAID setup of the SAN storage backend, that allows hardware failure without data loss due to hot spare hard disk replacement possibilities. And additionally the file system is secured by incremental backups of all data on tape storage facilities provided by the RRZK.

6.1.2. Middleware

The Typo3 application tier, meaning the Typo3 CMS and all its extensions, use MySQL (Oracle Inc. 2015) as its persistence layer and backend. The MySQL backend is centrally provided, maintained and securely backed-up by the RRZK. The Extbase framework automatically creates and manages MySQL database tables to persist any application data, like relations, users and any content, unless a different backend is specified. For example the CKAN based metadata, and the metadata of the GeoNode based SDI are not persisted in the MySQL backend. As described in section 5.2.1, where the Typo3 instance is maintained by the Z2 project staff, the MySQL instance is maintained by the RRZK, though the Typo3 instance has full access rights on the database for the CMS and Extension persistence.

As introduced in section 3.3.1, 5.1, and 5.2.2 the data catalogue of the CRC806-Database is based on a CKAN instance, that facilitates together with the model and controller part of the data catalogue Typo3 extension the middleware part of the data catalogue and publication database application. The CKAN instance handles the metadata, as well as tools and interfaces for search and discovery, of the resources held by the CRC806-Database. For this task it provides mature tools and features and a set of standard Data Catalog Vocabulary (DCAT) (Maali et al. 2014) based metadata elements for organizing the resources. In CKAN every dataset can contain many (0 or more) resources. These resources are either files or links (a URL pointing to an externally hosted resource). Metadata is assigned per dataset, this way a dataset serves as a container for handling one to many resources.

An *Extbase & Fluid* (see section 5.2.1) based extension was implemented to interface the CKAN REST API for integration into the CRC806-Database web application. The controller and the model are the middleware, the view and UI part of the extension is described in the frontend section (6.1.5).

An *Extbase & Fluid* Extension is organized in a strictly defined file structure, as shown in listing 6.2, currently consisting of 26 directories, and 100 files. This complex structure is because some parts of the Extension logic is defined from its file and folder structure. This has the advantage, that by following this structure, a lot of logic, that has normally to be implemented in programming code is not needed to be implemented, because its implicit from the file system structure and naming conventions.

Listing 6.2: File tree structure of the Extbase & Fluid Data extension.

<pre> . -- Classes/ -- Controller/ -- CrcDataController.php -- DoiController.php -- ImportController.php -- LogController.php -- MaintainershipController.php '-- StatsController.php -- Domain/ -- Model/ -- Accessrequest.php -- Dataset.php -- Doirequest.php -- Edithistory.php -- License.php -- Log.php -- Maintainancerequest.php -- Region.php -- Relation.php -- Resource.php '-- Temporal.php '-- Repository/ -- AccessrequestRepository.php -- DatasetRepository.php </pre>	<pre> -- DoirequestRepository.php -- EdithistoryRepository.php -- LicenseRepository.php -- LogRepository.php -- MaintainancerequestRepository.php -- RegionRepository.php -- RelationRepository.php -- ResourceRepository.php -- '-- TemporalRepository.php -- Service/ -- CkanService.php -- DoiService.php -- '-- RelatedService.php -- Task/ -- '-- ReminderTask.php -- Tools/ -- Benchmark.php -- CkanPackageBuilder.php -- DateTimeProvider.php -- Diff.php -- MapBuilder.php -- MetadataProvider.php -- '-- RelatedDataManager.php -- ViewHelpers/ -- -- IsArrayViewHelper.php -- Configuration/ -- TCA/ -- tx_unikoeInrcrdata_domain_model_accessrequest. </pre>
---	---

```

php
| |-- tx_unikoelnrcrcdata_domain_model_dataset.php
| |-- tx_unikoelnrcrcdata_domain_model_doirequest.php
| |-- tx_unikoelnrcrcdata_domain_model_edithistory.php
| |-- tx_unikoelnrcrcdata_domain_model_license.php
| |-- tx_unikoelnrcrcdata_domain_model_log.php
| |--
|   tx_unikoelnrcrcdata_domain_model_maintainancerequest
|   .php
| |-- tx_unikoelnrcrcdata_domain_model_region.php
| |-- tx_unikoelnrcrcdata_domain_model_relation.php
| |-- tx_unikoelnrcrcdata_domain_model_resource.php
| '-- tx_unikoelnrcrcdata_domain_model_temporal.php
|-- ext_emconf.php
|-- ext_icon.gif
|-- ext_localconf.php
|-- ext_tables.php
|-- Resources/
| |-- Private/
| | '-- Templates/
| | |-- CrcData/
| | | |-- ApproveAccessRequest.html
| | | |-- Create.html
| | | |-- Delete.html
| | | |-- Edit.html
| | | |-- EditMap.html
| | | |-- EditResources.html
| | | |-- ExportBibtex.html
| | | |-- ExportRdf.html
| | | |-- GetJson.html
| | | |-- GetResource.html
| | | |-- List.html
| | | |-- ShowDatamap.html
| | | |-- ShowHistory.html
| | | '-- Show.html
| | |-- Doi/
| | | |-- Approve.html
| | | |-- List.html
| | | '-- Request.html
| | |-- Import/
| | | |-- ImportRegions.html
| | | |-- ImportTemporals.html
| | | '-- MigrateOldResources.html
| | |-- Log/
| | | |-- List.html
| | | '-- Show.html
| | |-- Maintainership/
| | | |-- ApproveMaintenanceRequest.html
| | | |-- List.html
| | | '-- RequestMaintenance.html
| | '-- Stats/
| | '-- Show.html
|-- Public/
|-- css/
| |-- jquery.bonsai.css
| |-- pikaday.css
|-- Icons/
| |-- mrelation.gif
|-- js/
| |-- chart.js
| |-- createData.js
| |-- d3.js
| |-- editData.js
| |-- jquery.bonsai.js
| |-- jquery.fileupload.js
| |-- jquery.iframe-transport.js
| |-- jquery.qubit.js
| |-- jquery.ui.widget.js
| |-- listData.js
| |-- moment.js
| |-- pikaday.js
| |-- relatedFunctions.js
|-- xssPagination.js
'-- Tests/
    '-- Unit/
        |-- CkanPackageBuilderTest.php
        '-- CkanServiceTest.php

```

The controller contains all logic and functionality to handle requests triggered through user interaction from the frontend. Communication with CKAN API is implemented as a service, which are mostly used by the models part of the extension (contained in the Domain folder, see listing 6.2), where are also some additional models reside for persisting internal relations and annotations, that are not directly stored in the CKAN database.

The Typo3 *Extbase & Fluid* Data extension interfaces CKAN as the metadata backend for the data catalog. Figure 6.2 shows the MVC based architecture of the Extension. The user interacts with the UI of the data catalog, that is rendered by the View, any interaction with the UI is processed by the view and triggers (1) the controller, e.g. show dataset with ID X, the controller then triggers (2) the model, for accessing the backend to look up ID X, in this case the backend is CKAN, and a so called *CKAN Service*, that was implemented for interfacing the CKAN API (see section 5.2.2) without interaction with the Extbase model, is triggered (3) to send an according

request to the CKAN server. The CKAN server sends back a response (4) in JSON format. The data extensions model parses the response and formulates (5) an object the view component can render (6) as a response to the users request.

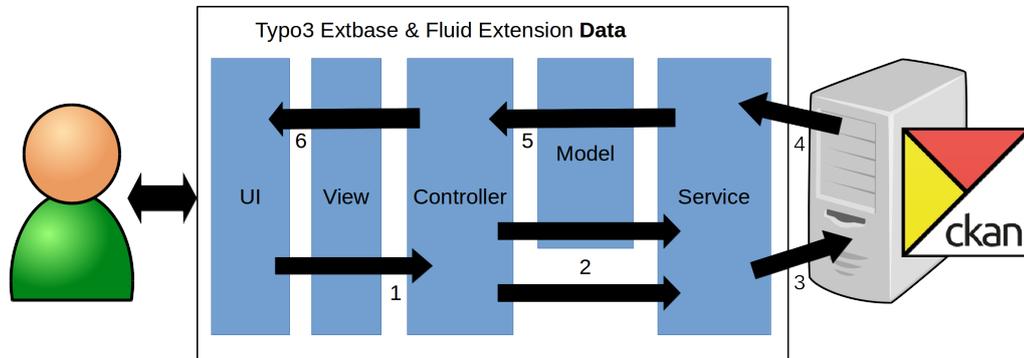


Figure 6.2.: MVC based architecture of Typo3 Extbase & Fluid Data extension.

The advantage of having all CKAN related functionality implemented in a distinct service, allows for reuse of the functionality within other extensions, that was taken advantage for by the publication DB extension (see section 6.2).

The filter functionality of the data catalogue interface is based on distinct CKAN queries. It is possible to facet the filters, meaning be able to chain and combine filters to refine the result. For example, first query for a string in the full-text search and then filter by cluster or project, to further refine the result set, see section 9.2 for further details on the catalogue interface and its filters..

Another notable feature, that was implemented in the middleware layer is the related datasets feature. This feature allows to relate datasets, maps and publications to each other, see also section 6.1.7. Related resources are shown on each of the dataset, maps and publications detail pages. The relations are stored in an extra database table by the Extbase extension. This information is thus not persisted in the CKAN backend.

6.1.3. Metadata

As introduced in section 2.2.1, metadata annotation is crucial for interoperability and reuse of the data in a repository like the CRC806-Database. The metadata schema of CKAN datasets is based on the DCAT vocabulary. DCAT is an RDF vocabulary designed to facilitate interoperability between data catalogues published on the Web (Maali et al. 2014). The basic meta data schema of CKAN is given in table 6.2.

DCAT is used by CKAN as default vocabulary for describing its resources. The vocabulary makes extensive use of terms from other vocabularies, in particular Dublin Core (Maali et al. 2014). The official W3C recommendation for the DCAT vocabularies, states some very clear criteria that a data catalogue must implement to be conform with DCAT:

- It is organized into datasets and distributions.

- An RDF description of the catalog itself and its datasets and distributions is available.
- The contents of all metadata fields that are held in the catalog, and that contain data about the catalog itself and its dataset and distributions, are included in this RDF description, expressed using the appropriate classes and properties from DCAT, except where no such class or property exists.
- All classes and properties defined in DCAT are used in a way consistent with the semantics declared in this specification.
- DCAT-compliant catalogs may include additional non-DCAT metadata fields and additional RDF data in the catalog's RDF description.

Table 6.2.: CKAN metadata schema. Source: Open Knowledge Foundation (2014).

Term	Description
Title	Allows intuitive labeling of the dataset for search, sharing and linking.
Unique Identifier	Dataset has a unique URL which is customizable by the publisher.
Groups	Which groups the dataset belongs to if applicable. Groups (such as science data) allow easier data linking, finding and sharing amongst interested publishers and users.
Description	Additional information describing or analyzing the data. This can either be static or an editable wiki which anyone can contribute to instantly or via admin moderation.
License	View of whether the data is available under an open license or not. This makes it clear to users whether they have the rights to use, change and re-distribute the data.
Tags	What labels the dataset in question belongs to. Tags also allow for browsing between similarly tagged datasets in addition to enabling better discover-ability through tag search and faceting by tags.
Formats	The different formats the data has been made available in quickly in a table, with any further information relating to specific files provided inline.
API Key	Allows access every metadata field of the dataset and ability to change the data if you have the relevant permissions via API.
Extra fields	Hold any additional information, such as location data (see geospatial feature) or types relevant to the publisher or dataset. How and where extra fields display is customizable.
Revision History	CKAN allows you to display a revision history for datasets which are freely editable by users.

The CRC806-Database CKAN backend implements all these criteria, and is DCAT conformal.

Through proxying the CKAN functionality through the data catalog Typo3 extension, and also proxying the CKAN RDF endpoint, the CRC806-Database also implements the DCAT schema.

It is possible to request the metadata for each dataset in RDF and many further serialization formats. The different formats, given in table 6.3, can be accessed by calling an URL of the following form:

```
https://{ckan-instance-host}/datasets/{dataset-id}.{format}
```

Table 6.3.: Metadata Serialization formats available from CKAN.

Extension	Format	Media Type
.xml	RDF/XML	application/rdf+xml
.rdf	RDF/XML	application/rdf+xml
.ttl	Turtle	text/turtle
.n3	Notation3	text/n3
.jsonld	JSON-LD	application/Id+json

The RDF/XML and JSON-LD formats deliver proper LinkedData, and thus implement a part of the CRC806-Database LinkedData concept.

6.1.4. Assignment and issuing of DOIs

DOIs can be assigned to any dataset uploaded and published via the CRC806-RDM infrastructure data catalog interface. The process of assigning a DOI is simple and can be achieved in a few steps:

1. Create a dataset.
2. Link and upload resources.
3. Annotate the minimal set of Metadata according to the DataCite Schema (see section 2.4.3).
4. Request for minting a DOI.
5. Evaluation of Dataset according to the CRC806-Database DOI policy.

First, a dataset via the CRC806-RDM application has to be created. Every CRC806-Database user can do this, by providing according information in a user friendly web based form. The second step, to add resources to the dataset, is the most important step of the procedure. Because the dataset is just the container for the specific data content, that is given through the according resources. The third step, the annotation of metadata is crucial for the documentation of the data, and for its reuse. A minimal set of metadata is mandatory, according to the DataCite schema (DataCite Metadata Working Group 2011) and has to be provided. Additionally, the data uploading user is encouraged to annotate more metadata, for example a spatial and temporal context, if available. Steps five and six are of technical and administrative nature. Here the data is evaluated for meeting the minimal requirements for minting a DOI, according to the DataCite policy and the according DOI standard (NISO 2010).

The CRC806-Database DOI policy is sober in the sense of minimalistic, and thus practicable. Any dataset stored in the CRC806-Database can be assigned a DOI, as long as its annotated with the minimum set of metadata, required by the DataCite schema (DataCite Metadata Working Group 2011), and as long as there is only one DOI per resource.

As described in section 2.4.3, the CRC806-Database cooperates with the Deutsches GeoForschungsZentrum (GFZ) Potsdam for registration and minting of DOI's. The GFZ maintains a service called DOIDB (Ulbricht et al. 2016), that acts as a proxy service to the main DataCite metadata store. This DOIDB system offers a web interface for minting DOIs through a web form, and an API interface, to register DOI's automatically, for example from within a web application. Both methods require a metadata record, describing the dataset that will be assigned with a DOI, formulated in the DataCite Schema XML syntax. In listing 6.3 a real world example metadata set for issuing a DOI through the DataCite API is given.

Listing 6.3: Example XML Metadata for issuing a DOI, DataCite Schema 3.0.

```
<?xml version="1.0" encoding="UTF-8"?>
<resource xmlns="http://datacite.org/schema/kernel-3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://datacite.org/schema/kernel-3 http://schema.datacite.org/meta/kernel-3/metadata.xsd">
  <identifier identifierType="DOI">10.5880/SFB806.2</identifier>
  <creators>
    <creator>
      <creatorName>Willmes, Christian</creatorName>
    </creator>
    <creator>
      <creatorName>Becker, Daniel</creatorName>
    </creator>
    <creator>
      <creatorName>Brocks, Sebastian</creatorName>
    </creator>
    <creator>
      <creatorName>Hütt, Christoph</creatorName>
    </creator>
    <creator>
      <creatorName>Bareth, Georg</creatorName>
    </creator>
  </creators>
  <titles>
    <title>Köppen-Geiger classification of MPI-ESM-P LGM simulation</title>
  </titles>
  <publisher>CRC806-Database</publisher>
  <publicationYear>2014</publicationYear>
  <subjects>
    <subject>Climate Classification</subject><subject>Köppen-Geiger</subject><subject>Climate simulation</subject><subject>CMIP 5</subject><subject>PMIP 3</subject><subject>GRASS GIS</subject><subject>Geospatial</subject><subject>GIS</subject><subject>Paleoclimate</subject><subject>LGM</subject><subject>Last Glacial Maximum</subject>
  </subjects>
  <language>eng</language>
  <resourceType resourceTypeGeneral="Dataset">Dataset</resourceType>
  <version>1</version>
  <descriptions>
    <description descriptionType="Abstract">This geospatial dataset, in raster and vector format, is a Köppen-
```

```

Geiger climate classification of the MPI-ESM-P Last Glacial Maximum (21k yBP) r11p1 model simulations
according to the PMIP III 21k experiment. The classifications were computed using the Python pyGRASS
library and GRASS GIS.</description>
</descriptions>
</resource>

```

6.1.5. Frontend

The data catalog UI is based on Fluid (Lobacher 2014), HTML and Cascading Style Sheets (CSS) styling templates, and mainly server side interaction handling. A website layout formulated in Fluid markup is generated through templates and partials. Each view (list view, detail view, data entry, header, footer, etc.) has an own template, see figure 6.3. These templates contain placeholders for rendering dynamic informations. These placeholders can display contents directly from the model (data base) or further smaller templates, so called partials. Partials are for example header and footer, or the boxes on the side of the dataset detail pages. Elements that can occur repeatedly within an template or partial according to the content query, are rendered using the `f:` for Fluid-viewhelper (see listing 6.4 for an example).

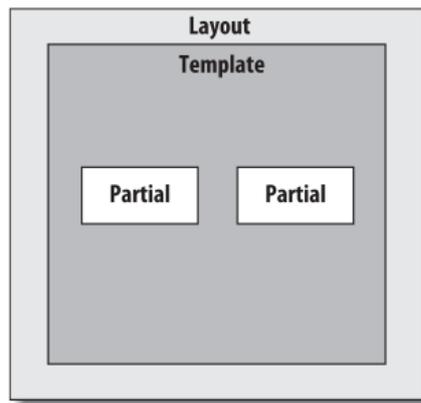


Figure 6.3.: Layouts build the outer frame for a template, whereas recurring elements can be implemented in a template with partials. Source: (Rau et al. 2013).

Listing 6.4: Example view helper, for displaying a list of blog posts.

```

<f:for each="{posts}" as="post">
<div class="flatBox">
  <div class="pull-right">
    <b>{post.publishdate}</b>
    <span class="userLink"><i class="glyphicon glyphicon-user"></i>
      <f:link.action extensionName="UnikoelnCrcusers" pluginName="Userprofile" pageUId="26"
        controller="Userprofile" action="show" arguments="{name: post.author.name}" >{post.
        author.name}</f:link.action>
    </span>
  </div><hr/>
  <h1><f:link.action action="show" pageUId="34" arguments="{post: post}">{post.title}</f:link.action></h1>
  <div><i>{post.abstract}</i> <f:link.action action="show" pageUId="34" arguments="{post: post}">[Read more
  ...]</f:link.action> <br/><hr/></div>

```

```
</div>  
</f:for>
```

In the following of this frontend section, some selected features (Datamap, Licensing and the Fife star open data ratings implementation) are presented in more detail.

Datamap

Within the data catalogue frontend, a special view element is of course the dataset map, see figure 6.4. This map is generated from the spatial metadata annotations stored in CKAN. The CKAN spatial extension handles location information in GeoJSON format (Butler et al. 2008), a geospatial data interchange format based on JSON. It is possible to query CKAN spatially, if the spatial extension is enabled, as it is the case for the CRC806-Database CKAN instance. By default only point based locations are shown on the datamap, bounding-boxes are displayed with their center point. By hovering over a center point of a bounding-box, the bounding-box is then displayed as shown in figure 6.4. This kind of visualization was chosen, because rendering all the bounding-boxes, would not be possible, because some bounding-boxes cover very large extends (continents, or the whole world) and would overlay other smaller bounding-boxes.



Figure 6.4.: Screenshot of the CRC806-Database data catalog datamap.

Some logic was developed and implemented to trigger spatial queries from interaction with the OpenLayers (OpenLayers Contributors 2015) based map, by obtaining the locations in form of points or bounding-boxes that are drawn on the map by the user while holding the Shift button. It is further possible to retrieve according metadata, including a link to their dataset page, of the displayed resources by clicking on the features.

Licensing

The users are able to define CC license information of the according dataset in a simple interfaces based on some few check boxes, see figure 6.5. According to this information, the resulting CC license, including its logo, are applied to the resources and displayed on the dataset detail page.

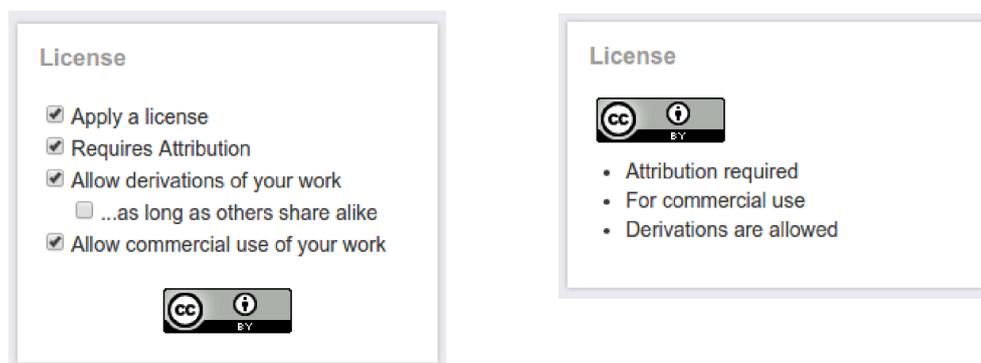


Figure 6.5.: UI element for license editing (left) and display (right) on the dataset pages.

It is possible to choose between possible CC licenses, by just answering five questions (as shown in fig. 6.5 on the left side) with Yes or No. As indicated also in the figure, Yes or No answers can be implemented by simple check boxes.

Five star open data

To annotate, and in some sense also reward, the degree of openness of a dataset published in the CRC806-Database, the 5 Star linked open data scheme (see section 2.2.4 and figure 2.3), introduced by Tim Berners-Lee was implemented. Each dataset is automatically reviewed according to the 5 Star linked open data definitions, and accordingly assigned a logo displaying the resulting number of Stars on the dataset detail page (see table 6.4). A program routine implemented in the Extbase controller of the data catalog, checks the criterias for assigning the logo badges. The one star badge is assigned, if the dataset is assigned with a license, and at least one resource of the dataset is accessible to the public. The two star badge is assigned, if at least one resource is in structured form. That are for example, CSV, Excel, Shapefile, etc. resources. Three stars are assigned, if the data is available in an open format, for example CSV or OWS. Four stars are assigned if the metadata is assigned with linked URI's, this is form example the case for OWS. And five stars are assigned if the data is available as Linked Data, for example as RDF or on a web page that contains RDFa markup of the actual data resource (not only the metadata).

6.1.6. Linked Data

As introduced in section 4.6, the HTML code of the CRC806-Database web application is extended using RDFa technology, to implement Linked Data concepts. The data catalogs HTML

Table 6.4.: Five star open data logo badges.

<ul style="list-style-type: none"> • Open License 	<ul style="list-style-type: none"> • Open License • Structured 	<ul style="list-style-type: none"> • Open License • Structured • Open Format 	<ul style="list-style-type: none"> • Open License • Structured • Open Format • URI's 	<ul style="list-style-type: none"> • Open License • Structured • Open Format • URI's • Linked Data

markup is enhanced with RDFa (Adida et al. 2015) annotations. The main reason is to support interoperability of the data catalog and the overall web application, by providing the metadata definitions directly with the HTML markup of the dataset pages, as well as the accordance and support to the Linked Data paradigm (see section 2.2.4). The metadata items of the data catalog are annotated using RDFa technology with well known vocabularies, as given in table 6.5.

Table 6.5.: Vocabularies applied for RDFa in the CRC806-Database system.

Vocabulary	Description	Reference
DCAT	Metadata for data catalog and implemented through CKAN	(Maali et al. 2014)
Dublin Core	Basic metadata for describing datasets	(Dublin Core Metadata Initiative 2004)
Bibo	Bibliographic Ontology, annotating the bibliographic metadata.	(D’Arcus et al. 2009)
Schema.org	Collaborative, community activity with a mission to create, maintain, and promote schemas for structured data on the Internet, on web pages, in email messages, and beyond	(Brickley et al. 2016)
Open Organizations	This vocabulary provides supplementary terms for organizations wishing to publish open data about themselves.	(Gutteridge et al. 2013)
Creative Commons Rights Expression Language	The Creative Commons Rights Expression Language (CC REL) lets you describe copyright licenses in RDF.	(Creative Commons Developers 2016)

RDFa implementation

The dataset detail pages are listing all the metadata annotations in HTML markup. This markup is enhanced with RDFa based links to well known vocabularies. In figure 6.6 an example rendering of bibliographic metadata, as can be found on data catalogue dataset detail pages is shown.

The vocabulary prefixes are defined in the <body> tag of the data catalogs HTML web sites (see listing 6.5).

Listing 6.5: RDFa vocabularies included in the data catalogue pages.

Bibliography

Becker, D., Verheul, J., Zickel, M., Willmes, C. (2015): LGM paleoenvironment of Europe - Map. CRC806-Database, DOI: 10.5880/SFB806.15

Authors	Becker, Daniel and Verheul, Jan and Zickel, Mirijam and Willmes, Christian
Type	Dataset
Title	LGM paleoenvironment of Europe - Map
DOI	10.5880/SFB806.15
Year	2015
Publisher	CRC806-Database

[Export BibTeX](#)

Figure 6.6.: Screenshot of data catalog example bibliography metadata information.

```
<body prefix="schema: http://schema.org/
  bibtex: http://purl.oclc.org/NET/nknouf/ns/bibtex#
  dcat: http://www.w3.org/ns/dcat#
  oo: http://purl.org/openorg/
  dct: http://dublincore.org/documents/dcmi-terms/
  cc: http://creativecommons.org/ns#">
```

The RDFa annotation of metadata for this example datasets detail page is given in listing 6.6.

Listing 6.6: Example RDFa on data catalogue sites.

```
<div id="bibliography" class="flatBox">
<h2>Bibliography</h2>
<p style="font-size: 0.9em;">Becker, D., Verheul, J., Zickel, M., Willmes, C. (2015): LGM paleoenvironment of
  Europe - Map. CRC806-Database, DOI: 10.5880/SFB806.15</p>
<table class="metadatatable">
  <tr class="datarow">
    <th>Authors</th>
    <td><span property="schema:author">Becker, Daniel</span> and <span property="schema:author">Verheul
      , Jan</span> and <span property="schema:author">Zickel, Mirijam</span> and <span property="
      schema:author">Willmes, Christian</span></td>
  </tr>
  <tr class="datarow">
    <th>Type</th>
    <td>Dataset</td>
  </tr>
  <tr class="datarow">
    <th>Title</th>
    <td>LGM paleoenvironment of Europe - Map</td>
  </tr>
  <tr class="datarow">
    <th>DOI</th>
    <td><a href="http://dx.doi.org/10.5880/SFB806.15">10.5880/SFB806.15</a></td>
  </tr>
  <tr class="datarow">
    <th>Year</th>
    <td><span property="bibtex:hasYear">2015</span></td>
  </tr>
  <tr class="datarow">
    <th>Publisher</th>
    <td><span property="schema:publisher">CRC806-Database</span></td>
  </tr>
</table>
```

```
[...]
</div>
```

RDF export

The data catalog interface provides also the possibility to export the full metadata record of the datasets in RDF format. This functionality is provided by CKAN, that provides export of the metadata in different formats, as described in section 5.2.2. A shortened example of such an RDF export is given in listing 6.7.

Listing 6.7: Example RDF export.

```
<rdf:RDF xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:owl="http://www.w3.org/2002/07/owl#" xmlns:rdfs="http://
www.w3.org/2000/01/rdf-schema#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:dcat="http
://www.w3.org/ns/dcat#" xmlns:dct="http://purl.org/dc/terms/">
<dcat:Dataset rdf:about="http://sfb806ckan.geographie.uni-koeln.de/ckan/dataset/lgm-major-inland-waters-of-
europe--gis-dataset1449846174">
  <owl:sameAs rdf:resource="urn:uuid:d4a383e9-fc2e-4d1b-bc9d-6f5c300b1638"></owl:sameAs>
  <dct:description>This is GIS dataset contains the major inland waters (rivers and lakes) of Europe during
  the LGM. The data was collected from data published in scholarly works. </dct:description>
  <foaf:homepage rdf:resource="http://sfb806ckan.geographie.uni-koeln.de/ckan/dataset/lgm-major-inland-waters-
of-europe--gis-dataset1449846174"></foaf:homepage>
  <rdfs:label>lgm-major-inland-waters-of-europe--gis-dataset1449846174</rdfs:label>
  <dct:identifier>lgm-major-inland-waters-of-europe--gis-dataset1449846174</dct:identifier>
  <dct:title>LGM major inland waters of Europe - GIS dataset</dct:title>
  <dcat:distribution>
    <dcat:Distribution>
      <dcat:accessURL rdf:resource="http://afs/rrz.uni-koeln.de/project/sfb806db/Phase_1/Z/Z2/data/000/
LGM_Lakes.zip"></dcat:accessURL>
      <dct:format>
        <dct:IMT>
          <rdf:value>ZIP</rdf:value>
          <rdfs:label>ZIP</rdfs:label>
        </dct:IMT>
      </dct:format>
      <dct:title>LGM_Lakes.zip</dct:title>
    </dcat:Distribution>
  </dcat:distribution>
  [...]
  <dct:creator>
    <rdf:Description>
      <foaf:name>Christian Willmes</foaf:name>
      <foaf:mbox rdf:resource="mailto:c.willmes@uni-koeln.de"></foaf:mbox>
    </rdf:Description>
  </dct:creator>
  [...]
  <rdf:Description>
    <rdfs:label>bibtex:author</rdfs:label>
    <rdf:value>Verheul, Jan and Zickel, Mirijam and Becker, Daniel and Willmes, Christian</rdf:value>
  </rdf:Description>
</dct:relation>
[...]
  <rdf:Description>
    <rdfs:label>bibtex:doi</rdfs:label>
    <rdf:value>10.5880/SFB806.14</rdf:value>
```

```

    </rdf:Description>
  </dct:relation>
  [...]
  <dct:relation>
    <rdf:Description>
      <rdfs:label>bibtex:publisher</rdfs:label>
      <rdf:value>CRC806-Database</rdf:value>
    </rdf:Description>
  </dct:relation>
  [...]
  <dct:relation>
    <rdf:Description>
      <rdfs:label>crc806authors</rdfs:label>
      <rdf:value>Jan Verheul,Mirijam Zickel,Daniel Becker,Christian Willmes</rdf:value>
    </rdf:Description>
  </dct:relation>
  <dct:relation>
    <rdf:Description>
      <rdfs:label>license</rdfs:label>
      <rdf:value>CC BY</rdf:value>
    </rdf:Description>
  </dct:relation>
  <dct:relation>
    <rdf:Description>
      <rdfs:label>spatial</rdfs:label>
      <rdf:value>{"type":"Polygon","coordinates":[[[-7,80],[68,80],[68,30],[-7,30]]]}</rdf:value>
    </rdf:Description>
  </dct:relation>
  [...]
</dcat:Dataset>
</rdf:RDF>

```

6.1.7. Related resources

Another feature implemented to integrate resources of the data catalog, with resources from the publications DB and the SDI, is the *related resources* feature. Figure 6.7 shows an example screenshot, of how related resources are rendered on a dataset detail page of the data catalogue.

A web UI was developed to choose resources published within the CRC806-Database RDM infrastructure to a given dataset. The relations will be annotated and shown both way. For example if a spatial dataset is annotated to a dataset of the data catalogue, the data catalogue dataset will be listed on the spatial datasets detail page in the SDI frontend too. The relations are modeled in an Extbase model and persisted in the MySQL Typo3 backend. This is one of the data annotations, that are only available from the Typo3 web application interfaces of the CRC806-Database.

6.2. Publications

As a central point for the storage and management of publication records produced by the CRC 806, a publication database was implemented. This application functions, in some sense, as a

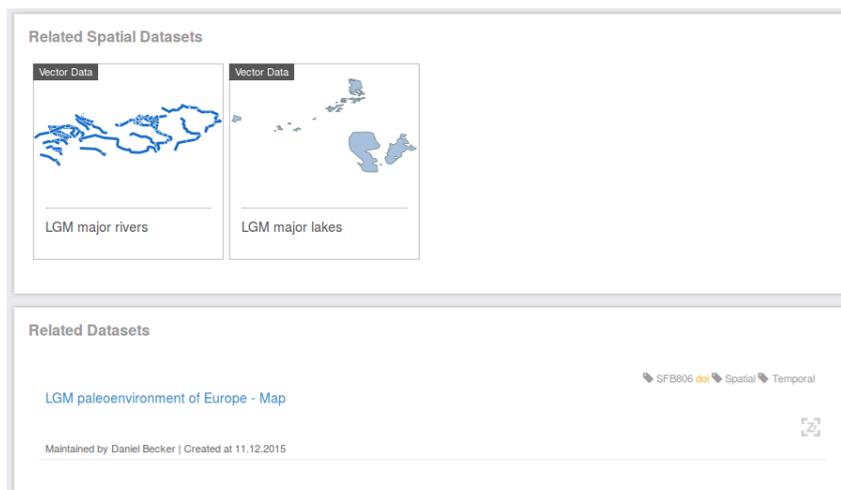


Figure 6.7.: Screenshot of data catalogue example related resources of a dataset.

display window to the public, for the published results of the CRC 806. If someone from outside the CRC 806 is interested in the research results of the project, the publication DB offers a good overview. The publication database also uses the CKAN instance as the data backend. Similar to the data catalog interface, the publication records are stored in the CKAN backend as datasets. These datasets can have resources, also similar to the datasets, that can be links to external publisher websites or for example the PDF file of the article and any other supplementary data, stored in the AFS based CRC806-Database backend. The publications can be related to datasets and geo-datasets of the CRC806-Database, this is implemented in the Exibase domain model of the data catalog, as described in section 6.1.7. It is also possible to tag publication records with topic categories, or keywords, to allow filtering among them in the database. The augmentation with spatial and temporal metadata, similar to the annotations in the data catalog is possible as well. This allows spatio-temporal filtering of literature records as well.

6.2.1. Bibliography model

The bibliography model of the publication database is based on the BibTeX schema, introduced by Patashnik (1988) in its documentation of the BibTeX package known from the LaTeX word processor community for managing bibliographies. BibTeX was chosen, because it proved to be successful in practice, almost any scientific publication platform and most desktop tools for bibliography management support the BibTeX format for exporting and/or importing bibliographic data. The BibTeX Schema is structured along so called *Entry Types* and according *Fields* (see table 6.6). The fields are distinguished for each entry into mandatory and optional fields, table 6.6 shows the schema in detail.

The publication entry interface of the CRC806-Database adapts its web forms, according to the chosen *Entry Type* automatically, by offering only the mandatory and optional fields, as shown in figure 9.4. An example bibliographic record formulated in BibTeX format is shown in listing 6.8.

Table 6.6.: The BibTeX Schema.

Entry Type	Mandatory Fields	Optional Fields	Description
article	author, title, journal, year	volume, number, pages, month, note	A (peer-reviewed) journal article.
book	author or editor, title, publisher, year	volume or number, series, address, edition, month, note, isbn	A book.
booklet	title	author, howpublished, address, month, year, note	A work that is printed and bound, but without a named publisher or sponsoring institution.
conference	author, title, booktitle, year	editor, volume or number, series, pages, address, month, organization, publisher, note	Scientific Conference.
inbook	author oder editor, title, chapter and/or pages, publisher, year	volume or number, series, type, address, edition, month, note	Part or Chapter of a Book.
incollection	author, title, booktitle, publisher, year	editor, volume or number, series, type, chapter, pages, address, edition, month, note	A part of a book having its own title.
inproceedings	author, title, booktitle, year	editor, volume oder number, series, pages, address, month, organization, publisher, note	Article in a conference proceedings publication.
manual	address, title, year	author, organization, edition, month, note	Technical documentation.
mastersthesis	author, title, school, year	type, address, month, note	A masters or diploma thesis.
misc	-	author, title, howpublished, month, year, note	For use when nothing else fits.
phdthesis	author, title, school, year	type, address, month, note	A Ph.D. thesis.
proceedings	title, year	editor, volume oder number, series, address, month, organization, publisher, note	A conference proceedings publication.
techreport	author, title, institution, year	type, note, number, address, month	Published report or a Standard document.
unpublished	author, title, note	month, year	Not formally published.

6.2.2. Typo3 Extension

On the CRC806-Database website the CKAN based publication database is interfaced by Typo3 Extbase & Fluid extension. The extension stores all metadata of the publication records in the CKAN database, for which it reuses the CKAN service, originally developed for the data catalog extension, which is reused by the Publications DB application.

The Publications interface, implemented as an additional view template, on the main website contains similar filters (full-text search, sorting, project filter, location, time and tags) like the data catalog interface. The detail views of the publications records are also quite similar to the detail views of the data catalog interface.

The bibliographic records of any data resource can be exported in BibTeX format, see listing 6.8 for an example. This feature allows easy integration into almost any state-of-the-art bibliography management software, because the format is widely adopted as described above.

The possibility to reuse the spatial and temporal annotation functionalities of the data catalog enables this features also for the publications DB. Although, the spatio-temporal filtering is only available from the CRC806-Database publication DB interface. Another useful feature, that also applies for publication records, is the *related datasets* feature, as described in section 6.1.7.

Listing 6.8: Example BibTeX entry.

```
@Article{Willmes2014,
  Title = {Building Research Data Management Infrastructure using Open Source Software},
  Author = {Willmes, Christian and K\"urner, Daniel and Bareth, Georg},
  Journal = {Transactions in GIS},
  Year = {2014},
  Pages = {496 - 509},
  Volume = {18},
  Doi = {10.1111/tgis.12060},
  ISSN = {1467-9671},
  Url = {http://dx.doi.org/10.1111/tgis.12060}
}
```

6.2.3. Interfaces to sfb806.de and sfb806irtg.uni-koeln.de

In order to maintain only one central “point of entry”, and avoid redundant entry of publication records for researchers and projects within the CRC 806, a publication database including interfaces to the three websites of the research centre (see figure 6.8):

- CRC 806 Main Website, <http://sfb806.de>,
- CRC 806 IRTG, <http://sfb806irtg.uni-koeln.de/>,
- CRC806-Database, <http://crc806db.uni-koeln.de>,

of the CRC 806 was requested for implementation by the project partners, and because this approach was a success and positive experience from the comparable TR32DB (<http://tr32db>).

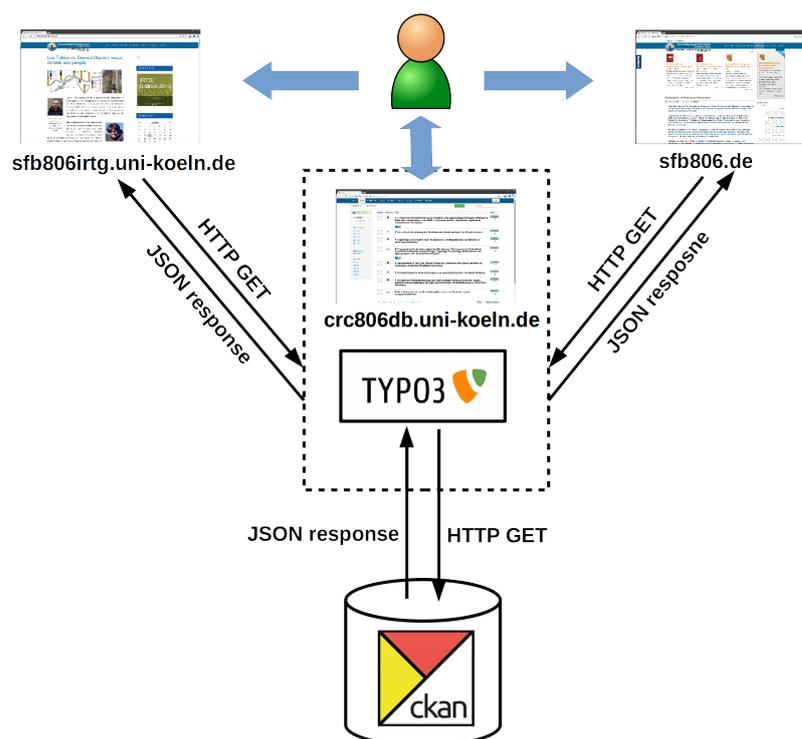


Figure 6.8.: Interfaces of the Bibliographic Database.

uni-koeln.de (Curdt et al. 2012)) interface implementation for the CRC / Transregio 32 web presence (<http://tr32.de>).

As shown in figure 6.8, publication records are entered and edited centrally via the interfaces of the CRC806-Database. The interface of the Integrated Research Training Group (IRTG) website connects directly to an endpoint implemented in the Typo3 extension, to retrieve the data from the publication DB.

Main CRC 806 website

The main CRC 806 website (<http://www.sfb806.de>) is developed by the Z2 project, based on Joomla CMS (Open Source Matters, Inc. 2015). To built an interface to the CRC806-Database publication store, a small Joomla extension was developed. It is that small, it can't be classified as a full extension, because the interface consists of small code snippets, that are included into the Joomla pages. These small snippets, see listing 6.9, get placed on each sub page of the website, that should show publication records, by calling an endpoint <http://crc806db.uni-koeln.de/dataset/getJson> that delivers all metadata of the CRC806-Database in JSON format.

Listing 6.9: PHP snippet for rendering the publications list.

```
<?php
$data = requestData('literature', 'byProject', 'z2');
echo makeLegend();
```

```

echo makeList($data);
function requestData($source = "all", $filter = "all", $filterValue = "") {
    $fq = '';
    if($source === 'literature' || $source === 'datasets')
        $fq .= '?datatype=' . $source;
    if($filter === 'byProject' || $filter === 'byCluster')
        $fq .= (strlen($fq) > 0 ? '&':'?') . 'project=' . strtolower($filterValue);
    if($filter === 'byUser')
        $fq .= (strlen($fq) > 0 ? '&':'?') . 'author=' . urlencode($filterValue);
    return json_decode(file_get_contents("http://crc806db.uni-koeln.de/dataset/getJson" . $fq), true)['result
        '][ 'results'];
}
function makeLegend() {
    return '<div style="font-size: 8pt; width:30px;float:left;">bold:</div><div style="float: left;font-size:
        8pt;font-weight: bold;">peer-reviewed</div>
        <div style="padding-left:30px; font-size: 8pt; width:100px;float:left;">asterisk (*):</div><
        div style="float: left; font-size: 8pt; font-weight: bold;">ISI-indexed</div><br style="
        clear: both;" /><br />';
}
function makeList($data, $class = '') {
    if(count($data) <= 0)
        return '<div>Currently no publications available.</div>';
    $list = '<ul style="list-style: none; padding: 0;">';
    if(strlen($class) > 0)
        $list = '<ul class="' . $class . '">';
    $count = 0;
    foreach($data as $row)
        $list .= '<li style="margin-bottom: 20px;font-family: Helvetica, Arial, sans-serif"><a href="' .
            $row['url'] . '" target="_blank">' . $row['citation'] . '</a></li>';
    $list .= '</ul>';
    return $list;
}
?>

```

The data is obtained from the database by calling the method `requestData()`. The method has three parameters `requestData(informationType, filter, filterValue)`, where `informationType` can be `literature` or `data`, to distinguish of retrieving publications or research data. The filter parameter can hold the values of `byProject`, `byCluster` or `byUser`, these different filters can take according `filterValue`, projects and cluster get according project and cluster names, and `byUser` can get an author name as filter value. For example a call to `requestData('datasets', 'byProject', 'z2')`; renders all research data sets from project Z2 in a list and the call to `requestData('literature', byUser', 'Christian Willmes')`; renders all publications of Christian Willmes as a list. Its also possible to get all publications (`requestData('literature', all', '-')`);, if the first parameter is replaced by `'data'` it retrieves all research data sets, also even all metadata records can be retrieved as a list, by calling `requestData('all', all', '-')`;

IRTG website

For the WordPress (Wordpress Contributors 2014) based CRC 806 IRTG website, a WordPress plug-in was developed, to show publication lists of PhD students. Here another approach, com-

pared to the main website, was chosen, obviously due to different CMS technology. The whole rendering of the publication lists is implemented on the WordPress side in the plug-in. The plug-in is interfacing the `getJson` Endpoint of the CRC806-Database, as described for the interface to the main CRC 806 website in the section above, and retrieves the publication lists as JSON objects. The JSON objects are then handled in the plug-in and the contents are rendered accordingly. The plug-in defines a marker, that can be included into WordPress pages. The marker has the following syntax:

`{literature:<filter>:<parameter>}`

Three filters are implemented; `byAuthor`, `byCluster`, and `byProject`. To retrieve a list of all publications of the author 'Christian Willmes', one would include the following tag into the Wordpress editor: `{literature:byAuthor:Christian Willmes}`. To retrieve all publications of Cluster Z, the following tag would do: `{literature:byCluster:z}`. And the tag: `{literature:byProject:b1}`, would get a list of all publications of project B1, for example.

6.3. Members directory

The CRC806-Database members directory is also developed as an *Extbase & Fluid* extension, building upon, and extending, the previously for the former members directory used `fe_users` (front-end users) extension of Typo3. This means, that the existing user data base could be reused from the new implementation. This was important, to not have to ask the users to register again for a new account. Additionally, all user data, registration and profile data, as well as some functionality is interfaced from the `fe_users` extension. The frontend renderings, of the user directory and profile pages, as well as the filters of the user list and the according dataset lists on the user pages are implemented from scratch using *Extbase & Fluid* technology.

Each user has a profile page displaying his or her contact details, a list of the datasets the user is author of and a list the user is maintainer of. These lists are generated from queries of the CKAN metadata backend, the contact details are stored in the *Extbase & Fluid* based data model of the extension. If the user is logged in, he or she can access the edit profile page, to modify the personal informations.

The members list, rendered on the start page of the members section, lists the users by its names, their positions, the according project, and the number of datasets they maintain. Further, the list is sortable and filterable by the items (name, position, project, datasets) shown for each users in the list. Additionally, the list can be filtered by project, as well as by a text search.

6.3.1. User roles

For administering the user base, an administration interface was developed. This interface allows to enable and disable accounts, and has interface to track for example pending access request.

There are currently five user roles implemented in the CRC806-Database:

1. Admin
2. Subadmin
3. Clusteradmin
4. Projectadmin
5. User

The Admin (1.) role is able to edit all datasets, and as the only role able to access the administration console, as described in section 9.2.4, and thus able to manage other users roles and access rights. Subadmin's (2.) are able to edit all datasets in the data catalog and publication DB. The Clusteradmin (3.) can edit all datasets of that users cluster, and Projectadmins (4.) can do the same for the given project. The normal Users (5.) can only edit the datasets they own (are maintainer of).

6.3.2. Linked Data and RDFa

The user profile pages are also annotated with RDFa attributes. An example RDFa annotated user profile is given in listing 6.10. For the user profiles only one vocabulary, Schema.org (Brickley et al. 2016), was implemented, because it covers all all properties used in a user profile in its Schema.org Person schema.

Listing 6.10: Example RDFa annotation on the user profile pages.

```
<div class="size66" vocab="http://schema.org" typeof="Person">
<div class="flatBox">
  <h2 id="profileTitle">
    
    <span property="name">Christian Willmes</span>
  </h2>
  
  <div class="datacontainer">
    <div class="datarow">
      <span>E-Mail:</span>
      <span property="email"><a href="mailto:c.willmes@uni-koeln.de">c.willmes@uni-koeln.de</a></
        span>
    </div>
    <div class="datarow">
      <span>Phone:</span>
      <span property="telephone">02214706234</span>
    </div>
    <div class="datarow">
      <span>Fax:</span>
      <span property="faxNumber">02214702280</span>
    </div>
    <div class="datarow">
      <span>Website:</span>
      <span><a href="http://crc806db.uni-koeln.de" property="url">http://crc806db.uni-koeln.de</a><
        /span>
    </div>
  </div>
</div>
```

```
</div>
```

6.4. News & Blog

The frontpage is based on an own Typo3 *Extbase & Fluid* extension, that handles the news and blog posts on the front page including the latest data, geodata / maps and publication streams and the display of the website visitor statistics.

The homepage contains as main element a news and blog post list, with the newest blog or news post on the top. A simple model for posts, that can be of type news or blog was implemented and persisted in the Typo3 MySQL instance using the default Extbase repository model. The post are edited in raw HTML markup and inserted into the MySQL database using phpMyAdmin, because until now, no editing interface for blogs and news posts is implemented. The streams for the latest data sets, publications and geo-data sets, the functionalities for accessing the according backends has been reused from the data catalog, maps and publications extensions, by injecting the according services. Thus functionalities from the maps, data and publication extensions are accessible for building the streams displaying the latest five data sets, publications and maps or geodatasets. A map showing the locations of the website visitors is included as a widget, provided by PIWIK (PIWIK Contributors 2014). For tracking user and visitor statistics a PIWIK system is setup, as described in section 5.1.

6.4.1. RSS and ATOM feeds

The blog, news, latest data, latest publications and latest geodata streams can also be subscribed to RSS or ATOM feeds, that are linked from the rendered streams.

Listing 6.11: RSS Feed shortened example.

```
<?xml version="1.0" encoding="UTF-8" ?>
<rss version="2.0">
<channel>
  <title>CRC806-Database Data Feed (RSS)</title>
  <link>http://crc806db.uni-koeln.de/</link>
  <description>Data feed of the CRC806-Database</description>
  <item>
    <title>Multi-emission luminescence [...]</title>
    <link>http://crc806db.uni-koeln.de/dataset/show/multiemission-luminescence-dating-1441370029/</link>
    <description>Sodmein Cave in Egypt is one of the rare archaeological sites in [...].</description>
    <pubDate>2015-09-04T12:33:31+02:00</pubDate>
    <author>A. Author</author>
  </item>
  <itme[...]</item>
</channel>
```

The RSS (W3C 2002) standard is not an official W3C specification, but hosted and maintained by the W3C. The aim of the format is to provide a standard for syndicating content from distributed content publishers on the web. This is most popular for blogs and podcast, that are

traditionally published in a distributed way. The format is XML based and thus it is relatively easy to implement using the XML libraries that are available in almost every web programming framework. Atom is an other format for content syndication. It is like RSS also XML based and has like RSS a direct mapping to RDF, which qualifies it also as a Semantic Web technology implementation.

6.5. Integrated Search

To provide a single interface, that allows the user to search all backends of the CRC806-Database from one central UI, the integrated search extension was implemented. The UI offers a simple text based search field and the tool triggers the same full text search entered into the search field on all backends of the CRC806-Database. This includes the CKAN based data catalog and publications DB, as well as the GeoNode backend of the SDI.

The results from the different backends are retrieved as JSON objects, and then merged and sorted by creation date and rendered as the list. This list can then be further filtered by project and tag/keyword. The records of the different backends are indicated by an according icon for its source. As introduced in section 5.2.2, CKAN has an API to access all its functionality (data access, search, upload, editing, etc.) via REST technology. GeoNode offers also an RPC-style API to access the GeoNode search and filter functionalities. Both interfaces provide their results in JSON, consisting some properties, that overlap and are thus suitable for integrated sorting and filtering:

- Title
- Creation date
- Topic

Per default, the results are sorted by creation date, but it is also possible to sort by title and topic. The integrated search is implemented as an own *Extbase & Fluid* extension, see listing 6.12 reusing the CKAN service of the data catalog extension, and the GeoNode service of the Maps SDI extension.

Listing 6.12: Folders and files of the Integrated Search extension.

```
.
|-- Classes/
| '-- Controller/
| |-- GlobalsearchController.php
| '-- QuicksearchController.php
|-- ext_emconf.php
|-- ext_icon.gif/
|-- ext_localconf.php
|-- ext_tables.php
'-- Resources/
    |-- Private/
    | '-- Templates/
    | |-- Globalsearch/
```

```

| | '-- Results.html
| '-- Quicksearch/
| '-- Show.html
'-- Public/
  |-- css/
  | '-- dropdowns-enhancement.css
  '-- js/
    |-- dropdowns-enhancement.js
    '-- globalSearch.js

```

6.6. Continuous integration and testing

Continuous integration – the practice of frequently integrating one’s new or changed code with the existing code repository – brings several advantage to the development process and the stability and security of the developed code base, see section 4.1.3. Based on this advantages, it was decided to deploy a CI-System pipeline for the CRC806-Database infrastructure, as shown in figure 6.9.

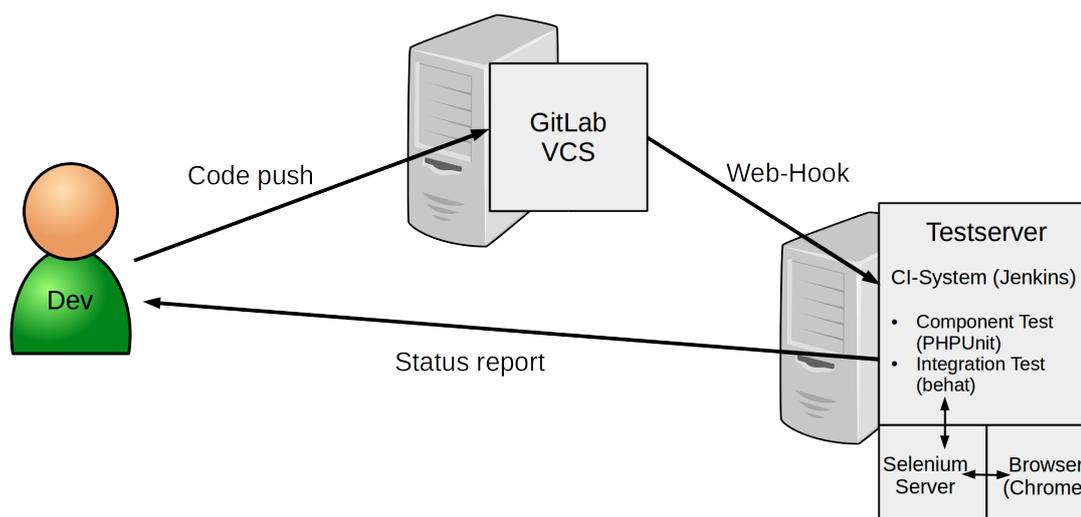


Figure 6.9.: The CRC806-Database CI-System pipeline. Source: Own work.

The CRC806-Database CI-System consists of a set of the following software components:

- GitLab (GitLab Community 2015),
- PHPUnit (Bergmann 2016),
- Behat (Kudryashov 2016),
- Selenium (Selenium Contributors 2016),
- Jenkins (Jenkins Contributors 2016).

GitLab (GitLab Community 2015) is the VCS used by the CRC806-Database, see section 5.2.5 for details of the setup.

PHPUnit (Bergmann 2016) is a unit testing framework for the PHP programming language. The framework uses assertions to verify that the behavior of the specific component – or "unit" – being tested behaves as expected.

Behat (Kudryashov 2016) is an open source Behaviour Driven Development (BDD) framework for the PHP programming language. BDD is a way to develop software through a constant communication with stakeholders in form of examples. Behat was heavily inspired by Ruby's Cucumber project (Wynne et al. 2012). Since v3.0, Behat is considered an official Cucumber implementation in PHP and is part of one big family of BDD tools. In the CRC806-Database CI-System Behat is used for conducting integration tests, as further described below.

Selenium is a suite of tools specifically for automating web browsers (Selenium Contributors 2016). In the CRC806-Database CI pipeline it is applied for testing UI components.

The main testing framework used in the CI-System is Jenkins (Jenkins Contributors 2016). In a nutshell, Jenkins is the leading open source automation server. Built with Java, it provides hundreds of plug-ins to support building, testing, deploying and automation for virtually any project (Jenkins Contributors 2016).

The CI pipeline of the CRC806-Database looks like the following:

1. Developer pushes his code commit (*Code push* in fig. 6.9), after finishing the development of a feature or fix and local manual testing, to the GitLab instance.
2. When GitLab receives a new commit, Jenkins is signaled that a commit event took place via *Web-Hook* (see fig. 6.9).
3. Jenkins will now pull the current code base from GitLab, for automatically testing the new code.
4. First, a component test provided by PHPUnit is conducted. Here only single components, like a function or a class, are tested for its function.
5. Then, an integration test using the Behat software, is conducted. Here the overall system is checked for errors. This test is also called black-box test, because only the interaction with the system is tested. The UI is tested using the selenium framework, that allows for testing scripted web browser user interactions, for example screen visibility problems, button clicks, and form submissions etc.
6. If both tests are successful, the current build will be marked as successful too, if a test fails, the developers are notified by email of the build fail (*Status report* in fig 6.9).

The browser based tests, through the Selenium software, are triggered by the Behat framework. Behat provides convenient so called scenarios for automated browser based testing.

7. Spatial Data Infrastructure

The CRC806-SDI consists of three main building blocks, a GeoNode based backend, a MapServer and MapProxy based additional OWS backend, and a Typo3 *Extbase & Fluid* extension to integrate the SDI into the CRC806-Database main web application. As the main facilitator, GeoNode (GeoNode Contributors 2014) is installed for delivering OGC compliant geospatial data services, also known as OWS (see section 2.6.2). Additionally, for services having higher performance or data amount demands, infrastructure based on MapServer (MapServer Contributors 2014) and MapProxy (Tonnhofer et al. 2014) are provided (see section 7.2). For the frontend of the SDI, that is the web based UI, the services are integrated into the CRC806-Database Typo3 based web portal via a custom developed *Extbase & Fluid* based extension (see section 7.3).

Because of the major update of the CRC806-Database system in 2014 (see chapter 6, as well as section 12.4.1 for a detailed overview and discussion of the 2014 update), the SDI part of the system was migrated from a MapServer, GeoServer, MapProxy, pycsw and GeoExt based implementation, as described in (Willmes et al. 2014e), to a GeoNode based system as described here.

The old setup, as described in detail in Willmes et al. (2014e), was much more complex, because it consisted of more software components to integrate into one SDI. Much of this complexity is now handled by the main application of the current setup, that is GeoNode. The high degree of complexity of the old system exposed the SDI to many vulnerabilities, because of its reduced complexity, the current system is much more resilient, because it has less points of possible failure. Additionally, GeoNode is a comparably large and well maintained Open Source project. It has several full-time developers dedicated to the project, funded by World Health Organization (WHO) and World Food Programme (WFP), and also contributors from smaller companies and also from the academic sector.

In figure 7.1 the architecture schema of the CRC806-Database SDI is given. The organization of the three building blocks and interfaces between GeoNode backend, MapServer and MapProxy backend and the Typo3 web based frontend are shown. The user is able to access the SDI via the web frontend or via OWS services, that are in case of GeoNode behind a reverse proxy, and thus have a different endpoint than the Mapserver and MapProxy OWS of the SDI.

The GeoNode server is running behind the UKLAN firewall. Thus, the GeoServer based OWS endpoints, of the GeoNode application, are proxy-ed through a reverse proxy implementation, that only takes valid OWS requests. This setup was chosen for security reasons, to not have the GeoNode server in "the open", vulnerable for compromising attacks on several infrastructure levels. The reverse proxy can also be facilitated to gather statistics about the use of the OWS

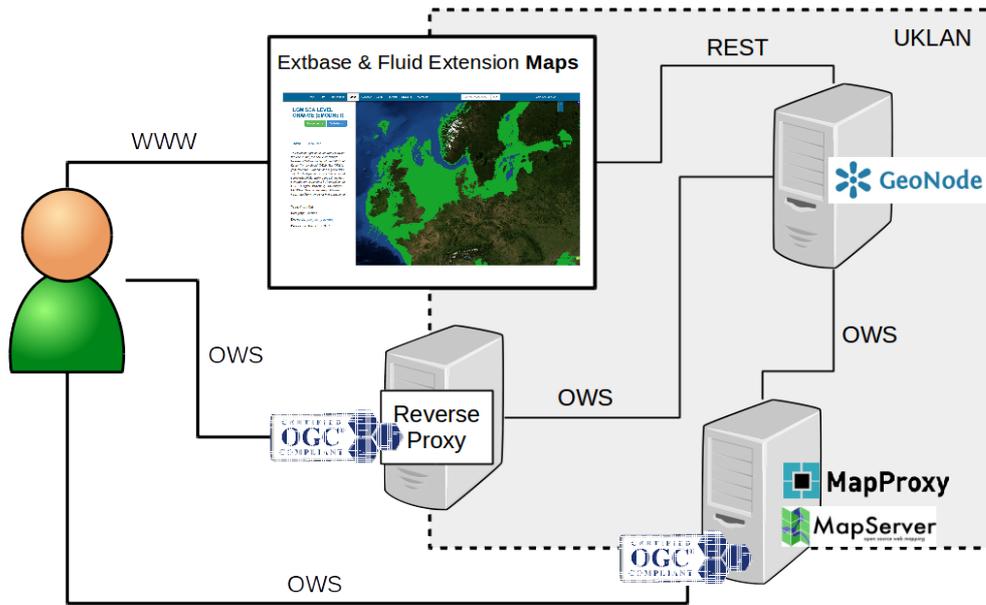


Figure 7.1.: Architecture of the CRC806-Database SDI. Source: Own work.

offered through the SDI. The CRC806-Database SDI also offers an OpenLayers (OpenLayers Contributors 2015) based WebGIS interface, the approach and its implementation is described in detail in section 7.3.

7.1. GeoNode

Similar to the CKAN interface (see section 6.1) for the data catalog implementation, a frontend to interface GeoNode from the CRC806-Database website, was developed. As introduced in section 5.2.3, GeoNode (GeoNode Contributors 2014) is an integrated SDI geo-content publishing system that integrates a complete SDI stack based on GeoServer (GeoServer Contributors 2014), pycsw (Kralidis et al. 2014), PostgreSQL (PostgreSQL Contributors 2015) with PostGIS extension (Ramsey et al. 2014), and GeoDjango (GeoDjango Contributors 2014) as web application framework. The application offers additionally to the OWS interfaces a REST based API to access the GeoNode application functionality remotely. The integration of GeoNode application functionality into the CRC806-Database website is described in section 7.3.

GeoNode was chosen as the main SDI facilitator of the CRC806-Database, because it offers web based functionality to easily create geospatial web services, including standard conform metadata annotation (see section 7.1.2), management of service metadata, OGC compliant CSW implementation, and integration of third party OWS into this CSW instance, for enabling a central CSW endpoint, see section 7.1.4. A main purpose of the GeoNode instance is its function for federating all SDI endpoints and services of the CRC806-SDI into one single and consistent repository.

The GeoNode application, or more precisely its underlying GeoServer instance builds upon

PostgreSQL as its persistence layer for the geodata. Additionally, the spatial metadata handled by GeoNode is persisted in PostgreSQL using its spatial extension PostGIS (Ramsey et al. 2014). The database stores application and configuration informations, including user, layers, maps, and the underlying geodata. To store the geodata in the PostgreSQL/PostGIS database, instead in multiple files in the GeoServer data directory, results in performance advantages, especially for queries based on the data and complex style rules based on attributes of vector data.

7.1.1. GeoNode server setup

As introduced in section 5.1, the GeoNode (GeoNode Contributors 2014) system is installed on Virtual Machine hosted at the RRZK. The basic server resources and the GeoNode software (see section 5.2.3) are already introduced.

The system is installed on an Ubuntu server 14.04 LTS OS (Ubuntu Community 2014). GeoNode was installed from the GeoNode Ubuntu package repository¹. Some custom changes to the locations of the data directories, and to the table store of the PostGIS (Ramsey et al. 2014) backend were done, to have the data growing folders and tables on an extra larger storage volume mounted into the system. Listing 7.1 shows the SQL commands to define the PostgreSQL table spaces on the extra hard disk.

Listing 7.1: Setup of the extra table space for the growing PostgreSQL tables.

```
CREATE TABLESPACE exthd LOCATION '/media/data/postgresql/data';
ALTER DATABASE geonode SET default_tablespace = exthd;

\connect geonode

ALTER TABLE documents_document SET TABLESPACE exthd;
ALTER TABLE layers_attribute SET TABLESPACE exthd;
ALTER TABLE layers_layer SET TABLESPACE exthd;
ALTER TABLE layers_layer_styles SET TABLESPACE exthd;
ALTER TABLE layers_layerfile SET TABLESPACE exthd;
ALTER TABLE layers_style SET TABLESPACE exthd;
ALTER TABLE layers_uploadsession SET TABLESPACE exthd;
ALTER TABLE maps_map SET TABLESPACE exthd;
ALTER TABLE maps_maplayer SET TABLESPACE exthd;
ALTER TABLE maps_mapsnapshot SET TABLESPACE exthd;
ALTER TABLE services_service SET TABLESPACE exthd;
ALTER TABLE services_servicelayer SET TABLESPACE exthd;
ALTER TABLE services_serviceprofilerole SET TABLESPACE exthd;
ALTER TABLE services_webserviceharvestlayersjob SET TABLESPACE exthd;
ALTER TABLE services_webserviceregistrationjob SET TABLESPACE exthd;
ALTER TABLE upload_upload SET TABLESPACE exthd;
ALTER TABLE upload_uploadfile SET TABLESPACE exthd;
```

The mentioned data partition is mounted under `/media/data/`, and the PostgreSQL data is stored under `/media/data/postgresql/data`, see listing 7.1.

¹<https://launchpad.net/~geonode/+archive/ubuntu/stable>, accessed: 2016-01-11.

7.1.2. Creating Data Services

GeoNode makes it really easy to publish geospatial web services. It offers two simple solutions for this task, the standard solution is using the GeoNode web applications UI, that offer simple web based forms to upload, data, define a visualization of the data and annotate the metadata according to ISO 19115 (ISO19115-1 2014) standard geospatial metadata (section 7.1.2). The second way is to use the desktop GIS QGIS, including the GeoNode plug-in, to create and upload geospatial web services for a GeoNode instance (see section 7.1.2).

GeoNode web application

GeoNode offers a feature rich web application, that allows to upload, edit, style, publish, organize and manage geospatial web services. The application leverages the Django web framework, offering an extensible environment for web developers accustomed to using any modern MVC web framework. GeoNode's frontend is based on jQuery and Bootstrap frameworks. A set of jQuery JavaScript plug-ins is used to implement the user interface, and Bootstrap CSS is used to style the pages (GeoNode Contributors 2014).

In figure 7.2 the web GUI based upload process of the GeoNode web application is shown. The basic upload process consists of two simple steps, a.) choosing of the geodata files on the local file system (fig. 7.2, #1 and #2), and b.) annotation of metadata (fig. 7.2, #3). In fig. 7.2 #4, the GeoNode backend view of a geodataset is shown. Additionally, it is possible to style the layer using the web GUI. For this feature, GeoNode provides a GeoExt (GeoExt Contributors 2014) web based Styled Layer Descriptor (SLD) editor. It is also possible to upload an SLD file, for example created with QGIS, as described in the following section, to visualize the geospatial layer.

QGIS GeoNode plugin

The OpenGeo Explorer (Boundless Inc. 2015) QGIS plug-in is an interface to manage and configure OpenGeo Suite components such a GeoServer and PostGIS. Figure 7.3, shows the UI of the QGIS plug-in.

Using the OpenGeo Explorer plug-in, it is possible to publish complete GIS Projects (data and its visualizations) with some few clicks to GeoNode. A convenient metadata editor is provided also, to annotate the spatial layers in Dublin Core, ISO 19115 and INSPIRE conform metadata. In theory almost all the geodata management could be executed through the QGIS plug-in, without using the web based fronted. But both interfaces have their advantages and disadvantages.

The most useful feature of the QGIS plug-in is the possibility to edit already published services through the plug-in. For example, adjusting the visualization or adding further features (data) to the service. This is also possible using the web interface, but the capabilities and usability appears to be more sophisticated in the QGIS plug-in.

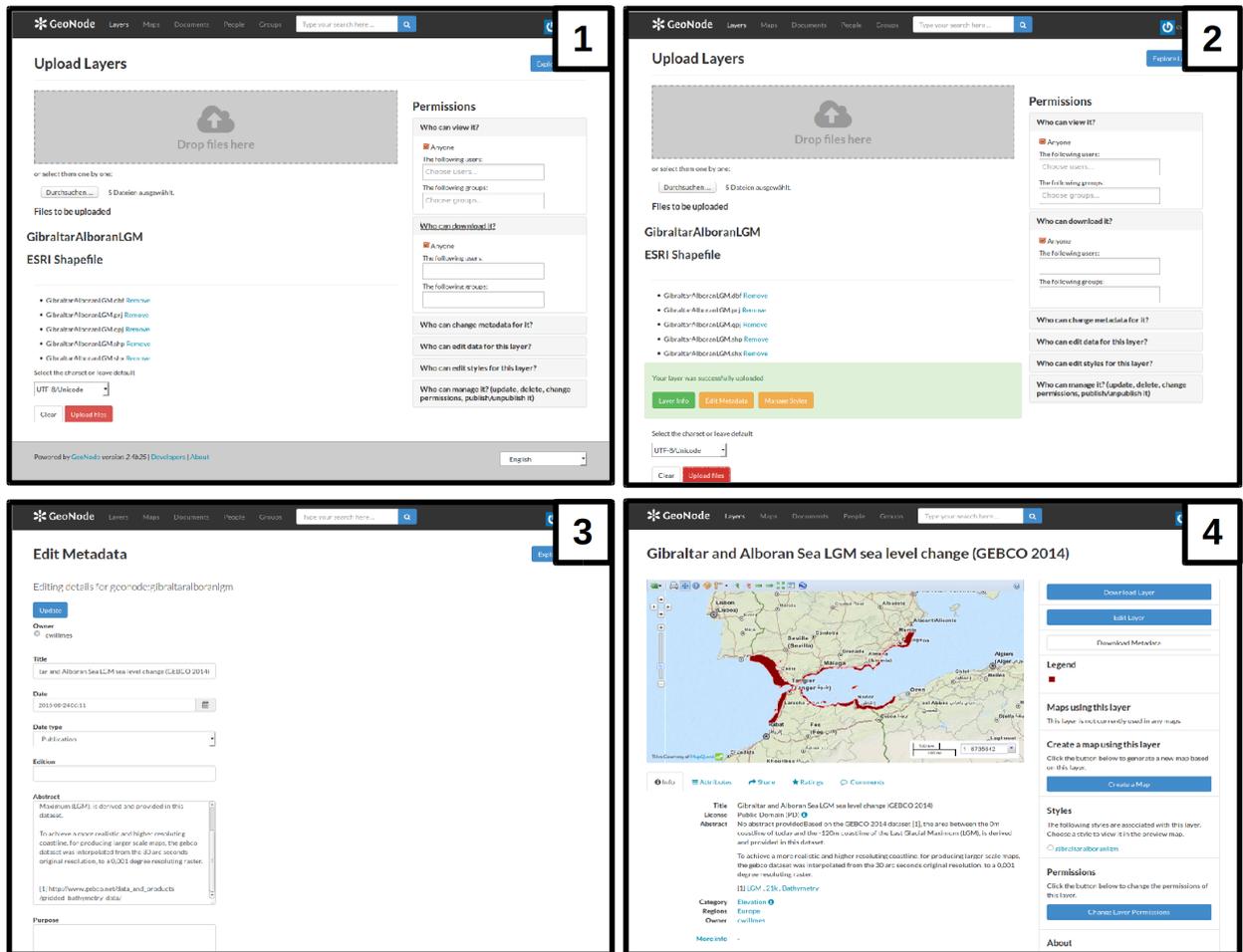


Figure 7.2.: Screenshots of the GeoNode Web GUI based upload process.

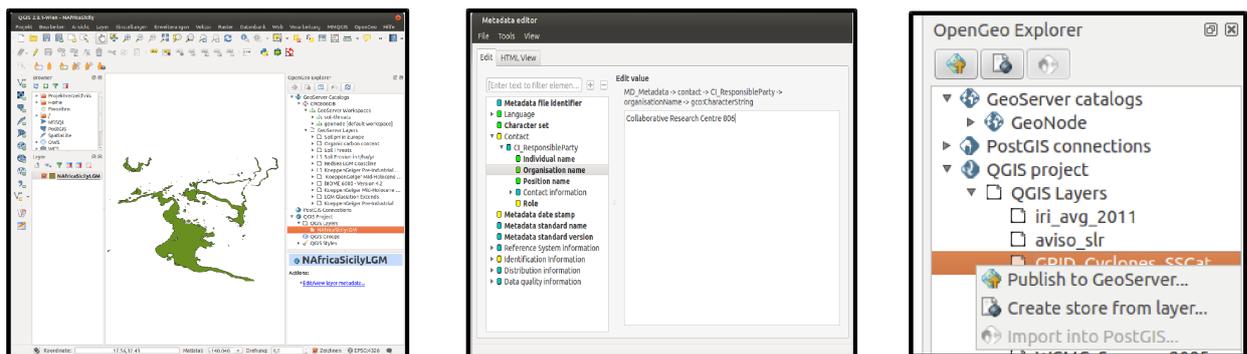


Figure 7.3.: GUI screenshots of the OpenGeo Explorer QGIS plug-in.

7.1.3. Integration of KB infrastructure data

To integrate data from the SMW based KB, a work flow as shown in figure 7.4 using the KML Semantic Result Format (SRF) was developed.

As described in detail in section 8.4.2, it is possible to export query results from the SMW based

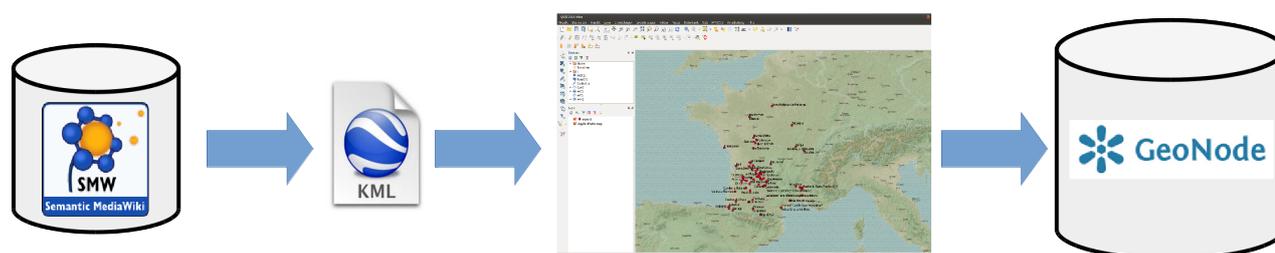


Figure 7.4.: Integration of spatial KB data into the SDI. Source: Own work.

KB in many different formats using the so called SRF extension. One of these formats is KML, that supports the markup of spatial features. An example query could be for archaeological sites, that have artefacts dated in the Aurignacien cultural period, can be exported as a KML dataset, containing informations about these sites, including its coordinates. QGIS as well as ArcGIS, support the KML format. From QGIS the data can be added to the GeoNode based SDI using the OpenGeo Explorer plug-in, as described in section 7.1.2.

7.1.4. CSW and integration of external OWS

It is possible to proxy and integrate OWS from third party OGC compliant infrastructures in GeoNode. The services can even be cached (WMS only), to speed up the display and protection against service failure of the external infrastructure. The integration is facilitated by a simple web UI, that allows to enter the service endpoint and provide additional metadata, that is not already provided by the service. The service metadata is retrieved automatically and relayed through the GeoNode system. The external services can be annotated and enhanced with additional metadata, such as keywords and categories, to fit the local custom GeoNode or SDI categorizations and annotations. The service metadata is then integrated and exposed via the OWS compliant GeoNode pycsw (Kralidis et al. 2014) CSW interface, see figure 7.5.

Since version 2.0 of GeoNode builds on pycsw (Kralidis et al. 2014), as its CSW component by default. In previous versions GeoNetwork (GeoNetwork Contributors 2015) was integrated, and can still be used as CSW by configuration of the GeoNode application. However, within the CRC806-Database SDI installation, pycsw is used as CSW. The GeoNode pycsw instance is facilitated as the central CSW of the CRC806-Database SDI. The services offered through MapServer and MapProxy are also listed in the GeoNode CSW, through the external service integration feature, as described above.

The CRC806-Database CKAN instance (see section 5.2.2) offers also an pycsw CSW interface for its spatial metadata. Until now the CKAN CSW is not exposed to the public and also not integrated with the CRC806-Database SDI but this is possible to implement in the future.

Figure 7.5.: GeoNode Web UI for registering external (remote) services. Source: Screenshot.

7.2. MapServer and MapProxy

WebGIS

The CRC806-SDI setup does not only consists of the GeoNode application. On the `sf806srv` server (see section 5.1), two additional OGC conform web services are deployed, as part of the SDI, using MapServer (MapServer Contributors 2014) and MapProxy (Tonnhofer et al. 2014) technology.

7.2.1. MapServer

The MapServer (MapServer Contributors 2014) instance of the CRC806-Database is installed on the `sf806srv` data server (see section 5.1) from the Enterprise Linux GIS (ELGIS) repositories².

Through MapServer only larger WCS services with at least several hundred MB base data volume are deployed. For example the SRTM DEM of the whole CRC 806 research area, that is Northern Africa, the Levant and Europe (see section 1.2), or the WorldClim and BioClim datasets. Because, MapServer is objectively much faster for delivering raster based OWS on large data bases, then the GeoServer based GeoNode infrastructure. Another large dataset are the coast-lines for lower sea levels, derived from bathymetrie datasets, that are about 1 GB in size. Advantage of MapServer is, at first it's the WCS implementation. It is considerably faster and also more robust compared to the GeoServer WCS implementation, that comes with GeoNode.

Listing 7.2: Mapserver configuration file for the CRC 806 SRTM WCS.

```
MAP
NAME "CRC806_SRTM_WCS"
STATUS ON
SIZE 400 300
EXTENT -6.860525 3.460765 41.576181 53.276239
UNITS dd #METERS
CONFIG "MS_ERRORFILE" "/var/local/geodaten/srtm/mapserver.log"

OUTPUTFORMAT
```

²http://wiki.osgeo.org/wiki/Enterprise_Linux_GIS, accessed: 2015-08-24.

```

NAME "GEOTIFFINT16"
DRIVER "GDAL/GTiff"
MIMETYPE "image/tiff"
IMAGEMODE "INT16"
EXTENSION ".tif"
END

WEB
IMAGEPATH "/var/local/geodaten/srtm"
IMAGEURL "/var/local/geodaten/"
METADATA
"wcs_label" "CRC806_SRTM" ### required
"wcs_description" "The SRTM is a digital height model from the Space Shuttle Mission. The service is made
accessible by the Institute for Geography, AG Gis and Remote Sensing, for the CRC 806"
"wcs_onlineresource" "http://sfb806srv.uni-koeln.de/srtm?" ### recommended
"wcs_fees" "none"
END
END

PROJECTION
"init=epsg:4326"
END

LAYER
NAME CRC806_SRTM
STATUS OFF
TILEINDEX "tindex.shp" #"/var/local/geodaten/srtm/tindex.shp"
TILEITEM "location"
TYPE RASTER
DUMP TRUE
METADATA
    wcs_label "CRC806_SRTM_WCS"
    ows_extent "-20 0 65 60"
    wcs_resolution "0.0008333333333333 -0.0008333333333333"
    wcs_size "102000 72000"
    #ows_srs "EPSG:4326"
    wcs_formats "GEOTIFFINT16"
    wcs_nativeformat "geotiff"
    wcs_rangeset_nullvalue "-32768"
END
PROJECTION
"init=epsg:4326"
END
END
END

```

The Service endpoint for the above listed example configuration is the following:

<http://sfb806srv.uni-koeln.de/srtm?>

7.2.2. MapProxy

The MapProxy instance is also deployed on the sfb806srv server. MapProxy is configured in a YAML settings file. Basically it allows to cache any WMS, and is able to deliver the cached

service in different projections and also in additional interface versions (e.g. WMS 1.0-1.3.0) and formats, like Tiled Map Service (TMS) and Web Map Tile Service (WMTS).

MapProxy is a Python (Python Software Foundation 2016) application, installed in virtual Python environment and deployed behind Apache `mod_wsgi`. The application is relatively resource saving, because it mostly just delivers already cached images via HTTP. Main advantage of MapProxy is its possibility to offer cached WMS of different source projections, and also offers re-projection on the fly. This provides advantages for delivering base maps for WebGIS applications using other base projections than the standard Web Mercator, best known from Google Maps.

The MapProxy instances that are used within the CRC806-Database SDI, are configured as so called *MultiMapProxy* setups (Tonnhofer et al. 2014). This allows to offer an own endpoint for each cached service. A MultiMapProxy deployment can run multiple instances in one process. In a MultiMapProxy deployment it is possible to add and remove services while the application is running. Listing 7.3 shows an example MapProxy service configuration. In this case the cache for the GEBCO Bathymetrie (General Bathymetric Chart of the Oceans 2014) WMS. Each configuration will be loaded on demand and MapProxy caches each loaded instance. The configuration will be reloaded if the file changes.

Listing 7.3: Example MapProxy configuration for the GEBCO Bathymetrie WMS.

```
services:
  demo:
    tms:
      use_grid_names: true
      # origin for /tiles service
      origin: 'nw'
    kml:
      use_grid_names: true
    wmts:
    wms:
      md:
        title: GEBCO 2014 WMS
        abstract: The General Bathymetric Chart of the Oceans (GEBCO) aims to provide the most authoritative
          publicly-available bathymetry data sets for the worlds oceans.
        online_resource: http://www.gebco.net

layers:
  - name: GEBCO
    title: General Bathymetric Chart of the Oceans
    sources: [GEBCO_cache]

caches:
  GEBCO_cache:
    grids: [webmercator]
    sources: [GEBCO_wms]

sources:
  GEBCO_wms:
    type: wms
    req:
      url: http://www.gebco.net/data_and_products/gebco_web_services/web_map_service/mapserv?
```

```

layers: GEBCO_LATEST
transparent: true
http:
  ssl_no_cert_checks: true

grids:
  webmercator:
    base: GLOBAL_WEBMERCATOR

globals:

```

The WMS endpoint of the example displayed in listing 7.3 is the following:

```
http://geonode.crc806db.uni-koeln.de:8081/GEBCO/service?REQUEST=GetCapabilities
```

Further MapProxy based endpoints are described in section ??.

7.3. Typo3 Extension

As indicated before, a Typo3 *Extbase & Fluid* extension was implemented to provide and integrate SDI functionality for the CRC806-Database web application under the name *Maps*. The structure of the extension is given in listing 7.4.

Listing 7.4: File tree structure of the Extbase & Fluid Maps extension.

```

.
|-- Classes/
| |-- Controller/
| | |-- EidDispatcher.php
| | '-- MapViewerController.php
| |-- Domain/
| | '-- Repository/
| | '-- LayerRepository.php
| |-- Service/
| | '-- GeonodeService.php
| '-- Utility/
|-- EidDispatcher.php
|-- ext_emconf.php
|-- ext_icon.gif/
|-- ext_localconf.php
|-- ext_tables.php
'-- Resources/
    |-- Private/
    | '-- Templates/
    | '-- MapViewer/
    | |-- AjaxSearch.html
    | |-- DetailSearch.html
    | |-- List.html
    | '-- Show.html
    '-- Public/
        |-- Icons/
        | '-- relation.gif/
        '-- js/
            |-- mapviewer.js
            '-- script.js

```

As described in section 5.2.3, the GeoNode functionality is integrated into the Typo3 frontend interfacing the REST API of GeoNode. The REST API interface is available since the version 2.1 of GeoNode, and exposes almost all functionality of the system for remote access, edit and update of its geospatial datasets and content. GeoNode is technically based on the GeoDjango Python framework (GeoDjango Contributors 2014). The GeoNode REST API is implemented and provided through the Tastypie library. Tastypie is a reusable application (it relies only on its own code and focuses on providing just a REST-style API) and is suitable for providing an API to any application without having to modify the sources of that application³.

All geodata and according ISO 19115 metadata is stored in the GeoNode (GeoServer and pycsw) PostGIS backend. The metadata of all OWS, that are not provided through GeoNode are also centrally stored and managed through the GeoNode pycsw instance, as described in section 7.1.4.

7.3.1. Browse and search interface

The extension rebuilds the functionality for the catalog layer list view and the WebGIS based detail view of geodatasets integrated in the Typo3 interface. Though, the extension only implements view, discover, browse and search functionality of the GeoNode application. Update, editing or deletion functionalities are not implemented, and only available through the web application on the local GeoNode server, that is only accessible from within the UKLAN.

7.3.2. WebGIS and Spatial datasets detail view

The detail view for geodatasets offers WebGIS features, that are implemented using the OpenLayers v2.14 framework (OpenLayers Contributors 2015). The WebGIS interface displays the geospatial dataset, as provided by GeoServer in Web-Mercator projection (EPSG:3785), and allows zooming and panning for detail visualization of the data.

7.3.3. Related resources

The detail view also offers the related data feature, already introduced in the data catalog extension description (section 6.1). This is implemented through an additional database table, storing the relations, instantiated through the Extbase model, and materialized in the MySQL backend of the Typo3 instance. The metadata of the related datasets are retrieved through the CKAN service provided by the data catalog extension.

³<https://django-tastypie.readthedocs.org/en/latest/tutorial.html>, accessed: 2015-08-23

8. Knowledge base

As introduced in section 3.3.3, the aim of the CRC806-KB system is to provide an information system for the CRC 806, that combines information additional to the scope of the CRC806-RDM data catalog and publications DB. In particular it aims to build a collaborative collection of available published internal and external data, as well as not yet published internal data. The CRC806-KB is developed on the basis of SMW technology, as described previously in section 5.2.7. Thus, a collaborative content editing environment is already implemented by the MW software framework. Due to the addition of the SMW extension, it is possible to handle structured and thus semantic data (information) in this system. A wiki based KMS as described here implements the demands by the project funders for facilitating VRE and in general research collaboration supporting systems, see section 3.1.2.

In the following sections, the technical details of creating a KB based on SMW are described. The basis of any KMS is its data model or ontology (Chalmeta et al. 2008), thus the implementation starts with the data model in section 8.1, that is iteratively developed from the schemata and properties of included data and information. The main data input method is by manually editing the wiki by researchers of the CRC 806, facilitated by several input forms as described in section 8.2. A special case of information resources in the academic sector are of course bibliographic informations. How they are handled in the KB system is shown in section 8.1.2. The automated data import, and integration process is described in section 8.3. Finally, in section 8.4, the development of queries and according user interfaces to work with the KB system are introduced.

8.1. Data model development

The data model as well as the development process evolved over time during the built up phase of the KB. This approach is not uncommon and even well defined in literature, thus the terminology is, that the data model is developed in a prototyping (see section 4.5) bottom-up data integration approach (Willmes et al. 2012a). For the practical and technical implementation of the data model in the SMW framework (see 5.2.7), the SDD tool Mobo (Heimler 2014) was applied.

A basic model, that implemented schemata Datasets, Resources, Bibliography resources, as well as schemata for topic, spatial and temporal annotations was developed from scratch, as the basic model of the KMS (see section 8.1.1). This basic model is continuously expanded and modified, during the import and addition of data and information, implementing the bottom-up prototyping approach (see section 4.3).

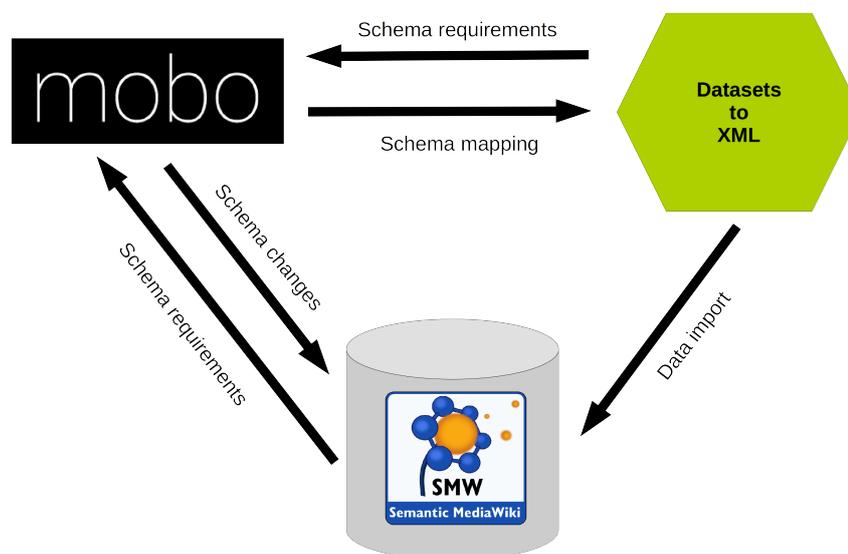


Figure 8.1.: Data model development feedback process during external dataset import. Source: Own work.

Figure 8.1 depicts the iterative data model modification and extension process of the CRC806-Knowledgebase schematically. The diagram shows the process that is applied in case of the import of a new external dataset, as described in more detail in section 8.3. First the dataset is mapped to the MediaWiki DataTransfer XML format (see sections 8.3.1 and 8.3.2). During this task it is possible, that the existing Mobo based schema needs to be adapted to cover the new schema. In this case, the schema is adapted through Mobo before the import of the DataTransfer XML. Additionally, the queries and filters (see section 8.4) that will be conducted on the KB are also influencing the data modelling process in a constant feedback loop. This is especially the case for the topic, spatial and temporal annotation models.

8.1.1. Mobo data modeling

As introduced in section 5.2.8, in Mobo models are formulated in JSON or YAML notation. Mobo uses the *JSON-Schema* Standard definitions (Galiegue et al. 2013) as the basis for the model development. In this context *JSON-Schema* provides the basic rules for defining object oriented paradigms and rules like class structures, objects and inheritance for example. To fit the model development better, some additions and simplifications to the *JSON-Schema* are introduced in Mobo. This adjusted *JSON-Schema* is referred to as *Mobo-Schema* (Heimler 2015b). The details of handling Mobo itself and configuring it will not be discussed here, only how the modelling process works, to give an overview of the data model implementation process.

In figure 8.2 the structure of the *Mobo-Schema* is shown. The basis of a Mobo developed model are the *Fields*, i.e. the SMW properties. The *Mobo Model* encapsulates 0..n *Fields* and sets them

in context, in a concept that seems similar to the class concept, known from Object Oriented Programming (OOP). The *Mobo Form*, allows to encapsulate 0..n *Mobo Models*, to enable OOP similar inheritance of Models to an encapsulating Form. This way, the model can be developed in an object oriented approach, allowing concepts like inheritance.

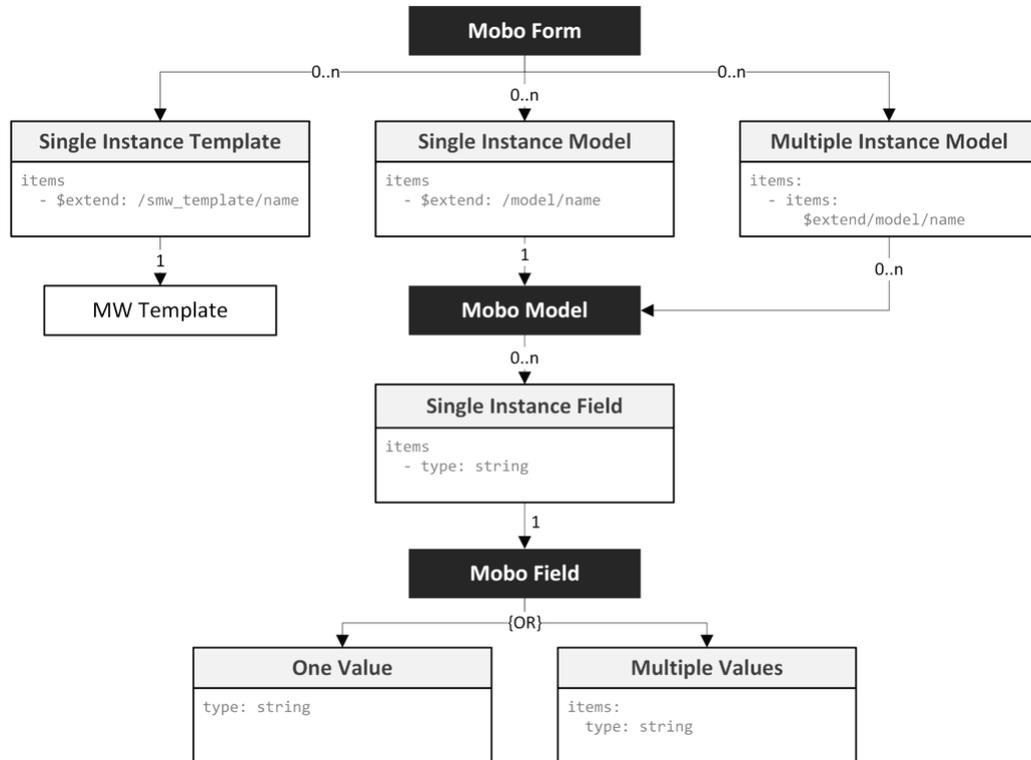


Figure 8.2.: Structure of the Mobo-Schema. Source: Heimler (2015a).

The Mobo tool runs on the local computer of the model developer and updates the data model in the wiki instance using a technology called MediaWiki bot. A Mediawiki bot is an automated tool or script, that performs certain, mostly often repeating tasks, such as updating data from external data sources, or checking for spam entries, by applying some logics. To enable this, a user with administration rights is created in the wiki instance and its credentials are configured for Mobo, to use that account for automatically updating the wiki, via the MW HTTP based API.

Another advantage of using Mobo for SMW based data modeling is, because the Mobo developed model is stored in text files in a folder structure, and can be easily managed in a VCS, for collaborative editing. The model can also be easily deployed to more than one SMW instance. To enable this, just the target instance has to be configured in the `settings.yaml` file. This has some further advantages for the applied model development approach, based on prototyping, because it allows to deploy several staging SMW instances.

In listing 8.1 the file and folder structure of the basic Mobo data model project for the CRC806-Knowledgebase is shown. A Mobo model consists, of a folder to contain field, form and model definitions organized in files and a strictly defined folder structure. The folders `smw_page`, `smw_query`, and `smw_template` are for SMW specific definitions. Additionally, a folder `mobo_template`

for Mobo specific definitions, that can contain layout definitions in form of templates.

Listing 8.1: CRC806-Knowledgebase Mobo project model folder and file structure.

```

.
|--field/
| |-- Bibliography.yaml
| |--Bliography/
| | |-- Abstract.yaml
| | |-- Address.yaml
| | |-- Author.yaml
| | |-- BibType.yaml
| | |-- Booktitle.yaml
| | |-- Chapter.yaml
| | |-- Citation.yaml
| | |-- DOI.yaml
| | |-- Edition.yaml
| | |-- Editor.yaml
| | |-- Howpublished.yaml
| | |-- Institution.yaml
| | |-- ISBN.yaml
| | |-- ISSN.yaml
| | |-- Journal.yaml
| | |-- Keyword.yaml
| | |-- Month.yaml
| | |-- Note.yaml
| | |-- Number.yaml
| | |-- Organization.yaml
| | |-- Pages.yaml
| | |-- Publisher.yaml
| | |-- School.yaml
| | |-- Series.yaml
| | |-- Title.yaml
| | |-- URL.yaml
| | |-- Volume.yaml
| | '-- Year.yaml
|-- DatasetURL.yaml
|-- Dataset.yaml
|-- Description.yaml
|-- Event.yaml
|-- Format.yaml
|-- ResourceLocation.yaml
|-- Resources.yaml
|-- ResourceType.yaml
|-- Source.yaml
|--Spatial
| | |-- Altitude.yaml
| | |-- BoundingBox.yaml
| | |-- Coordinates.yaml
| | '-- SpatialType.yaml
|-- Spatial.yaml
|--Temporal/
| | |-- Date.yaml
| | |-- EndDate.yaml
| | |-- StartDate.yaml
| | '-- TemporalType.yaml
|-- Temporal.yaml
|-- Topic.yaml
|-- Wikipedia.yaml
|--form/
| |-- Bibliography.yaml
| |-- Dataset.yaml
| |-- Resource.yaml
| |-- Spatial.yaml
| |-- Temporal.yaml
|-- Topic.yaml
|--mobo_template/
| |-- category.wikitext
| |-- form.wikitext
| |-- outline.wikitext
| |-- property.wikitext
| |-- query-ask.wikitext
| |-- query-sparql.wikitext
| |-- report.wikitext
|-- template.wikitext
|--model/
| |-- Bibliography.yaml
| |-- Dataset.yaml
| |-- Resource.yaml
| |-- Site.yaml
| |-- Spatial.yaml
| |-- Temporal.yaml
|-- Topic.yaml
|-- settings.yaml
'--smw_query/
| |-- bibliography-query.ask
| |-- resource-query.ask
| |-- resourcetype-query.ask
| |-- spatial-query.ask
| |-- temporal-query.ask
'-- topic-query.ask

```

The Mobo software compiles the data model defined in *Mobo-Schema* notation and organized in the file and folder structure, into an SMW data model, containing of SMW Page definitions, formulated in wiki text markup, for according Category, Template and Property definitions. These definitions are then automatically added to the wiki instance by a wiki bot and through the MW API.

In the following, the Dataset sub-schema is shown in detail as an example, of how the data

model is formulated in Mobo. A *Mobo-schema* consist of `/field/`, `/model/`, and `/form/` definitions.

/field/

The *Mobo Fields* are for the definition of SMW properties or attributes. They declare the property type and can contain settings for how to render the property in SMW. They further define a human readable title, an optional description, the data type according to the *JSON-Schema* specification, and whether a field links to a form. An example of a *Mobo Field* definition is given in listing 8.2.

Listing 8.2: An example Mobo Field definition: `/field/Dataset.yaml`.

```
title: Dataset
description: A Dataset.

type: array
items:
  type: Page
  form: Dataset
```

/model/

Mobo Models define SMW Templates and Categories, and the actual structure of the development model. Further, they define which models they inherit from, which fields they contain, the order of the fields, mandatory and recommended fields, the template output format, if they are modelled as properties or sub-objects in SMW, and can prepend and append optional wikitext. An example Mobo Model is given in listing 8.3.

Listing 8.3: An example Mobo Model definition: `/model/Dataset.yaml`.

```
title: Dataset
description: A dataset.

items:
  - $extend: /field/Source
  - $extend: /field/Abstract
  - $extend: /field/Temporal
  - $extend: /field/Spatial
  - $extend: /field/Topic
  - $extend: /field/DatasetURL
  - $extend: /field/Resources
  - $extend: /field/Bibliography
```

/form/

A Form describes all Mobo models a SMW form shall contain, which models to use, if the model should be implemented as single or multiple instance template, optionally which template to use,

and the order of models and templates in the generated SF. An example Form definition is given in listing 8.4.

Listing 8.4: An example Mobo Form definition: `/form/Dataset.yaml`.

```
title: Dataset

items:
  - $extend: /model/Dataset
```

8.1.2. Basic data model

The basic CRC806-Knowledgebase data model was developed from scratch to build a KB containing an overview of available datasets and bibliography of interest to the CRC 806. The model describes *Resources*, *Datasets* and *Bibliographic records* as basic items, that can be annotated with *Spatial*, *Temporal* and *Topic category* properties. In the following, the basic classes of the CRC806-Knowledgebase data model are explained.

Resource

The basis of the basic data model is the concept called *Resource*. A resource can be a file, or an URL pointing to a file or dataset published in an other location. It can be annotated with *Spatial*, *Temporal* and *Topic* properties, as well as *Bibliographic* references.

Property	Type	Description
ResourceLocation	url	A ResourceLocation can be a unique name, such as an URL.
ResourceType	Page	e.g., GIS data, spreadsheet, etc.
Format	string	e.g., shapefile or zipfile
Temporal	array Page	Annotations of type Temporal
Spatial	array Page	Annotations of type Spatial
Topic	array Page	Annotations of type Topic
Bibliography	array Page	Bibliographic items

Dataset

A dataset describes a collection of resources, the schema is primarily designed to model CRC806-Database data catalog datasets. But the model is also suitable to handle external and Internal, not yet published datasets. A property "Source" was implemented to indicate, if a given dataset is published in the *CRC806-Database*, or through an *External* platform or not yet published and handled as an *Internal* resource.

Property	Type	Description
Source	enum String	Internal, External, or CRC806-Database.
Abstract	String	Short description.
Temporal	array Page	Annotations of type Temporal.
Spatial	array Page	Annotations of type Spatial.
Topic	array Page	Annotations of type Topic.
DatasetURL	url	Link to Dataset
Resources	array Page	Resources of this Dataset.
Bibliography	array Page	Bibliographic items.

Bibliography

The Bibliography schema is implemented according to the previously elaborated BibTeX schema (Patashnik 1988), as given in table 6.6. It holds the bibliographic informations about published resources, that are referenced by resources and datasets.

Bibliography records, can also be annotated with spatials, temporals and topics, which makes it possible to browse and query literature according to these annotations. Also template queries, that automatically show all datasets and resources, that reference the given bibliographic item, are implemented in the Mobo model.

Spatial

The *Spatial* schema is implemented to annotate resources, datasets and bibliographic records with spatial properties. A spatial can be a BoundingBox, given in <West, South, East, North> WGS84 geographic coordinate notation, or a Coordinate given in WGS84 latitude and longitude.

Property	Type	Description
Geometry	enum String	Coordinates or BoundingBox
BoundingBox	string	<West,South,East,North> in WGS84 ordinates.
Coordinates	Geographic coordinate	The location of the Site in WGS 84 geographic coordinate notation.
SpatialType	enum String	Site, or Country, or Region.
Description	String	A short description of the spatial feature.
Wikipedia	url	Link to according Wikipedia site.

Temporal

The *Temporal* schema is implemented to annotate resources, datasets or bibliographic records with temporal properties. A *Temporal* can be either an Event or an Interval, depending on this it can have one Date or a StartDate and an EndDate. Dates are given in year before Christ (BC), because this is a well defined point in time. Before Present (BP), is not a well defined point in time, because the present is changing in time. In some applications BP is described as 1950 AD,

or in others 2000 AD, both referencing 0 AD as the offset, thus BC can be considered directly and avoiding to have uncertainties about what the present offset is.

Property	Type	Description
TemporalType	enum String	Interval or Event
Date	number	Years in BC. The date of an event.
StartDate	number	Years in BC. The onset of an Interval.
EndDate	number	Years in BC. The End of an Interval.
Topic	array Page	Annotations of type Topic.
Description	String	A short description of the Temporal feature.
Wikipedia	url	Link to according Wikipedia site.

8.2. Form based data entry

The CRC806-Knowledge base supports a set of methods for entering data. The probably most used method is the manual data entry by a user of the KB. Figure 8.3 shows the SF for creating or editing Dataset records of the KB.

The screenshot shows a web-based form for editing a dataset record. The title is "Edit Dataset: LGM paleoenvironment of Europe - Map". The form includes several input fields:

- Source:** A dropdown menu set to "CRC806-Database".
- Abstract:** A text box containing "The here documented GIS map and date".
- Temporal:** A text box with "LGM" and a small 'x' icon.
- Spatial:** A text box with "Europe" and a small 'x' icon.
- Topic:** A text box with "Paleoenvironment" and a small 'x' icon.
- DatasetURL:** A text box containing "http://crc806db.uni-koeln.de/dataset/show".
- Resources:** A list of three items: "LGM paleoenvironment Europe Map CRC806-Database.pdf", "LGM Europe Map v1.pdf", and "LGM Europe Map v1.png", each with a small 'x' icon.
- Bibliography:** A text box with "Becker2015" and a small 'x' icon.
- Freetext:** A large text area with a "Freetext [edit]" label. It contains a detailed paragraph describing the dataset: "The here documented GIS map and dataset contains a collection of Last Glacial Maximum (LGM, ~21k yBP) paleoenvironmental data. It is the first result of a project that aims to acquire, produce and publish GIS datasets from non-GIS based sources such as analogous maps, textual informations or figures of scientific publications for prehistoric time slices. In combination with modelling results and already available GIS-datasets related to the mentioned time frame, it should enable other researchers and members from other projects to use the maps in their work, properly cited and referenced. The map shows LGM land ice sheets, paleo-stream networks and inland water in Europe, a sea-level adapted (-120m) land mass and a Köppen-Geiger climate classification derived from climate model data."

At the bottom of the form, there are buttons for "Save page", "Show preview", "Show changes", and a checked checkbox for "Watch this page". The page footer includes "Privacy policy", "About CRC 806 Knowledgebase", "Disclaimers", and logos for "Powered by MediaWiki" and "Powered by Semantic MediaWiki".

Figure 8.3.: Form for editing/creating a dataset record in the CRC806-Knowledgebase. Source: Screenshot.

The screenshot in figure 8.3 shows further, that some input fields can hold more than one values. In the screenshot these fields are Temporal, Spatial, Topic, Resource and Bibliography. As shown in the schema description of the basic data model above, these properties are of type array Page.

In figure 8.4 a screenshot of the dataset record page, that was shown in the edit view in the screenshot above (fig 8.3), is shown.

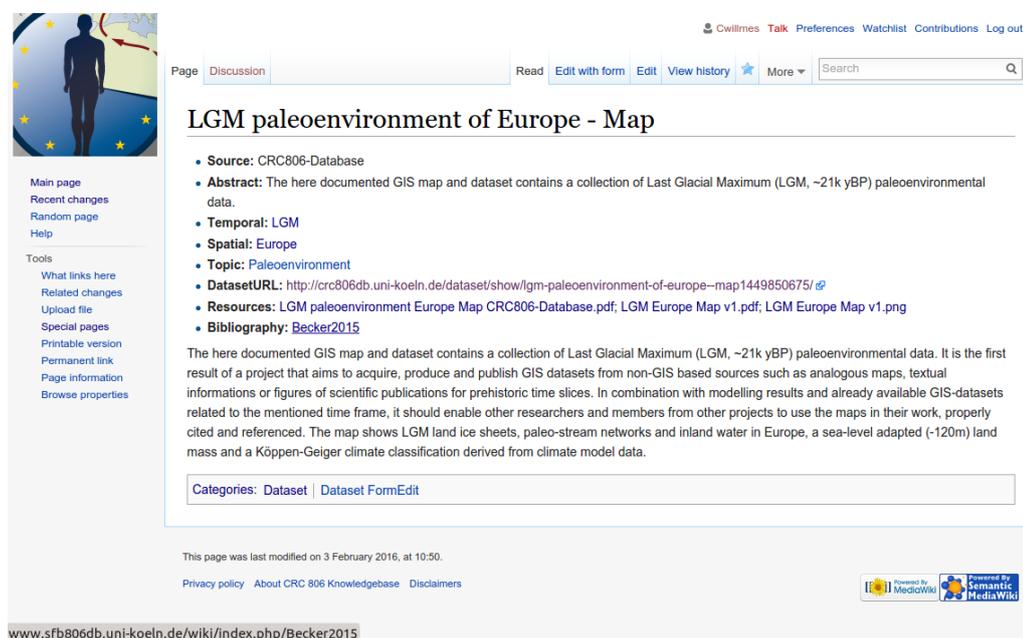


Figure 8.4.: A dataset record page of the CRC806-Knowledgebase. Source: Screenshot.

Each of the properties of type Page are automatically linked to another wiki page holding the according information. Clicking for example on the link labeled "Becker2015" would lead to the Bibliography record page as shown in figure 8.5.

This screenshot show two more important features of the presented KMS. First, it show some red links, that indicate, that the record linked to does not yet exist in the wiki. By following such a red link, the user will be led to a data input form for entering the according data and creating that record. This way it is possible to link to not yet existing information of the KB. The second interesting feature shown in the screenshot (fig. 8.5), are the "Referencing Datasets" and "Referencing Resources" lists (or tables), that are automatically created from defined queries, as later described in section 8.4.

8.3. Data import

SMW offers some different interfaces for automated bulk import of data. One of these interfaces is the use of the MW API to automatically enter data. This method is applied by Mobo, as explained above. Another interface is provided through anMW extension, DataTransfer (Koren 2015). By using the DataTransfer extensions capabilities, whole datasets, spreadsheets or even relational databases can be imported into a MW instance. The extension offers two main way for importing structured data. The first way is the possibility of importing Spreadsheets, that are filled with data rows. This method is described in section 8.3.1. The second method, is to import XML, that is formulated according to a custom DataTransfer extension schema. This method is

Becker2015

- **BibType:** Dataset
- **Title:** LGM paleoenvironment of Europe - Map
- **Author:** Daniel Becker; Jan Verheul; Mirjam Zickel; Christian Willmes
- **Year:** 2015
- **Publisher:** CRC806-Database
- **DOI:** 10.5890/SFB806.15
- **URL:** <http://crc806db.uni-koeln.de/dataset/show/lgm-paleoenvironment-of-europe--map1449850675/>
- **Topic:** Map; Paleoenvironment
- **Temporal:** LGM
- **Spatial:** Europe

Referencing Datasets:

Source	Abstract	Temporal	Spatial	Topic	DatasetURL	Resources	Bibliography
LGM paleoenvironment of Europe - Map	The here documented GIS map and dataset contains a collection of Last Glacial Maximum (LGM, ~21k yBP) paleoenvironmental data.	LGM	Europe	Paleoenvironment	http://crc806db.uni-koeln.de/dataset/show/lgm-paleoenvironment-of-europe--map1449850675/	LGM paleoenvironment Europe Map CRC806-Database.pdf LGM Europe Map v1.pdf LGM Europe Map v1.png	Becker2015

Referencing Resources: [edit]

ResourceLocation	ResourceType	Format	Abstract	Topic	Bibliography
http://crc806db.uni-koeln.de/dataset/getResource?tx_unikoelncrcdata_crcdata[id]=c0c819b9-71d9-4e69-89a5-db7ae31e8c12&tx_unikoelncrcdata_crcdata[type]=806&cHash=7804c6db1d62b9ab45689846f529dce6j	CRC806-Database	PDF		Glaciation Climate Inland Water Hydrology	Becker2015
http://crc806db.uni-koeln.de/dataset/getResource?tx_unikoelncrcdata_crcdata[id]=91dc09a1-44bc-4a64-8499-fbddca958ead&tx_unikoelncrcdata_crcdata[type]=806&cHash=8538fad4e04e26d135db1c96041754f3j	CRC806-Database	PNG		Glaciation Climate Hydrology Inland Water	Becker2015
http://crc806db.uni-koeln.de/dataset/getResource?tx_unikoelncrcdata_crcdata[id]=c19b221e-691d-491a-9c1b-d761dad83e51&tx_unikoelncrcdata_crcdata[type]=806&cHash=d91c0d7f5e433d2d12758ec7a1c16ee2j	CRC806-Database	PDF		Glaciation	Becker2015

Figure 8.5.: A Bibliography record page of the CRC806-Knowledgebase. Source: Screenshot.

described in section 8.3.2.

8.3.1. CSV and Spreadsheet import

CSV and Spreadsheet import are almost similar, because here the data must be formulated as table rows according to a schema defined in the first row of the table.

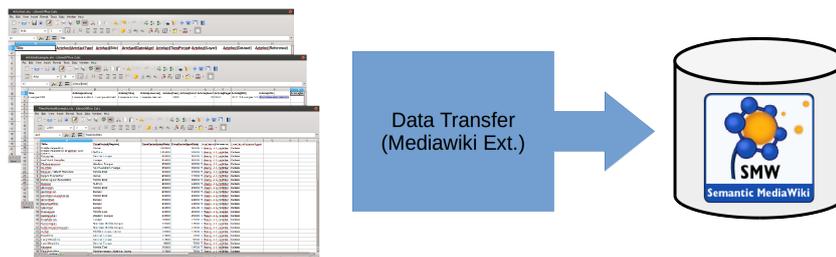


Figure 8.6.: Spreadsheet import via Data Transfer extension. Source: Own work.

The rows need to be structured according to a pre-defined way. The very first column has to have the page Title, all further columns are defined as:

template-name[field-name]

An example CSV file would look like given in listing 8.5.

Listing 8.5: An example DataTransfer CSV import file.

```
Title,Dataset[Source],Dataset[Abstract],Dataset[Temporal],Dataset[Spatial],Dataset[Topic],Dataset[DatasetURL],
Dataset[Resource],Dataset[Bibliography],Free Text
An example Dataset,Internal,"This is an example dataset.",LGM,Europe,Culture,http://example.com/testdataset,
Testdata.xls,Testauthor2016
```

8.3.2. XML import

The XML data import process as schematically visualized in figure 8.7, shows the steps needed to import datasets into SMW. The datasets to be imported need in most cases some editing and refinement. An Open Source Tool called Open Refine (Open Refine Contributors 2016) is used in some cases to refine messy datasets. This is followed by the import itself, for this task the datasets are transformed into an XML-Schema, defined through the DataTransfer extension. The transformation is implemented in Import Scripts mostly written in Python, as described in detail in section 8.3.3.

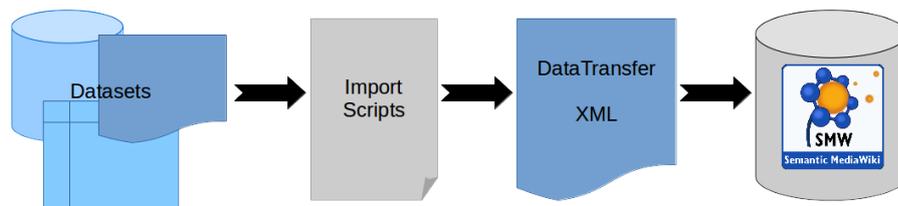


Figure 8.7.: DataImport process tool-chain. Source: Own work.

The DataTransfer extension defines a custom XML Schema to export and import MW content. It is possible to export/import single pages, or whole categories and even a whole wiki. An approach to parse and reformulate the data into MW XML markup, supported for import into the SMW instance using the Data Transfer extension (Koren 2015), was developed and implemented.

The fields and values of the DataTransfer XML format are taken from the fields and values in any template calls contained in the wiki pages selected for export, any non-template text is put into one or more "free text" tags.

In listing 8.6 an example XML file containing the contents of a Site object (MW page).

Listing 8.6: Example DataTransfer XML.

```
<Pages>
<Category Name="Site">
  <Page ID="155" Title="Abauntz">
    <Template Name="Site">
      <Field Name="SiteType">Cave</Field>
      <Field Name="Altitude">351</Field>
      <Field Name="Town">Arraiz</Field>
      <Field Name="Region">Navarra</Field>
      <Field Name="Country">Spain</Field>
      <Field Name="Coordinates">43.02734, -2.04167</Field>
```

```

    </Template>
  </Page>
</Category>
</Pages>

```

8.3.3. Generating import XML from datasets

For the creation of import XML files based on different types of datasets, Python scripts are used.

The scripts all have a common structure, first a reader according to the dataset format, using existing libraries for accessing spreadsheet (Excel), relational database or for data formats like JSON, XML and CSV. Then it is iterated through the dataset, and the data entities are written to an XML tree object. The standard python XML library is used to handle the XML tree object. See listing 8.7 for an example script header importing the `xlrd` and `xml` python libraries. The `xlrd` lib is used to access spreadsheet contents in this example, the `xml` lib is used to build the XML document.

Listing 8.7: An example script header, of a python import script.

```

#!/usr/bin/python

import xlrd
import xml.etree.cElementTree as ET

```

The main part of the script is the iteration through the contents and writing them into an XML tree object. This looks always something like listing 8.8. First an iterator is defined to loop through the dataset, then the contents of each data entity are mapped to the XML tree structure.

Listing 8.8: Example main part of an Python import script.

```

iterator = range(1, sheet.nrows-1).__iter__()
for i in iterator:
    #get the current row of the Excel sheet
    row = sheet.row_values(i)

[...]

#####
#Site
Site = ET.SubElement(root, "Page")
Site.set("Title", sname)

SiteTemplate = ET.SubElement(Site, "Template")
SiteTemplate.set("Name", "Site")

Town = ET.SubElement(SiteTemplate, "Field")
Town.set("Name", "Town")
Town.text = unicode(row[4])

Region = ET.SubElement(SiteTemplate, "Field")
Region.set("Name", "Region")
Region.text = unicode(row[5])

```

```

Country = ET.SubElement(SiteTemplate, "Field")
Country.set("Name", "Country")
Country.text = unicode(row[6])

Coordinates = ET.SubElement(SiteTemplate, "Field")
Coordinates.set("Name", "Coordinates")
Coordinates.text = unicode(row[8]) + ", " + unicode(row[7])

[...]

```

Finally, as shown in listing 8.9 the XML tree is then written into an XML document, that can be imported into the SMW instance.

Listing 8.9: Serialization of the XML tree into an XML document.

```

#write output
tree = ET.ElementTree(root)
tree.write("Dataset.xml")

```

8.4. Data queries and display

SMW has different features, interfaces and capabilities for data query, export and visualization. The basic concepts of this ASK language, and how queries are formulated were introduced in section 5.2.7. The data model development process is to a significant part influenced by the queries that should be run on the data. In the following, the data query and visualization interfaces are explained.

8.4.1. Semantic Search

SMW offers a complex query, browse and search interface, called *Semantic Search*.

In this UI, it is possible to build queries with the help of a form based user interface. The results of the query can be displayed in different result format and visualizations. These different result formats are facilitated through the Semantic Result Formats (SRF) extension (Koren et al. 2015b). Currently, the SRF extension contains 42 different formats. See figure 8.8 for a screenshot of the form based UI, the dropdown menu in the screenshot shows some of the available result formats.

8.4.2. Template inline queries

Mobo allows to define to some extent, how the templates for a model are rendered. On particularly useful feature is, that it is possible to include ASK queries that will appear on every instance of the given model. In listing 8.10, the Mobo notation of the Temporal model is given.

Listing 8.10: Temporal Mobo model, '/model/Temporal.yaml'.

```

title: Temporal
description: A temporal annotation, like an event or an intervall

```

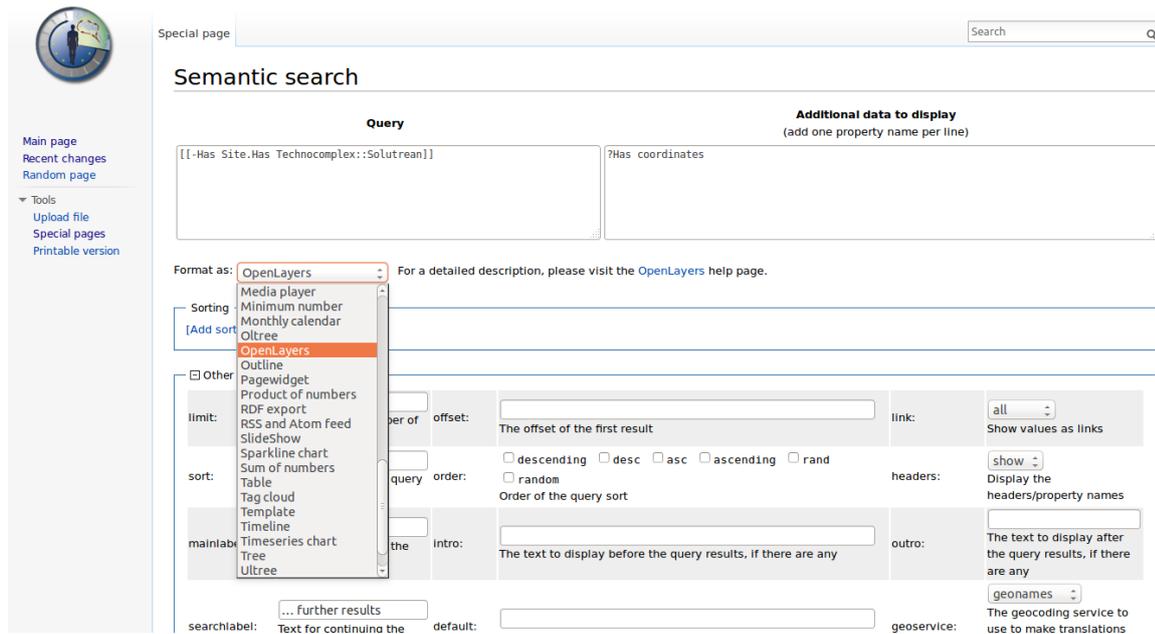


Figure 8.8.: Semantic Search user interface. Source: Screenshot.

```

items:
  - $extend: /field/TemporalType
  - $extend: /field/Topic
  - $extend: /field/Date
  - $extend: /field/StartDate
  - $extend: /field/EndDate
  - $extend: /field/Wikipedia

smw_appendPage: '{{Temporal-query-ask}}'
    
```

It includes an ASK query that is given in listing 8.11. This query actually contains two separate queries. In the first query, a search for all datasets, that are annotated with the instantiated Temporal is triggered and displayed as a broadtable. The second query does the same for Resources annotated with the given Temporal.

Listing 8.11: Temporal model ASK queries, '/smw_query/temporal-query.ask'.

```

=== Datasets annotated with this Temporal: ===

{#{ask:
[[Category:Dataset]] [[Temporal::{{FULLPAGENAME}}]]
|?Source
|?Abstract
|?Temporal
|?Spatial
|?Topic
|?DatasetURL
|?Resources
|?Bibliography
|format=broadtable
}}
    
```

```

=== Resources annotated with this Temporal: ===

{{#ask:
[[Category:Resource]] [[Temporal:{{FULLPAGENAME}}]]
|?ResourceLocation
|?ResourceType
|?Format
|?Abstract
|?Topic
|?Spatial
|?Temporal
|?Bibliography
|format=broadtable
}}

```

This results in displaying two tables on the according Temporal pages, one with datasets and one with resources annotated with the given Temporal. Techniques like this allow to develop standard queries, that are displayed for *Pages* of certain *Categories* in the KB. This feature enhances the user experience notably, it allows to browse through the KB application along individual paths of interest and curiosity.

8.4.3. Custom inline queries

Through the SRF extension (Koren et al. 2015b), the queries are very powerful tools, because formats for visualisation as well as for export of query results are available.

Basically all data can be queried and visualized using the SMW semantic search inline queries. An example inline query, would look like listing 8.12.

Listing 8.12: Example query for artefacts in a given timePeriod.

```

{{#ask:
[[Category:DatedAge]]
[[Age::>{{#show: {{PAGENAME}} | ?endDate}}]]
[[Age::<{{#show: {{PAGENAME}} | ?startDate}}]]
|?Age
|?Uncertainty
|?LabCode
|?SampleMaterial
|?Method
|format=broadtable
|link=all
|headers=show
|searchlabel=... further results
|class=sortable wikitables smwtable
}}

```

The display format is defined by the `|format=<srf>` command. In the example (shown in listing 8.12), the default format `|format=broadtable` is defined. As the name suggest, the results are displayed in a table, that can contain many columns, the number of columns is theoretically not limited, technically it is of course, but the number is very high. Popular SRF are for example; maps that allows to display interactive webmaps, if the queried items contains a property of type

Geographic Coordinate, or niche Pie or Bar charts, provided by the D3 JavaScript library can be displayed, if the queried items contain according properties of type Number.

8.4.4. Data visualization

The same temperiod query, where the result is visualized on map, would look like listing 8.13. This is possible, because the property Coordinates is of type Geographic Coordinate, and thus can be displayed by the Maps SRF.

Listing 8.13: An example inline query yielding a map as output format, showing artefacts dated in a given TimePeriod.

```

{{#ask:
[[[-Site::<q>[[ -ArtefactID::<q>[[Category:DatedAge]]
[[Age::>{{#show: {{PAGENAME}} | ?endDate}}]]
[[Age::<{{#show: {{PAGENAME}} | ?startDate}}]]</q>]]</q>]]
|?Coordinates
|format=map
}}
    
```

In figure 8.9 the results of the two above described inline queries are shown, the table goes much longer in the original output, but is cut her to save space.



	Age	Uncertainty	LabCode	SampleMaterial	Method
A Valina-GrN-20833-INQUA6747	31,730	2,800	GrN-20833	bone from level V?	14C
	31,600	250	GrA-3014	charcoal	AMS
A du Blot (Cerzat)-Ly-564-Stage3 Arch-551	21,700	1,200	Ly-564	carbonaceous earth	C14
	24,841	1,112			
A du Blot (Cerzat)-Ly-565-Stage3 Arch-552	21,500	700	Ly-565	carbonaceous earth	C14
	24,699	697			
A du Mas Viel (St-Simon)-Gif-3281-Stage3 Arch-542	26,770	800	Gif-3281	bone	C14
	30,435	1,004			

Figure 8.9.: Outputs of the two queries, map result and broadtable result. Source: Screenshot.

8.5. Data export

An ASK API request, containing the query as given in listing 8.12, would look URL encoded like listing 8.14.

Listing 8.14: An example ASK API request.

```
api.php?action=ask&query=[[Category::DatedAge]]|%3FAge|%3FUncertainty|%3FLabCode|%3FSampleMaterial|%3FMethod&format=jsonfm
```

Requesting this URL will trigger the according query in the SMW database, and an result will be returned. Many result formats, like CSV, XML, HTML, or JSON are available. Of course, result formats like maps or charts would not make sense, thus the formats for ASK API queries are limited to the above named ASCII based serialization formats. In this example the result will be returned in JSON format as shown in listing 8.15.

Listing 8.15: JSON result.

```
{
  "printrequests": [
    {
      "label": "DatedAge",
      "typeid": "_wpg"
    },
    {
      "label": "Age",
      "typeid": "_num"
    },
    {
      "label": "Uncertainty",
      "typeid": "_num"
    },
    {
      "label": "LabCode",
      "typeid": "_txt"
    },
    {
      "label": "SampleMaterial",
      "typeid": "_txt"
    },
    {
      "label": "Method",
      "typeid": "_txt"
    }
  ],
  "results": {
    "'t Ronde-GrN-4869-INQUA6157': {
      "printouts": {
        "Age": [
          7790
        ],
        "Uncertainty": [
          950
        ],
        "LabCode": [
          "GrN-4869"
        ],
        "SampleMaterial": [
          "charcoal"
        ],
        "Method": [
          "14C"
        ]
      },
      "fulltext": "'t Ronde-GrN-4869-INQUA6157",
      "fullurl": "http://www.sfb806db.uni-koeln.de/context/index.php/%27t_Ronde-GrN-4869-INQUA6157",
      "namespace": 0,
      "exists": true
    }
  }
}
```

[...] }.

Part IV.

Results

9. CRC806-RDM

The results of the development of the RDM Infrastructure, the CRC806-Database main web application are presented in this section. The web application is mainly a web browser based application and accessible from the following URL:

<http://crc806db.uni-koeln.de>

As described in detail in the previous chapters of this work, the web application is based on the Typo3 CMS and provides interfaces to a research data repository, a publication data base, and the CRC806-Database SDI, which is described in detail in chapter 10. These interfaces are all integrated into one consistent web page, and available from the main navigation bar, as shown in figure 9.1.



Figure 9.1.: Navigation bar of the CRC806-Database web application. Source: Screenshot.

The main navigation of the CRC806-Database web application contains the links to the main sections as listed and described in table 9.1. As it is also visible from figure 9.1, the web application provides additionally an search interface, that allows to search all three repositories (data, publications, and maps) of the CRC806-Database, integrated into the main navigation bar, as described in section 9.5.

Table 9.1.: The sections of the CRC806-Database web application.

Section	Description	Details
Start	The frontpage, including news streams, blog posts and visitor statistics.	Section 9.1
Data	The data catalog web interface.	Section 9.2
Publications	The publications DB web interface	Section 9.3
Maps	The public SDI web interface	Chapter 10
Services	An overview of the services, the Z2 project offers.	–
About	An overview of the CRC806-Database infrastructure and the members of the Z2 project	–
Imprint	The legal imprint of the web application.	–
Members	The members directory web interface.	Section 9.4

The Services, About and Imprint sub-pages are simple textual and image content holding web pages, describing the further data, surveying, remote sensing and GIS services offered by the Z2 Project in the Services section, as well as the About section that describes the RDM infrastructure and the Z2 project team for the web site visitors, the same holds also for the mandatory legal Imprint information page. These pages will not be further described in this section, instead the data catalog and the publication DB interfaces will be presented in more detail.

The redesign of the website, as described in section 12.4.1, enabled also to implement a comprehensive web site metrics application for access, usage and download tracking. The website statistics itself (access logs) are tracked using the open source PIWIK (PIWIK Contributors 2014) system, this system is extended by an additional detailed tracking of data access and downloads, as described in section 9.6.1. The web access statistics tracked by PIWIK are shown to the interested users from the PIWIK map widget placed under the news streams on the CRC806-Database frontpage.

9.1. Frontpage, news and blog

The start page, as shown in figure 9.2, contains the news streams of the newest data sets, publications and the geodata resources published in the according repositories, the blog interface and the visitor statistics.

Summarizing streams of five short entries lists of the latest, datasets, geodatasets and publications are implemented as boxes on the right side of the page, see figure 9.2. By following the links shown in this streams, the detail pages of data or publication records are shown. In case of geodata, the user will be forwarded to the interactive maps detail view of the geodataset. These streams are also available in RSS and Atom format. This allows users to subscribe to this feeds in a client application supporting this formats, like for example Mozilla Thunderbird, to automatically get notified, if new content is added to the CRC806-Database. A map showing the locations of the web applications users is included at the bottom right of the CRC806-Database frontpage. The map is enabled through a JavaScript based map widget provided by PIWIK, see section 5.1 and 6.4, that is facilitated to deliver the visitor statistics of the web application. The latest eight teaser boxes, including title, author, publication date and short abstract (the teaser), for news announcements and blog posts constitute the main content of the front page. The blogs and news are also served via RSS and Atom feeds, as described with technical details in section 6.4.

The overall web design of the CRC806-Database aims to mimic the *flat design* (Pratas 2014) approach to web design. Flat design embraces simplicity as its core principle, and aims to design and implement clean UI that do not disturb the user from the functionality of the application. It is easier to quickly convey information while still looking visually appealing and approachable. The large title picture on the frontpage is also inspired by this design approach. This design approach is applied by major UI implementing companies, such as Microsoft, Apple and Google for example. The technical flat design is to a considerable part delivered by the Bootstrap UI

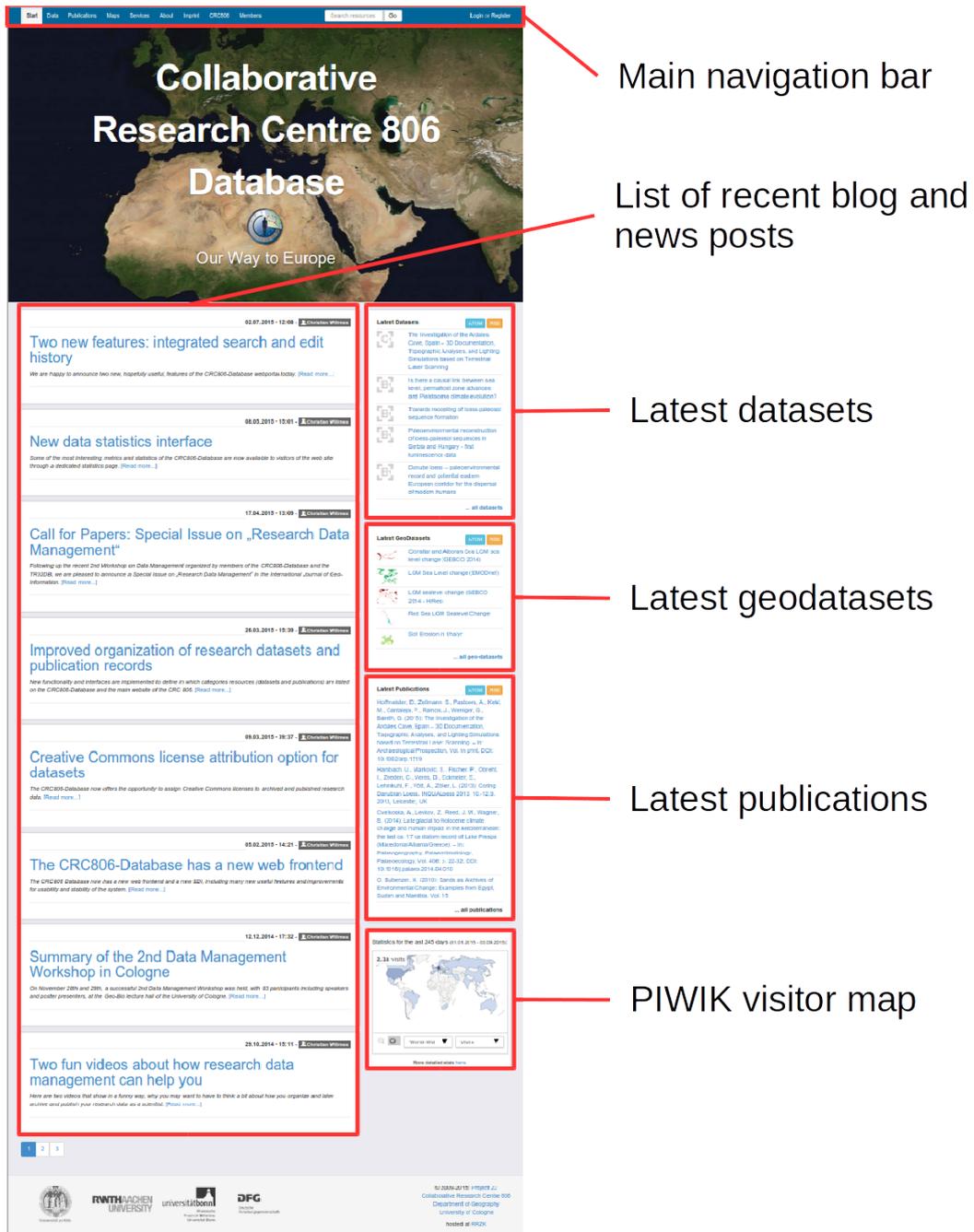


Figure 9.2.: The frontpage of the CRC806-Database web application. Source: Screenshot.

framework, that also provides seamless design features, which allow that the web applications are functional on small screens and mobile devices too, also known under the concept *responsive design*.

9.2. Data catalog

The data catalog of the CRC806-Database RDM infrastructure is a complex interface to facilitate all features, that are required to upload and archive, describe, and annotate research results, as well as features for discovering (search, browse, and filter) the contents of the data repository.

The figure consists of two screenshots of the CRC806-RDM data catalog interface. The left screenshot shows the main search and filter interface. It features a navigation bar at the top with links for 'Start', 'Data', 'Publications', 'Maps', 'Services', 'About', 'Imprint', 'CRC806', and 'Members'. Below the navigation bar, there is a search bar and a 'Go' button. The main content area is divided into two columns: 'All Datasets' on the left and 'Search' on the right. The 'All Datasets' column displays a list of dataset records, each with a title, a brief description, and a 'Project' icon. The 'Search' column contains various filters, including 'Sorting' (Creation Date), 'Projects' (A, B, C, D, E, F, Z), 'Additional Filters' (Find datasets with 'GIS', 'DOI', 'CRUI'), 'Location' (Filter by location), 'Time' (Find datasets from 'Temporal Interval', Start, End, Filter by time), and 'Keywords' (0k, 19k, 21k, 30k, 33k, 10k).

The right screenshot shows a detailed dataset page for 'Köppen-Geiger classification of MPI-ESM-P LGM simulation'. The page includes an abstract, resources, related spatial datasets, a bibliography, and additional metadata. The 'Abstract' section contains the following text: 'This geospatial dataset, in raster and vector format, is a Köppen-Geiger climate classification of the MPI-ESM-P Last Glacial Maximum (21k-18k BP) 10 yr model simulations according to the MGR 10.218 experiment. The classifications were computed using the Python pyGRASS library and GRASS GIS.' The 'Resources' section lists two datasets: 'MPI-ESM-P LGM 10yr Köppen-Geiger classification.tif' and 'LGM Köppen-Geiger Shapefile.zip'. The 'Related Spatial Datasets' section shows three maps: 'Köppen-Geiger LGM Capped', 'Köppen-Geiger LGM', and 'Köppen-Geiger LGM (Raster)'. The 'Bibliography' section lists a reference: 'Witnes, C., Becker, D., Brocks, S., Hül, C., Barth, G. (2014) Köppen-Geiger classification of MPI-ESM-P LGM simulation. SFB/CRC 806 Database, DOI: 10.5880/SFB806.2'. The 'Additional Metadata' section includes: 'License: CC BY', 'Spatial: ["Point","Polygon","coordinates":["[[[-100,90],[-100,90],[-90,100],[-90,90]]]]]', and 'Temporal: Last Glacial Maximum'. The right sidebar contains a 'Project' section, 'Linked CRC 806 Authors', 'Dataset' information, 'License' (Attribution required, For commercial use, Derivatives are allowed), 'Spatial' map, 'Interval' (Name: Last Glacial Maximum, Onset: 25000 y.B.P., Ending: 19000 y.B.P., Description: Heinrich Type Global period, Region: Global), and 'Tags' (New classification, Data, Metadata, DOI, License, Attribution, SFB 806).

Figure 9.3.: The data catalog interface. Search, filter and browse interface on the left. Detail dataset page on the right. Source: Screenshot.

9.2.1. Catalog search and discovery

As shown in figure 9.3 on the left side, the catalog interface is structured into a main list of dataset records, displayed according to the current filter or query (by default most recent record first) on the left side, and tools to filter and search the repository on the right side.

The listed record boxes show a number of most interesting and relevant information of each record. These are the title, the first 200 characters of the description, the maintainer, creation date, the keywords, the number of resources and their data types shown as icons, as well as the project affiliations of the dataset. Additionally, indications if a CRC 806 DOI is minted for the resource, if it has spatial and if it has temporal annotations and according metadata available. It is possible to filter along these information, by just clicking on them. For example, by clicking on the project icon, the data repository will be filtered for datasets annotated with that project, and all matching records affiliated with this project will be returned, and shown on the list.

On the right side of the catalog interface, a number of boxes containing tools and features to search, filter and browse the repository are given. These tools are:

- i. a full text search box,
- ii. a sorting (by title, creation date and maintainer) feature,
- iii. a filter interface containing a hierarchical list of the clusters and projects,
- iv. and additional filters for CRC 806 DOIs, spatial, and temporal annotations,
- v. a location filter tool,
- vi. a temporal filter interface,
- vii. and the keywords (also known as topics and tags).

The spatial filter, linked in the *Location* box, on the right side of the search and browse interface (see left side of fig. 9.3), is provided to filter data by location. It is possible to either draw a bounding box (rectangle) on the map, that filters for all dataset that have their spatial extent within this box, and by a list of predefined regions (e.g. Europe, Africa, Germany, etc.), that filters for datasets with spatial extents within the predefined region. Furthermore, it is possible to browse the datasets, that contain a spatial annotation on a map. The dataset map is linked from the top of the records list, and shown in figure 6.4.

Additionally, a temporal filter is implemented, that allows to filter the catalog by a time interval or an event, and return all records, that are temporally within this interval, or annotated with the given event. It is possible to define a custom start and end date, as well as choose from a list of predefined time periods. Finally, it is possible to filter by the tags, that were annotated by the dataset maintainers. By clicking on a tag name icon, all datasets annotated with that tag are listed.

9.2.2. Dataset detail information

As shown in figure 9.3 on the right side, the dataset detail page is structured into, up to twelve, content boxes. The boxes contain the metadata the dataset is annotated with. The boxes are shown if the according metadata is available for the given dataset, and the content boxes are the following:

- i. Title and maintainer information,
- ii. Project affiliation,
- iii. Abstract,
- iv. CRC 806 authors linked to this dataset,
- v. Resources,
- vi. Citation example (if bibliographic annotation is provided),
- vii. License information,
- viii. Related resources and datasets,
- ix. Bibliography information,
- x. Spatial annotation,
- xi. Temporal annotation,
- xii. Topic category and Tags annotation.

The core content boxes i., ii., iii., and iv., are given on any dataset detail page. A title, the maintainer, an abstract (even if empty, e.g. not provided), the CRC 806 affiliations for projects and members are always rendered. The Resources box is also always shown, even if no resources are uploaded for the dataset. If resources are uploaded, they are rendered with an icon indicating the resource format, and information of when the resource was updated last time, as well as how often the resource was downloaded.

Below the resources of the dataset, the related resources (viii.) box is displayed. In this box, all the resources annotated as related, from the two other resource domains of the CRC806-Database are listed. Datasets and Publication records are indicated with according graphical icons, next to their titles. The related geodatasets are shown with thumbnails of a visualization of the according geodata WebGIS layer and their, as well linked title, below the thumbnail. If a bibliographic annotation is provided for the dataset, an example or suggestion for how to cite the dataset is provided in the Citation box (vi.). The spatial extent (x.) is shown as a rectangle (Bounding-Box), or as a location (Coordinate) on a small map on the right. Temporal metadata (xi.) is displayed in an artless box, simply displaying the data, event or interval name, the date or start and end date, as well as a short description of the interval or event. The License (vii.) is shown by the according CC logo linked to the according CC license text, as obtained from (Creative Commons 2015), and additionally by short summarizing statements of what is allowed or prohibited with the data. Bibliography information (ix.) is displayed like in the publication DB, in a simple table containing key and value pairs of the information. The bibliographic information can be exported in BibTeX format, by clicking on a link that is provided in the lower right corner of the box. And last, but not least the keywords, tags and topics are shown in a box (xii.). It is possible to click on a term, to show a list of all other datasets annotated with that term in the CRC806-Database system.

9.2.3. Data upload and publication

The data upload and publication process is realized directly through the public web application of the CRC806-Database. The upload feature is not available for non members. If the upload functionality is available depends on the individual user rights of the member.

For the upload of data an intuitive web form is provided, see left side of figure 9.4. The metadata input forms are organized in boxes for each metadata topic. The boxes are the same as described in the previous section 9.2.2: Title, Projects, Maintainers, Abstract, License, Bibliography, Spatial, Temporal, Tags, Additional Metadata (free number of textual key value pairs), Resources and related Data. The input form UI's are implemented with much care about usability and intuitivity. For example, the spatial annotations can be entered, by clicking on a map to enter a coordinate, or to draw a rectangle on the map to enter Bounding-Boxes. The licenses can be chosen by simple check-boxes. Projects and clusters can be annotated by clicking on their logos.

On the right side of figure 9.4, the form for editing existing datasets is shown. The form is similar to the upload form, but the metadata boxes and fields are filled with the annotations of

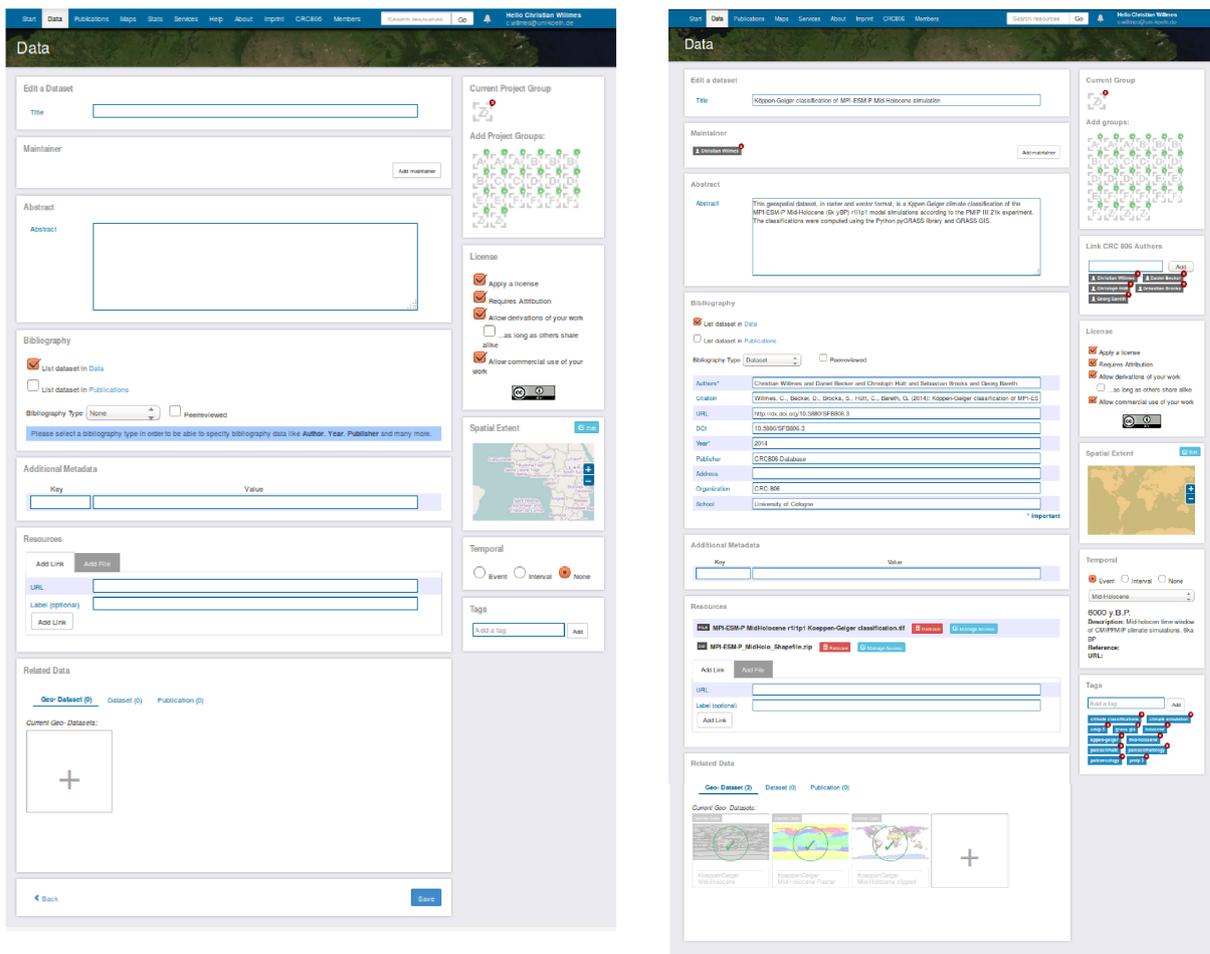


Figure 9.4.: Form for uploading a new dataset (left). Form for editing an existing dataset (right). Source: Screenshot.

the dataset that is currently edited. This, allows to directly edit, add or remove the information that should be changed. Positive feedback from CRC806-Database users support the claim, that this UI implementation has a good usability.

9.2.4. Access rights and user roles

The CRC806-Database implements a very fine grained access rights management for datasets and its resources. Figure 9.5 shows two screenshots of the UI for managing access rights. The UI element shown on the left side of figure 9.5 is the default view that a CRC806-Database member gets presented, who wants to edit the access rights of a previously uploaded resource. Thus, the default access rights to a resource is "Visible to public", and as such also "Visible to CRC 806 Members". If the check-box, next to "Visible to public" is removed, the resource would only be accessible to members of the CRC 806. If the check-box, next to "Visible to CRC 806 Members" is removed too, the UI element, as shown on the right side of figure 9.5 is shown. In this interface the CRC806-Database Member can define access to members of certain projects,

or only for certain other members of the CRC806-Database.

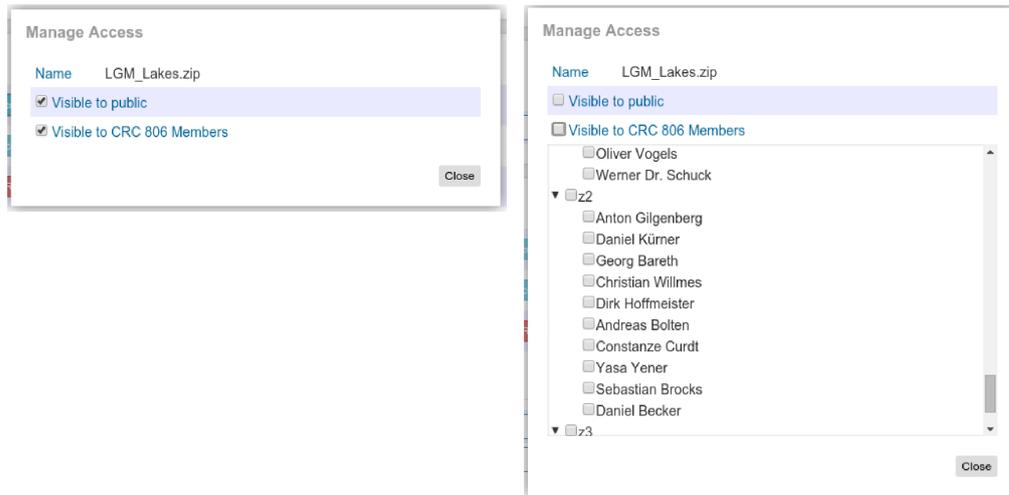


Figure 9.5.: Screenshot of access rights UI. Source: Screenshot.

Request access

To provide external visitors the opportunity to ask for access to an access restricted resource, the *Request Access* feature was implemented.

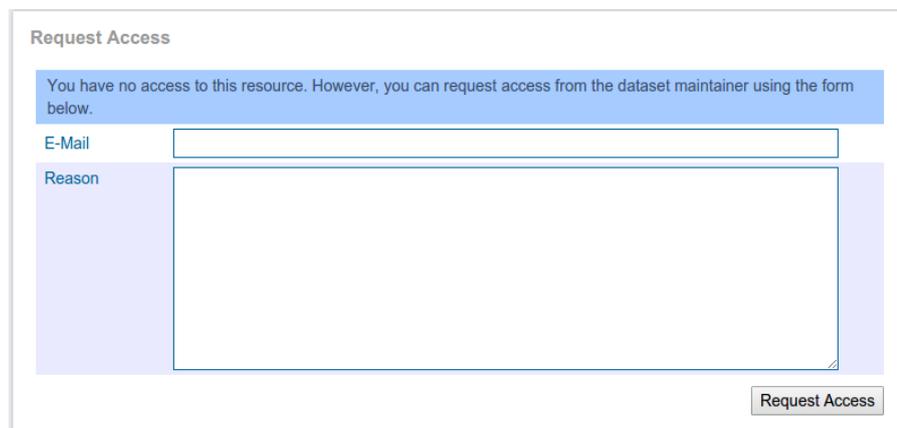
The screenshot shows a 'Request Access' form. At the top, a blue banner contains the text: 'You have no access to this resource. However, you can request access from the dataset maintainer using the form below.' Below the banner, there is an 'E-Mail' label followed by a text input field. Underneath that is a 'Reason' label followed by a larger text area for providing details. At the bottom right of the form is a 'Request Access' button.

Figure 9.6.: Screenshot of the Request Access form. Source: Screenshot.

If a site user tries to access a restricted resource, he or she is redirected to a form, shown in figure 9.6, that allows to request access to the given resource. The dataset owner(s), as well as the CRC806-Database administrators get an email including the access request. In this email two links are provided, one for granting access to the resource, and one for denying access. If access is granted, the requesting user gets an email with a link, containing an access token to the resource, that allows access to the resource for 7 days under the given access token. If access is denied, the owner needs to provide a reason for denying the access, that will be filed

in a database table of the Typo3 backend, and will be send to the requesting user per email.

9.3. Publications DB

The main interface of the publications DB is based on the data catalog extension, and thus has many similarities. As technically described in section ??, the publication Laser DB reuses most of the data catalog extensions code by a technique called service injection, as well as data template reuse. Non the less, the display is developed in own specialized views, that are explained here in the following.

The figure consists of two side-by-side screenshots of a web application interface for a publications database. The left screenshot shows a list view of publications. At the top, there is a navigation bar with 'Publications' selected. Below it, a search bar and a list of publications are visible. Each publication entry includes a title, authors, project name, and resources. The right screenshot shows a detailed view of a specific publication titled 'The Investigation of the Ardales Cave, Spain – 3D Documentation, Topographic Analyses, and Lighting Simulations based on Terrestrial Laser Scanning'. This view includes an abstract, a list of resources (including a PDF file), a bibliography table, and additional metadata like the license (CC BY).

Figure 9.7.: The publications DB interface. List, search, filter and browse interface on the left. Detail metadata information page on the right. Source: Screenshot.

9.3.1. Publications DB Catalog

Figure 9.7 shows on the left side the main publications DB interface, for listing, browsing, filter and searching the bibliographic database. On the left side of this interface the current result set of bibliographic records is shown in form of a list of so called record boxes. These boxes contain the most relevant information of each bibliographic records, that are: citation (as title), publication type, affiliated projects, attached resources, record maintainer, an indication if the publication is peer-reviewed, and indications if spatial and temporal metadata is provided. Similar to the catalog interface, on the right side next to the records list, some filter and search tools are provided. These tools are: a full-text search, a sorting (by publication date, record creation

date, title) interface, a cluster / project filter, a publication type filter, spatial and temporal filters similar to the implementation of the data catalog interface, and a keywords filter, that is similar to the tags filter implementation of the data catalogue.

9.3.2. Publication detail information

On the right side of figure 9.7, the bibliographic record detail information page is displayed. Also similar to the data catalog implementation, the metadata information are displayed in boxes for each sub category. The boxes are: title box including maintainer and creation date, abstract, bibliographic metadata according to BibTeX schema, project information, related CRC 806 Authors, a citation suggestion (for copy & paste) including BibTeX export feature, License information, a map showing spatial metadata if available, and a box indicating temporal metadata if available.

9.3.3. Publications on main CRC 806 website

To display the publications of the CRC 806, stored in the publications DB, on the main website of the project, a Joomla (Open Source Matters, Inc. 2015) plug-in was developed, see section 6.2 for a detailed description. The plug-in interfaces the Typo3 publications DB middleware, that implements a custom API endpoint to retrieve the bibliographic data via API calls.

Figure 9.8 displays how the publication records are rendered on the main CRC 806 website¹. By interfacing the Typo3 publication DB Extension middleware directly, it is possible to render the bibliographic informations completely according to the web design of the main CRC 806 website.

On the left side, the main publication list of all records, sorted in the CRC 806 publication DB date are shown. Also some filters are implemented, for example by clicking on one of the project icons in the side bar, the according projects publication list is shown, as rendered on the right side of figure 9.8. A full-text search from the main website is not yet implemented though. In this case the users are forwarded to the CRC806-Database main publication DB UI, as described above in section 9.3.1.

The right side screenshot in figure 9.8, shows how the bibliographic records are rendered on the projects *Publications* sub-pages on the main CRC 806 website. Additionally, to the publications lists, an interface to render the projects datasets published through the CRC806-Database was implemented. This interface is available form the *Project Data* sub-pages of each project. This interface was implemented almost similar to the publications list, only the API request differs in requesting datasets and not publications records from the Typo3 custom developed extension middleware endpoint. In both lists, the titles are linked to the detail publication record page on the main CRC806-Database website, as also shown on the right side of figure 9.7.

¹<http://www.sfb806.de/>, accessed: 2015-12-16.

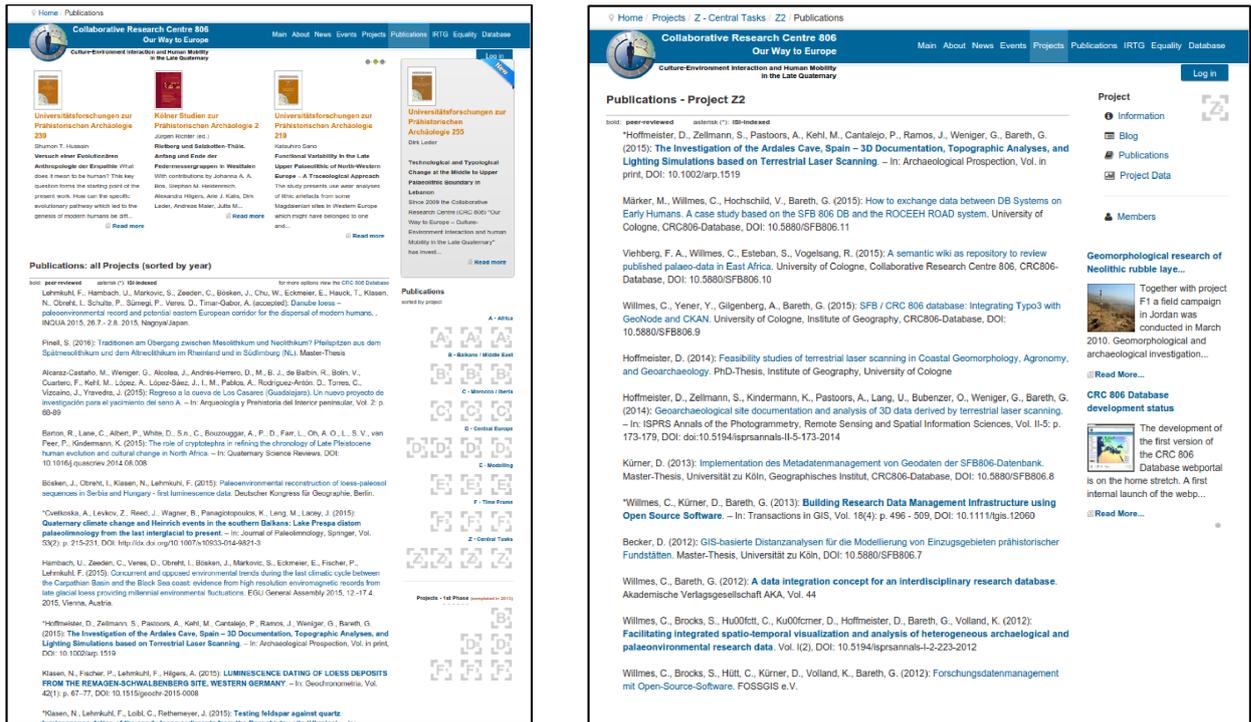


Figure 9.8.: The publication lists rendered on the main CRC 806 website. Source: Screenshot.

9.3.4. Publications on IRTG website

Similar to the plug-in for the main CRC 806 website, a Wordpress (Wordpress Contributors 2014) plug-in for the IRTG website² was developed, to allow the display of publication record for the doctoral students and IRTG alumni profile pages, see section 6.2 for a detailed technical description of the implementation. As shown in figure 9.9, on the IRTG website, the publications are rendered on the doctoral students CV like profile pages. The titles are linked to the detail pages of the publications in the CRC806-Database web publications DB main web application.

The right side of figure 9.9 shows the publications list in more detail. For the IRTG website no additional features like filters were implemented, because they are not needed and requested for the scope of the IRTG website.

The integration of the publication DB into the two other CRC 806 websites (main website and IRTG website), has some notable advantages for the CRC806-Database website. The first obvious advantage is of course, that this interlinking from two websites directs more visitors to the CRC806-Database website. The second not so obvious but also very helpful advantage is the fact, that through the relatively high number of links, from the linking of the publications detail pages of the lists on the CRC 806 main and the IRTG websites, the Google page rank increases, what results in more visitors that are directed from Google at the CRC806-Database website.

²<http://sfb806irtg.uni-koeln.de>, accessed: 2015-12-16.

The screenshot displays a user profile for Dr. Anne Böhm on the IRTG website. The profile includes a header with navigation links (HOME, ABOUT, SCHEDULE, MEMBERS, ALUMNI, CONTACT, LOGIN), a profile picture, and a bio: 'Dr. Anne Böhm, Aufgebauer'. Below the bio, there are sections for 'Research Title' (with a sub-section 'Die Klima- und Umweltgeschichte des Balkans während des letzten Glazials und des Holozäns, rekonstruiert anhand von Sedimenten des Prespases (Mazedonien/Albanien/Griechenland)'), 'Education' (listing a PhD from the University of Cologne in 2010 and a BSc from the University of Bonn in 2009), and 'Publications' (listing several peer-reviewed articles). To the right of the profile, there is a 'Publications' section with a list of articles, including 'Climate variability over the last 92 ka in SW Balkans from analysis of sediments from Lake Prespa' and 'Distinct lake level lowstand in Lake Prespa (SE Europe) at the time of the 74 (75) ka Toba eruption'. At the bottom of the profile page, there is a link 'Back to Alumni overview'.

Figure 9.9.: The publication lists rendered on the IRTG website. Source: Screenshot.

9.4. Members directory

For managing access to data resources, a user management system for the CRC806-Database was requested, see section 3.1.3, designed and implemented, see section 6.3. The result is a user management system that is based on the Typo3 URM technology, but enhanced with CRC806-Database specific functionalities. The members directory consists of a main CRC806-Database users list, shown in figure 9.10 on the left side, and profile pages for each member, shown on the right side of figure 9.10.

The Members list shows the members user accounts in alphabetical order, each member is represented by structured information containing a profile picture, the name, the project affiliation, the academic position, and the count of affiliated datasets and resources published in the CRC806-Database. On the right side of the list, two boxes are present, the upper one contains a full-text search interface, the lower one contains a project based filter, as it is known from the data catalog interface. The list is sortable by Name, academic position, CRC 806 project, and number of owned datasets. This last metric was included to provide a small incentive for project members to publish more datasets in a kind of competition with their colleagues.

The members profile pages are separated into three information boxes. The first box contains structured personal informations like, *name*, *project affiliation*, *contact details* and a *profile picture*. The second box lists all datasets and publications the member is annotated as author of. The third box lists all datasets of which the member is annotated as maintainer. This lists are helpful for the members to have an instant overview of their datasets, and also the datasets of

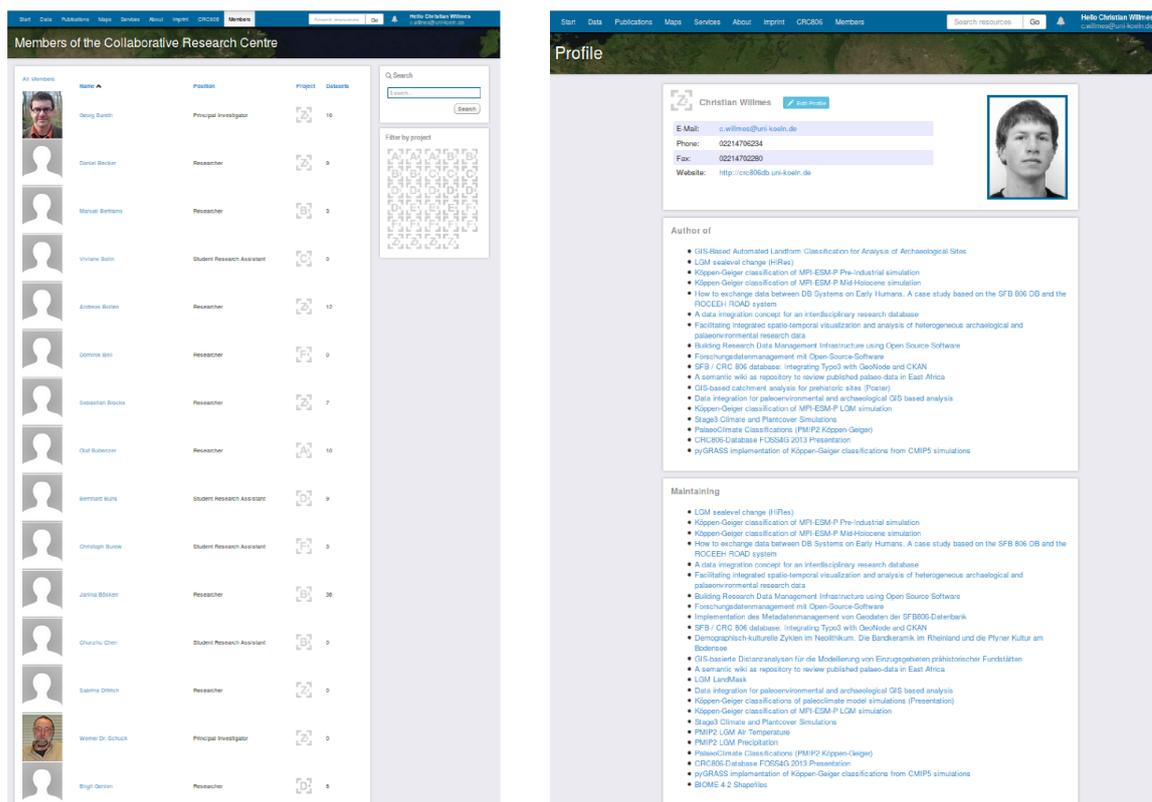


Figure 9.10.: The members list (left). A members profile page (right). Source: Screenshot.

their colleagues.

9.5. Integrated search

As described in section 6.5, an interface to integrate the three repositories (data catalog, publication DB and maps) was implemented under the name integrated search. The integrated search interface allows to search the CKAN backend, including data catalog and publication DB entries, as well as the GeoNode SDI backend in one integrated interface. This allows to yield results from data, publications and maps in one result list, as shown in figure 9.11.

The search is triggered on the CKAN and on the GeoNode backend, after the responses of both backends are received, a list ordered by creation date is compiled and shown as response, see figure 9.11.

The interface is held simple, no extra section link was included into the main navigation bar, but directly a full-text search field. If a search is triggered through that field, the user is redirected to the custom integrated search interface, that displays the search results from the three repositories according to the search query. The result list is further sortable and filter-able. UI interfaces implemented as drop-down fields are implemented to facilitate this, see figure 9.11 on the right side.

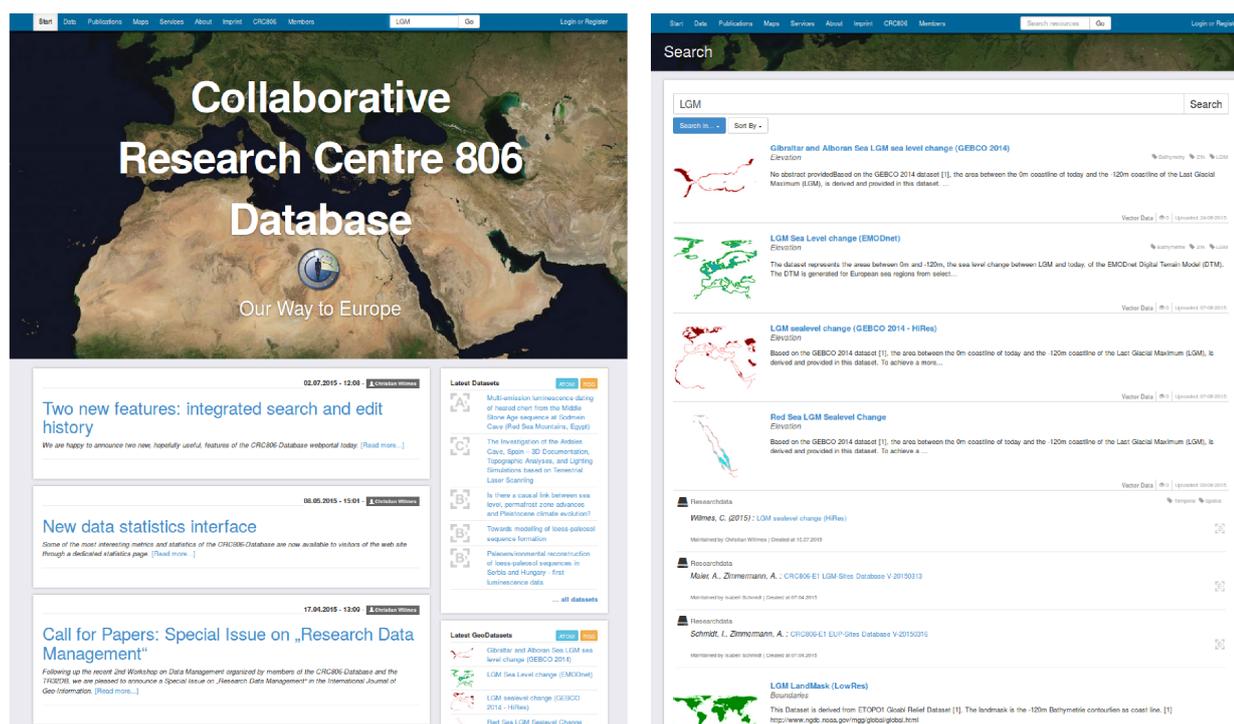


Figure 9.11.: The interface of the integrated search. Integrated searches are triggered through the search box in the navigation bar (left). The results are shown in a list (right). Source: Screenshot.

9.6. Data metrics and visitor statistics

A data metrics interface, as well as publicly accessible user visitor statistics interface were developed and implemented, to allow the CRC806-Database users and also interested website visitors, to gain some insight into how the CRC806-Database is used, and what kind of data is available from the platform.

9.6.1. Data metrics

For this application, data metrics are understood as the quantitative descriptive statistics about the data that is published and stored in the CRC806-Database. For now, only five measures are shown on the publicly accessible web site, see the screenshot of the interface in figure 9.12. The measures are:

- i. Dataset types,
- ii. Incoming datasets,
- iii. Data by cluster,
- iv. Most popular resources,
- v. Most popular datasets

The statistics are rendered, according to the interval, that can be defined in the top box, by

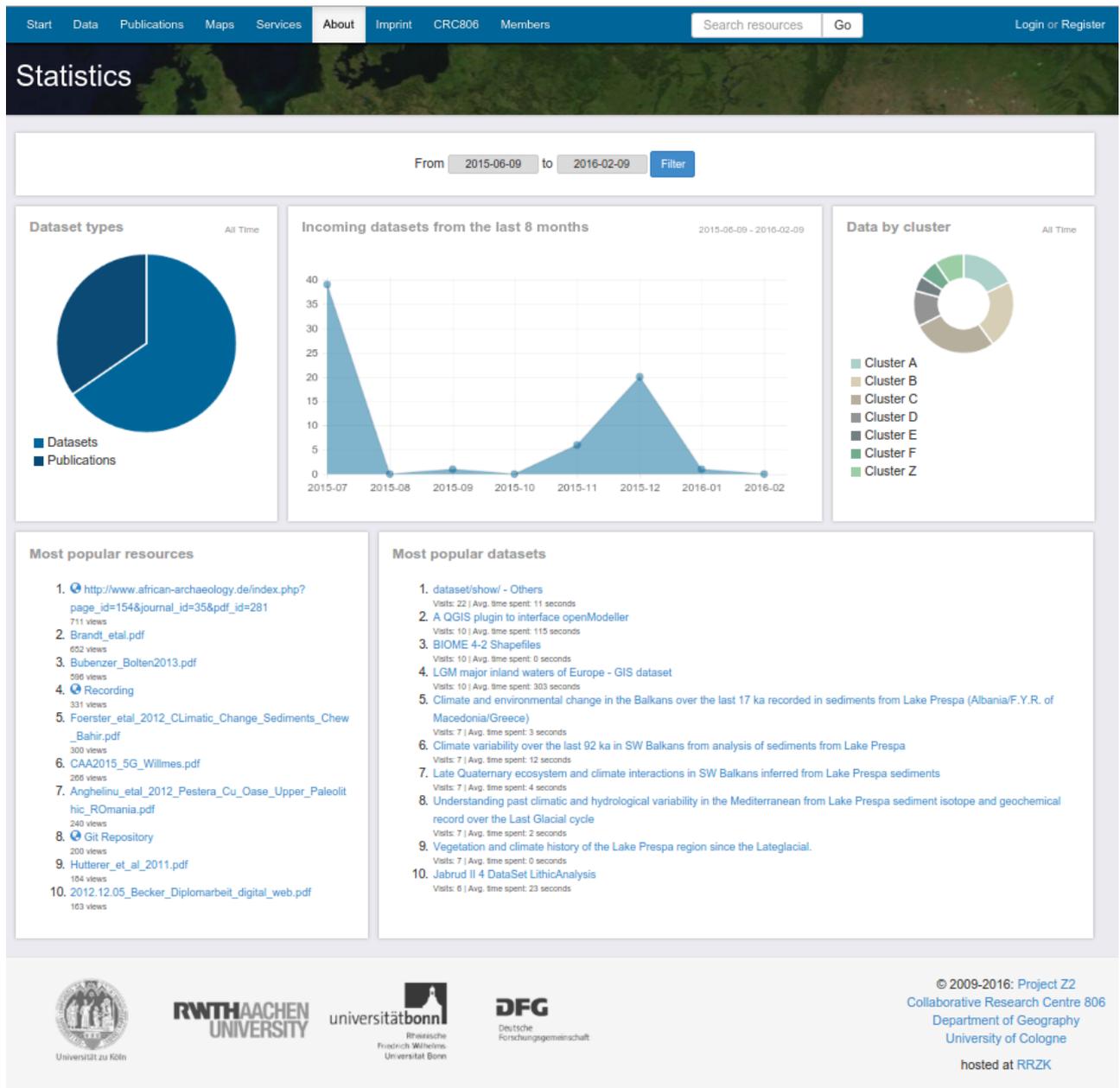


Figure 9.12.: The data metrics interface of the CRC806-Database. Source: Screenshot.

defining a start (From) and a end (to) date. But two measures are excluded from the filter. That are the *Dataset types*, in the top left, that show the ratio between Datasets and Publications in the CRC806-Database. And the other measure is the *Data by cluster* statistic. These two measures are shown for the whole database, and are not filtered by a time interval. In the center of the page, the *Incoming datasets* graph is shown. The graph shows the new created datasets and publication records in the CRC806-Database by month. Defined by the time interval filter are the *Most popular resources* measure, that renders the top ten most accessed resources. The data catalog and publications DB Typo3 extension middleware logs each access in its DB

persistence backend. Also, the *Most popular datasets* measure is dependent on the time interval defined in the filter at the top of the page. This measure shows the top 10 most accessed datasets and publication records in the CRC806-Database.

9.6.2. Visitor statistics

The visitor statistics of the CRC806-Database are captured and logged with the help of the OSS tool PIWIK (PIWIK Contributors 2014). This applications offers a comprehensive UI and visitor and usage statistics data collection, see figure 9.13.

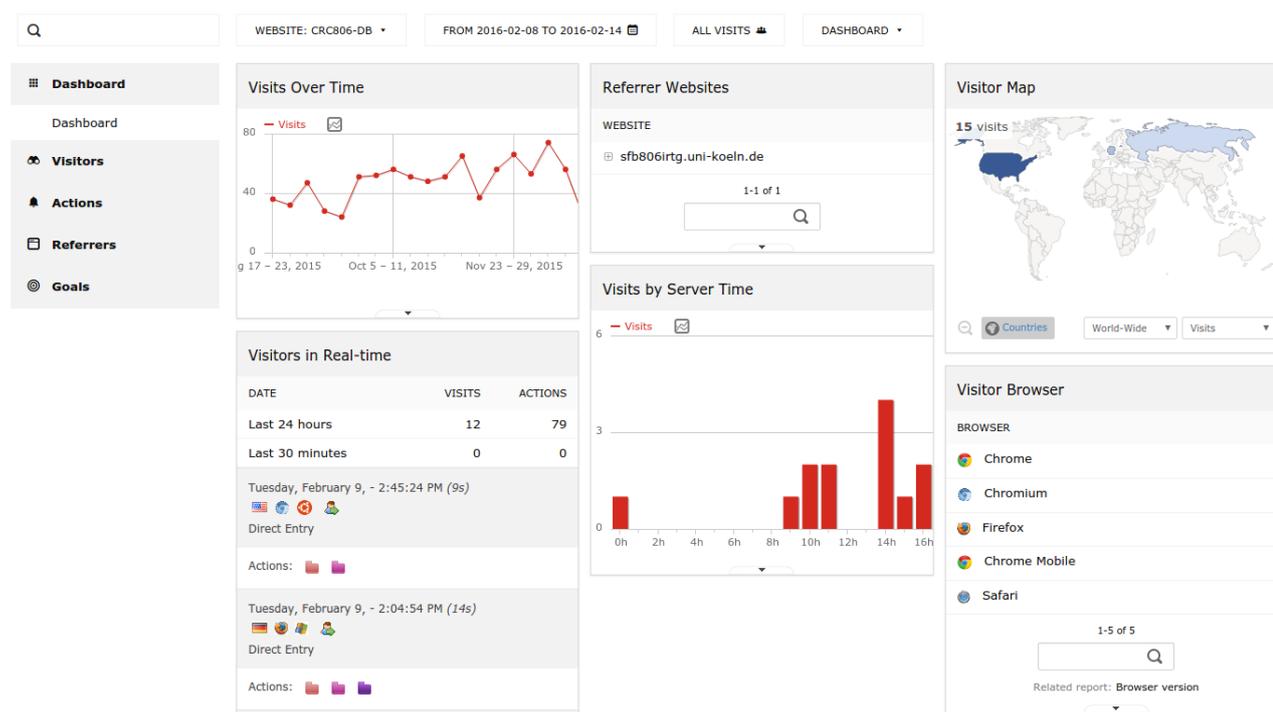


Figure 9.13.: The PIWIK visitor statistics interface. Source: Screenshot.

All statistics shown in PIWIK depend on the currently selected time interval, that can be defined through an according UI item at the top of each page. Figure 9.13 shows the dashboard of the CRC806-Database PIWIK instance. this dashboard can be customized with statistics measures of choice. In the CRC806-Database version, the dashboard consists of the following items:

- i. Visits Over Time
- ii. Visitors in Real-time
- iii. Referrer Websites
- iv. Visits by Server time
- v. Visitor Map
- vi. Visitor Browser

Visits Over Time shows a graph with number of visitors, for the currently selected interval, in the screenshot (fig. 9.13) its per week (ranging between 35 and 80 visits a week). *Referrer*

Websites show visits, that were refereed by links from other websites. The *Visitor Map* shows the locations of the visitors in the current interval on a map. *Visitors in Real-time* shows the current and recent visitors in a stream format. *Visits by Server Time* shows at which time of the day, the visitors are visiting the website. And *Visitor Browser* show the browsers that are used by the visitors, sorted by frequency.

9.7. API endpoints

The RDM Infrastructure is additionally to the web based UI programmatically accessible via API endpoints. The data repository can be accessed through the CKAN Action API, as described in section 9.7.1, this is also the case for the publications DB, that is also managed by the same CKAN instance.

9.7.1. CKAN API

CKAN instances can be harvested by other CKAN instances. The `ckanext-harvest` extension can automatically import (“harvest”) datasets from multiple CKAN websites into a single CKAN website, and also provides a framework for writing custom harvesters to import data from non-CKAN sources. This harvesting is facilitated through the CKAN action API, as introduced in section 5.2.2 and 6.1.1. The CRC806-Database CKAN instance endpoint is available from this URL:

```
http://sfb806ckan.geographie.uni-koeln.de/ckan
```

Accessing the CKAN API, it is possible to build complete applications on top of the CRC806-Database backend, but the API is protected by an API key, that would only be granted, if project Z2 would agree to the goals of that external application. The details, of how to access the endpoints are well explained in the available CKAN documentation (Open Knowledge Foundation 2014; Open Knowledge Foundation 2016).

9.7.2. Typo3 API endpoints

The CRC806-Database Typo3 endpoint that is used by the external publication and data lists on the two further CRC 806 websites, as described in section 6.2.3, are in theory also available for thrid party applications.

```
http://crc806db.uni-koeln.de/dataset/getJson
```

This URL can be appended with some parameters, for search string and certain filters. Though because this interface is at this point openly for internal use, the details of the API will not be published here. The RDFa markup and the RDF export functionality, as described in section 4.6, of the CRC806-Database can be regarded as an API too.

9.8. Administration console

An administration console was developed to handle a set of regularly repeating administrative tasks directly through the CRC806-Database web application. The console offers an UI for the management of CRC806-Database members, for managing user roles and rights. Additionally, it has tools for general repository management tasks like the moderation of resource access requests, or requests for maintainership, and of course DOI minting.

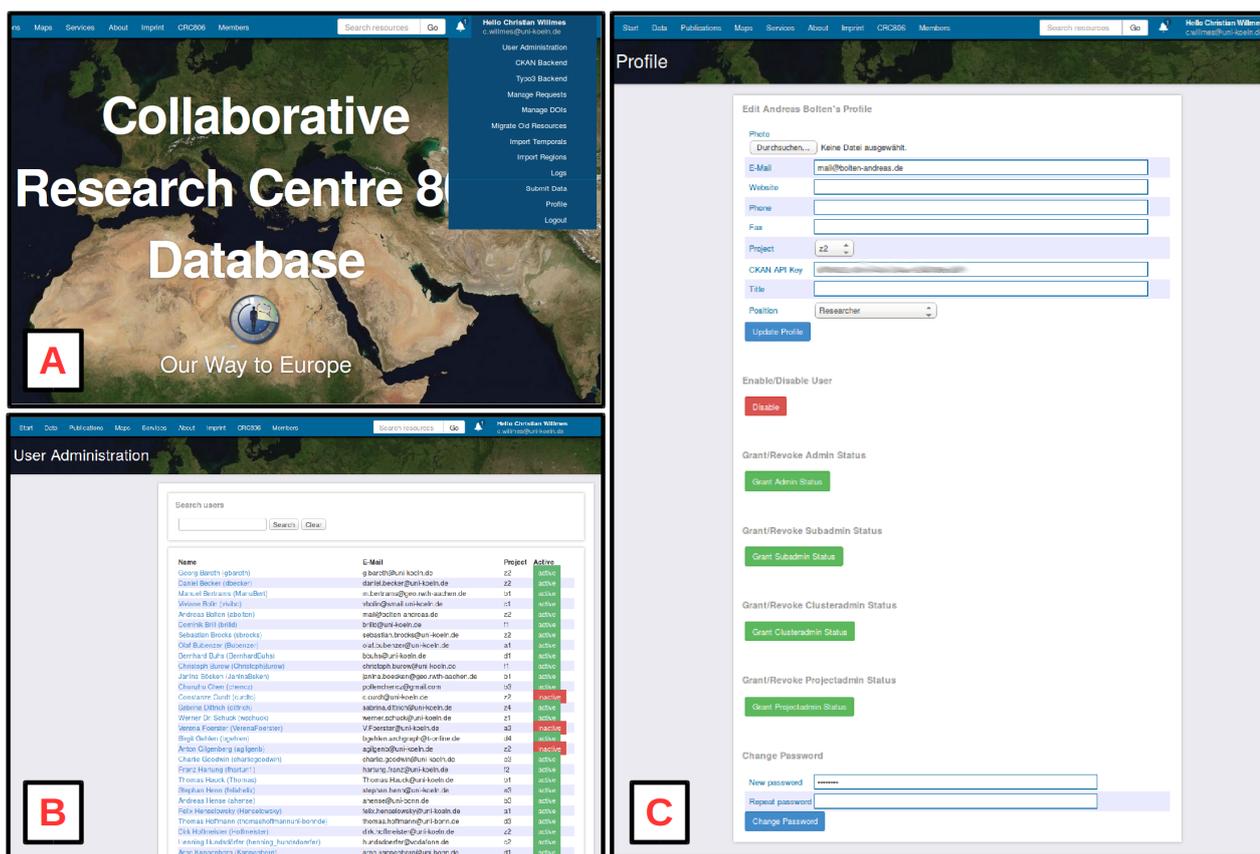


Figure 9.14.: Screenshots of the CRC806-Database admin console. A: Admin console drop-down menu. B: Member admin user list view. C: Member profile edit view. Source: Screenshot.

The admin console is only shown to users with role *Administrator*. An administrator gets additional functionalities and menu items in the context drop-down menu, as shown in figure 9.14 A. The additional items are the following:

- i. User Administration,
- ii. CKAN Backend,
- iii. Typo3 Backend,
- iv. Manage Requests,
- v. Manage DOI's,
- vi. Import Temporals,

- vii. Import Regions,
- viii. Logs.

The *User Administration* (i.) view is shown in figure 9.14 B. The view consists of a list of all registered users of the CRC806-Database. By clicking on a users name, the *Profile edit* view is shown, see figure 9.14 C. In this view, it is possible for an administrator to edit all settings of a user profile, or disable it completely, and as well grant or revoke certain rights or roles. The following roles are available:

1. Admin Status,
2. Subadmin Status,
3. Clusteradmin Status,
4. Projectadmin Status.

A user with admin status has all rights possible. Clusteradmins, can edit all Datasets related to the Cluster, the Clusteradmin has the rights for. Projectadmins can edit any Dataset related to the project, the Projectadmin is assigned to.

The *CKAN Backend* (ii.) and *Typo3 Backend* (iii.) items are just simple links to the according backend login pages. The *Manage Requests* (iv.) UI allows to moderate Access Requests, see section 9.2.4. Following this link directs to a simple UI showing pending requests, and the details of the request, including which resource was requested and what reason was provided, as well as the request date, and any follow-up conversation between requesting individual and the dataset maintainer. The *Manage DOI's* item (v.) leads to a UI for handling DOI requests, see figure 9.15. As described in section 6.1.4, it is possible to request DOI's for datasets of the data catalog repository. The DOI requests are reviewed by administrators of the CRC806-Database, if the request is accepted, the DOI will be requested and minted automatically at our DOI partner WDC-Terra, as described in section 6.1.4, by the Typo3 extension using the DataCite API.

The *Import Temporals* (vi.) and *Import Regions* (vii.), are small tools to read the temporal definitions and the Regions definitions from the SMW based CRC806-KB into the Temporal and Spatial annotation lists of the CRC806-RDM infrastructure, as described in section ???. And the Logs item (viii.) leads to the logs UI, where all interactions of logged in members are logged. Also major or fatal errors, that occur on the web application are listed here.

DOI Requests

Action	Title	Dataset
Edit	A QGIS plugin to interface openModeller	10.5880/SFB806.16
Edit	LGM paleoenvironment of Europe - Map	10.5880/SFB806.15
Edit	LGM major inland waters of Europe - GIS dataset	10.5880/SFB806.14
Edit	"Hunter-gatherer situations" – Keynote speech held at the 11th Conference on Hunting and Gathering Societies (CHAGS 11) in Vienna, Austria, 7th September 2015	10.5880/SFB806.13
Edit	CRC806-E1 LGM-Sites Database V-20150313	10.5880/SFB806.12
Edit	pyGRASS implementation of Köppen-Geiger classifications from CMIP5 simulations	10.5880/SFB806.1
Edit	A semantic wiki as repository to review published palaeo-data in East Africa	10.5880/SFB806.10
Edit	How to exchange data between DB Systems on Early Humans. A case study based on the SFB 806 DB and the ROCEEH ROAD system	10.5880/SFB806.11
Edit	Köppen-Geiger classification of MPI-ESM-P LGM simulation	10.5880/SFB806.2
Edit	Köppen-Geiger classification of MPI-ESM-P Mid-Holocene simulation	10.5880/SFB806.3
Edit	Köppen-Geiger classification of MPI-ESM-P Pre-Industrial simulation	10.5880/SFB806.4
Edit	Large-eddy simulation of Convective Turbulent Dust Emission	10.5880/SFB806.5
Edit	Jabrud II 4 DataSet LithicAnalysis	10.5880/SFB806.6
Edit	GIS-basierte Distanzanalysen für die Modellierung von Einzugsgebieten prähistorischer Fundstätten	10.5880/SFB806.7
Edit	Implementation des Metadatenmanagement von Geodaten der SFB806-Datenbank	10.5880/SFB806.8
Edit	SFB / CRC 806 database: Integrating Typo3 with GeoNode and CKAN	10.5880/SFB806.9

Approve DOI for A QGIS plugin to interface openModeller

Dataset: A QGIS plugin to interface openModeller

Uri:

Prefix:

Suffix:

Requester: Christian Willmes

Metadata

The below XML-Document has been successfully checked against the [DOI Schema Version 1.0](#).

```

<?xml version="1.0" encoding="utf-8"?>
<resource xmlns="http://datacite.org/schema/kernel-3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://datacite.org/schema/kernel-3 http://www.w3.org/2001/XMLSchema-instance"
  resourceType="resourceTypeGeneral" dataset="dataset" />
<identifier identifierType="DOI">10.5880/SFB806.16</identifier>
<creator>
  <creatorName>Daniel Becker</creatorName>
</creator>
<creator>
  <creatorName>Christian Willmes</creatorName>
</creator>
<title>A QGIS plugin to interface openModeller</title>

```

Wrong metadata? Please consider correcting the dataset's bibliography, to avoid editing raw xml.

Figure 9.15.: The UI for administering DOI requests. The left side shows the list of DOI requests and its status. The right side shows the editing UI for DOI datasets. Source: Screenshot.

10. CRC806-SDI

The CRC806-SDI, consists of a Typo3 extension, that builds the frontend, and is part of the main CRC806-Database web application. And a server infrastructure, consisting of GeoNode, MapServer and MapProxy applications, for delivering the OGC conformal OWS implementations, as well as concepts and tools for the spatial data management. In the following of this chapter, first the Maps frontend of the SDI will be described in section 10.1. This is followed by a description of the SDI endpoints, meaning the OWS interfaces, in section 10.2. Finally, the GeoNode backend web application is described, that is used for the management of spatial data services within the CRC806-Database, see section 10.3.

10.1. Maps

The frontend of the CRC 806 SDI is simply called *Maps*, and is a part of the main navigation bar of the CRC806-Database web application in the same way as the *Data* and *Publication* repositories. As described in section 7.3, the *Maps* interface is developed as a custom Typo3 *Extbase & Fluid* extension, that provides an interface integrated into the CRC806-Database website for the GeoNode based SDI backend. This gives the SDI frontend on the same accessibility, as well as the look and feel of the two further repositories for research data and publication records of the CRC806-Database infrastructure.

The *Maps* application interface offers a catalog style interface, as described in section 10.1.1, analogue to the data catalog and the publication catalog interfaces, including a list view and a WebGIS based spatial dataset detail view.

Look and feel of the *Maps* interface was designed to accommodate the remaining overall CRC806-Database website, using the same style criteria, like colors, fonts and partitioning of the websites into content boxes. Comfortably, in web design this is well possible by reusing the CSS definitions of the website in the *Maps* section accordingly.

10.1.1. Catalog interface

In figure 10.1 on the left, a screenshot of the catalog list view interface to the CRC806-Database SDI is shown. The geodatasets are displayed in a list of boxes containing the most relevant meta informations about the geodatasets, next to a thumbnail (small picture) of the geodata. The meta informations displayed in each box are:

- title,

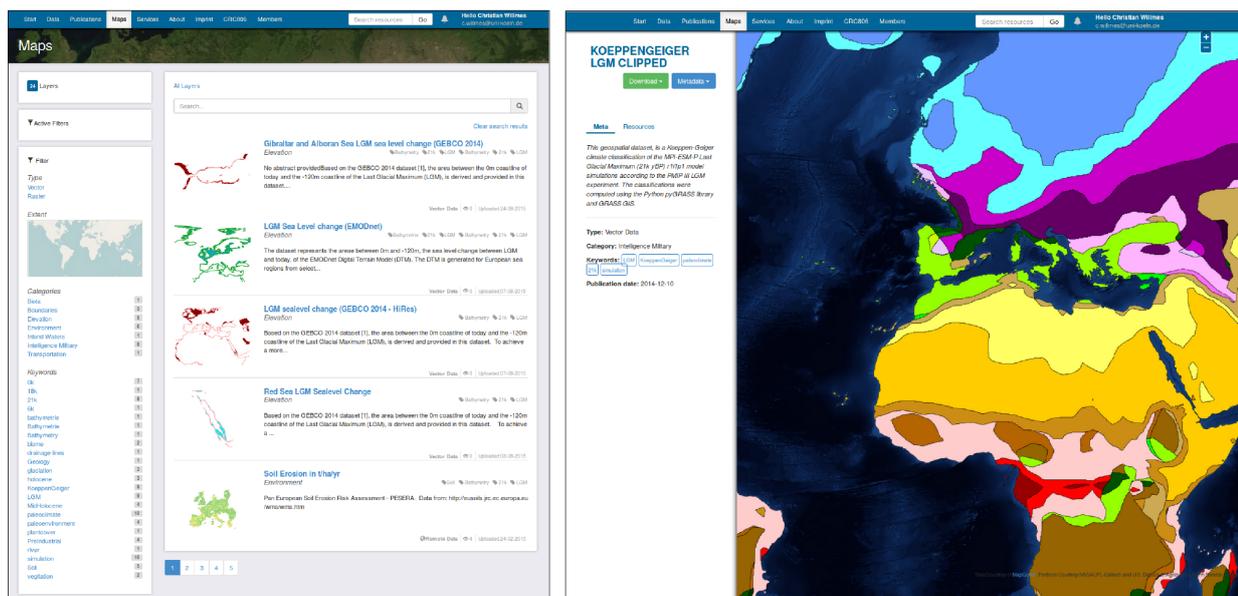


Figure 10.1.: The maps SDI catalog style interface (left). WebGIS based geodata layer detail view (right). Source: Screenshot.

- topic category,
- keywords,
- geospatial representation type (raster or vector),
- access count,
- and upload date.

A full text search box is placed above the layers list. On the left side of the page, further boxes with informations and filter capabilities are present. The filters are, by geospatial representation type, a spatial (map based) filter of course, by category and by keyword.

The full-text search as well as the browsing and interfaces are calling the GeoNode REST API, as described in section 7.1 and 7.3 in detail. This means the application is at this stage not calling the SDI's OWS infrastructure, because the performance as well as the filter capabilities of the REST API are richer than the available GeoServer, pycsw, MapServer and MapProxy OWS interfaces.

Clicking on a tag in the geodataset boxes in the list view, will filter the repository according this tag, the same is implemented for author names, and topic category.

Additionally, the catalog allows faceted browsing, meaning, it is possible to combine several filters to refine a search. For example a user can first do a full text-search about "Last Glacial Maximum (LGM)", then define a spatial region using the spatial filter, and additionally a keyword like "Glaciation". The result would be all layers with the term "LGM" in their title or abstract, covering at least parts of the selected spatial region and matching the selected keyword. This allows to refine the query results quite precisely.

WebGIS detail view

As shown in figure 10.1 on the right side. The detail map view is divided into two parts. On the left, an information panel is shown, on the right an OpenLayers (OpenLayers Contributors 2015) based web map for dynamic visualization of the geodata is shown. It is possible to zoom and pan the geospatial data layer, overlaid and placed on a world map for orientation. It is possible to change the base map to one of two further available maps, via in OpenLayers *Layer Switcher* UI element placed in the top left of the map window. Default is an Imagery layer, provided by MapQuest, alternatives are the standard OSM map and a topographic map based on OSM data also provided by MapQuest.

The information panel on the left displays the metadata information, it is divided into two tabs, *Meta* and *Resources*. The *Meta* tab displays information like abstract, type, category, keywords and publication data. Additionally, the *Resources* tab shows related resources (see figure 10.2), like the dataset pages of the data catalog, if the shown spatial data is also available as GIS Datasets from this repository, as well as the links to the OWS endpoints for access of the web service with third party applications, like Desktop GIS or external WebGIS, as further described in the following section.

Data access and download

The main feature of the information panel are the data download and the metadata download capabilities. The datasets, as provided through the GeoNode backend, and accessible from the green *Download* button, can be downloaded in many well known formats. For vector data that are (see fig. 10.3, C):

- Zipped Shapefile,
- GML 2.0 and 3.1.1,
- KML,
- GeoJSON,
- CSV or Excel format,
- and rendered pictures in JPEG, PNG or PDF format.

Figure 10.2.: List of related resources in the information panel of the maps detail view. Source: Screenshot.

The vector data is provided by the GeoServer WFS endpoint, as so called download service implementation. The rendered bitmaps (JPEG, PNG, etc.) are provided through the WMS endpoint.

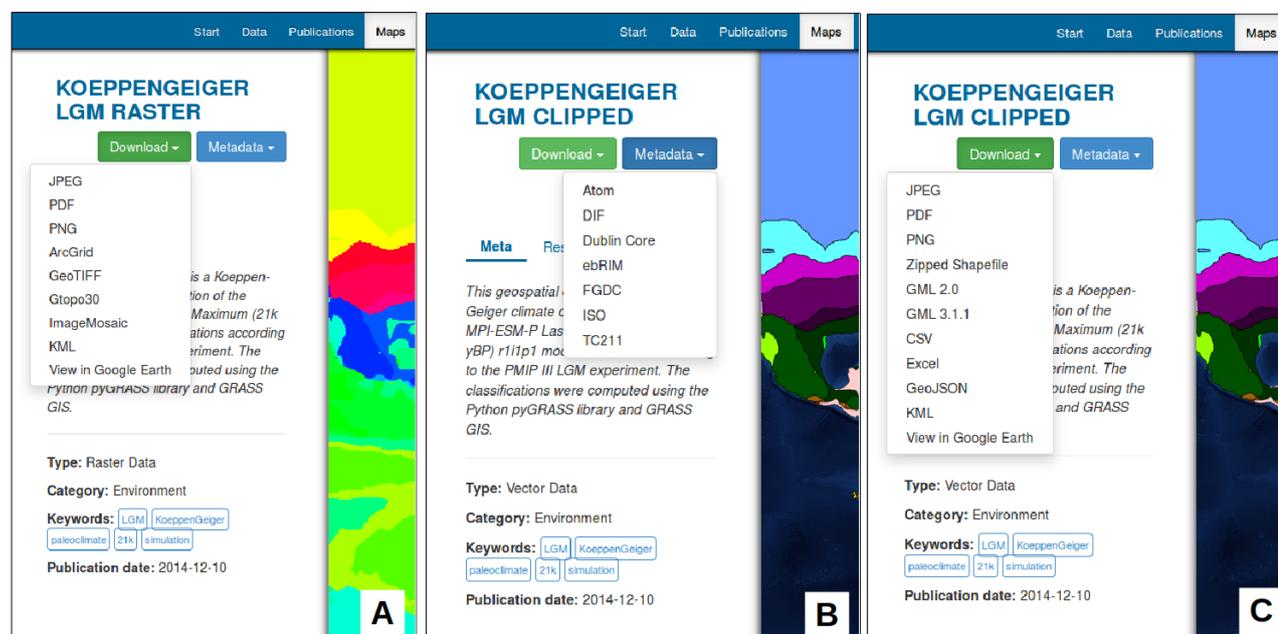


Figure 10.3.: Detail views of the maps datasets detail page information panel. Source: Screenshot.

For raster data the formats are (see fig. 10.3, A):

- ArcGrid,
- GeoTIFF,
- Gtopo30,
- ImageMosaic,
- KML,
- and rendered JPEG, PNG and PDF versions.

Raw raster geodata is provided through GeoServer's WCS endpoint, the rendered bitmaps by GeoServer's WMS. The metadata is available for download in XML format, according to the following schemas (see fig. 10.3, B):

- Atom,
- DIF,
- Dublin Core,
- ebRIM,
- FGDC,

- ISO,
- and TC211.

The data and metadata offered in this interface is served from the GeoNode based SDI backend by according OWS requests. For example if a geodataset is requested in Shapefile format an according download request to the underlying GeoServer WFS is triggered. In the same way, if a resource is requested in GeoTIFF format, an according request to the GeoServer WCS is triggered. If metadata is requested, the CSW endpoint of the GeoNode instance is accessed.

Related resources

The related resources feature is also implemented in the Maps application. It allows to relate spatial datasets with datasets of the data catalog and publications DB repositories. In figure 10.2, a screenshot of the related resources annotations of an example spatial datasets is shown.

From the Maps application, it is not possible to create a relation between resources and datasets, only the relation with spatial datasets that were created in either the data catalog or the publications DB, are shown here.

OpenLayers web map

The interactive web map of the geodataset detail view, was implemented using the OpenLayers framework (OpenLayers Contributors 2015), see section 7.3.2 for a description of the technical implementation.

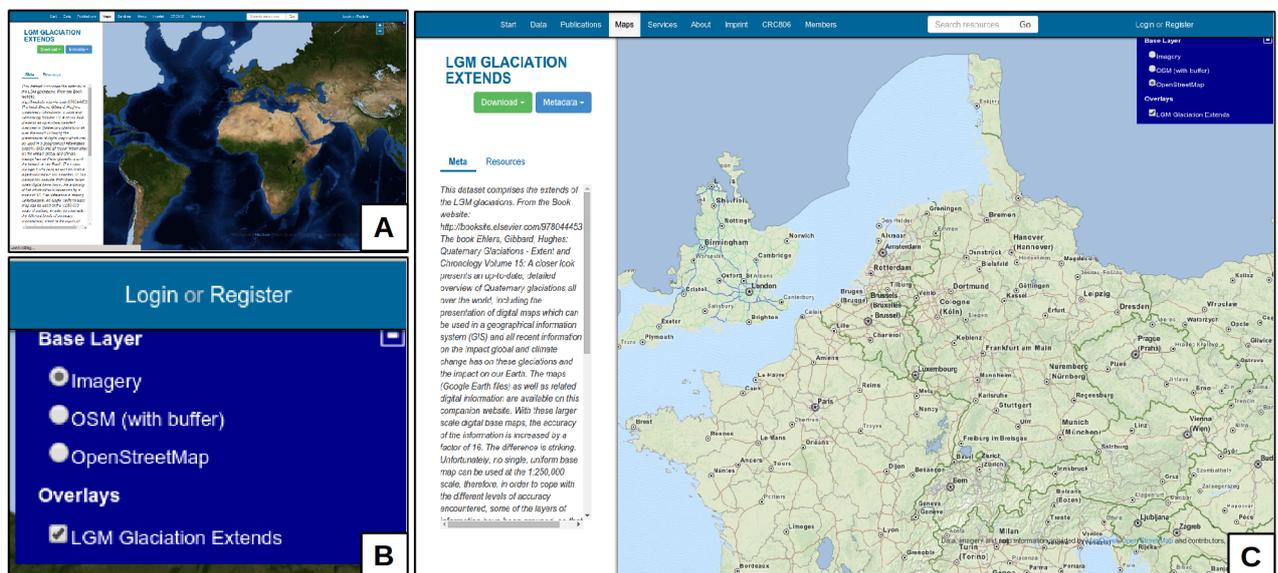


Figure 10.4.: The OpenLayers based WebGIS interactive map. Source: Screenshot.

The map view allows to examine the dataset interactively, using zoom and pan functionality. Geospatial data is rendered in Web Mercator Projection (EPSG:3758 or EPSG:900913), also

known as Google Projection, that is the standard for interactive web maps nowadays. The main reason this projection got so widely accepted, is the possibility of *tiling* (and thus caching) this projection most efficiently. *Tiling* a web map is understood as saving the rendered map as small *tiled* pictures. The web Mercator projection, projects the World onto a quadratic surface, which allows to tile the map, and each defined zoom level (fixed scale) into exactly 4 pieces of the next higher scale. Zoom level 0 is the whole world projected onto one quadratic, the further zoom (or scale) levels are then always a fourth of the previous zoom level extent but in a four times higher resolution. This has the advantage of a clean caching, that can be implemented in a pyramid of tiles according to the fixed zoom levels. See Battersby et al. (2014) for a detailed discussion of this topic.

The default view (see figure 10.4 screenshot A), is centered on the current layers center, and the extent of the map (zoom) is in accordance with the extent of the layer. If a world covering dataset is shown (like it is the case in 10.4, screenshot A), the map is centered on the 0° meridian.

The OpenLayers based web map interface allows to display the data on three different base maps, that can be selected via the *Layer Switcher* UI (see fig. 10.4, B). The OSM based base map is shown in figure 10.4, C.

10.2. OWS Interfaces

This section gives an overview of the different CRC806-SDI OWS endpoints and services.

The SDI is accessible through one major federating endpoint GeoNode. GeoNode then offers several possibilities to access the provided services and data. First, the metadata information is accessible through the GeoNode REST interface (see section 7.1), and additionally through the GeoNode CSW endpoint, that is provided through pycsw, as further described in section 10.2.1.

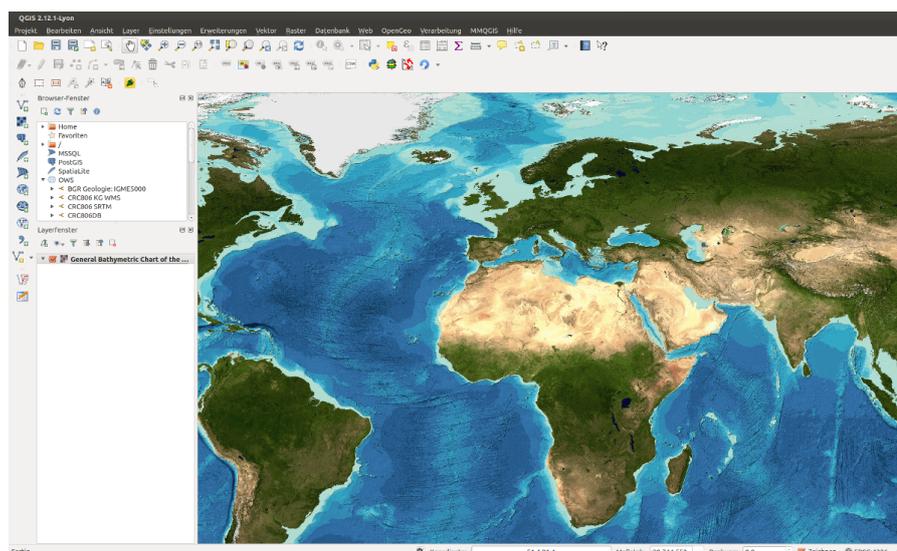


Figure 10.5.: The MapProxy cached GEBCO WMS displayed in QGIS Desktop. Source: Screenshot.

The main purpose for offering geospatial data as OWS is interoperability. And also, the possibility for researcher to include the geospatial services in their desktop applications for analysis and visualization is meant to be a valuable service.

10.2.1. Catalog Service Web

As mentioned, the CRC806-SDI's CSW endpoint is facilitated through the GeoNode's instance pycsw component. A CSW endpoint offers the ability to access metadata about the OWS endpoints and their underlying data of an SDI. The CRC806-SDI CSW endpoint is accessible from:

<http://geonode.crc806db.uni-koeln.de/catalogue/csw>

By implementing the OGC CSW standard, the CRC806-SDI is able to offer Infrastructure for Spatial Information in the European Community (INSPIRE) compliant ISO 19115 metadata (ISO19115-1 2014) of its geospatial services and datasets. Additionally, pycsw is an official OGC reference implementation of the CSW standard (Nebert et al. 2007). An additional feature of the CSW interface are its service *harvesting* capabilities. This allows external federating and integrating open data repositories to automatically index the geodata services provided by the CRC806-SDI and offer them through their interface and in their applications. This can help to increase the visibility of the CRC 806 and its research.

10.2.2. MapServer Services

MapServer is deployed in particular to serve large GIS Datasets as OWS. In the context of the CRC806-SDI, a GIS dataset is defined as large, if the raw data size for the underlying GIS data of a layer is > 100 MB. A second purpose for the use of MapServer is the WCS implementation, that works well, in terms of compatibility and performance with QGIS and ArcGIS clients.

Title	Endpoint	Description
SRTM WCS	http://sfb806srv.uni-koeln.de/srtm?	SRTM WCS of the CRC 806 area.
Sealevel WMS & WFS	http://sfb806srv.uni-koeln.de/sealevel?	Shapefiles of the 10m interval coastlines, between 0m and -150m, of the CRC 806 area.
Stage3 Simulations WMS	http://sfb806srv.uni-koeln.de/sealevel?	The Stage3 paleoenvironment simulations as WMS.

10.2.3. MapProxy Services and Frontend

For accelerating and proxy-ing WMS, a MapProxy instance was setup. MapProxy is mainly deployed to cache external WMS. This approach is chosen for two reasons, i.) if the external WMS has for example technical issues, the service will not be interrupted in the CRC806-SDI, because it can serve the data from cache, and ii.) for performance increase. The caching of map tiles

accelerates an WMS considerably. The proxy-ing approach also decreases the performance load for external services providers for services offered through the CRC806-SDI. Technical setup of the MapProxy instance was described in detail in section 7.2.2. The MapProxy services can be accessed via, the MapProxy web interface too (see figure 10.6).

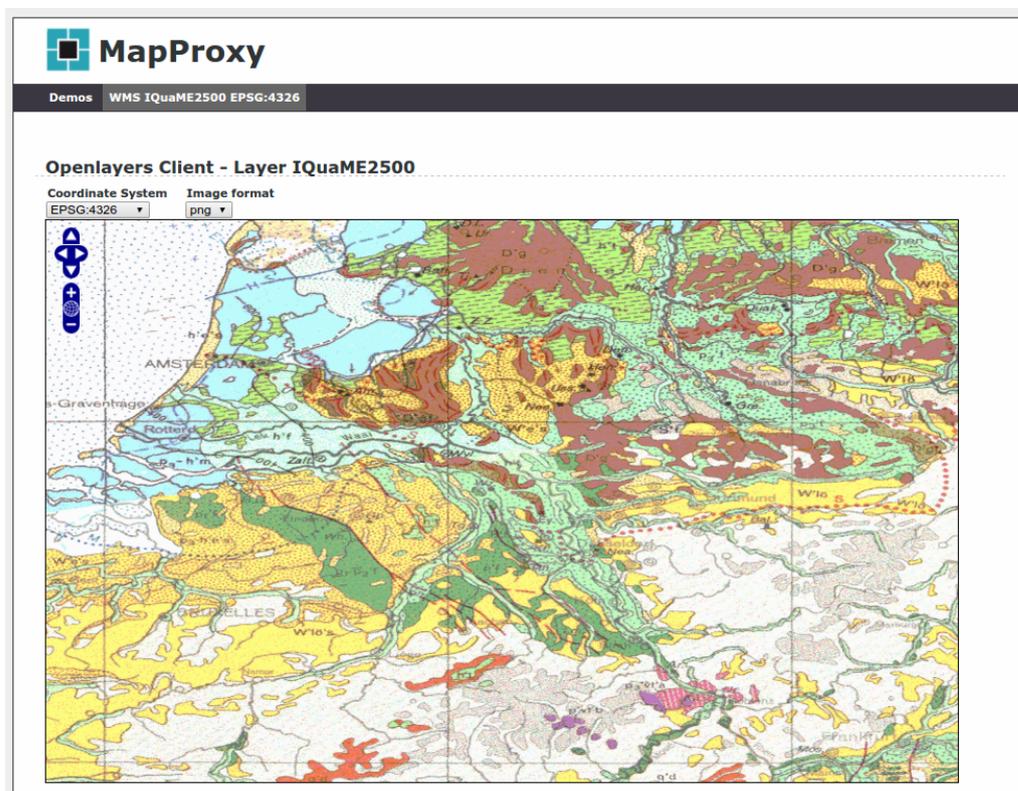


Figure 10.6.: MapProxy web interface. Source: Screenshot.

The MapProxy web interface is accessible from the following URL:

<http://geonode.crc806db.uni-koeln.de:8081/>

The following services are available from the CRC806-SDI MapProxy instance:

Title	Endpoint	Description
GEBCO	http://geonode.crc806db.uni-koeln.de:8081/GEBCO	General bathymetric chart of the oceans WMS.
IGME5000	http://geonode.crc806db.uni-koeln.de:8081/IGME5000	Internationale Geologische Karte von Europa und angrenzende Gebiete 1:5.000.000.
IQuaME2500	http://geonode.crc806db.uni-koeln.de:8081/IQuaME2500	Internationale Quartärkarte von Europa 1 : 2 500 000.

10.3. Backend and data management

GeoNode functions as a CMS for the CRC806-SDI. The GeoNode web application offers a comprehensive UI for uploading, creating and managing geospatial web services, as described in detail in section 7.1.2. In figure 10.7 screenshots of the GeoNode web applications UI are shown.

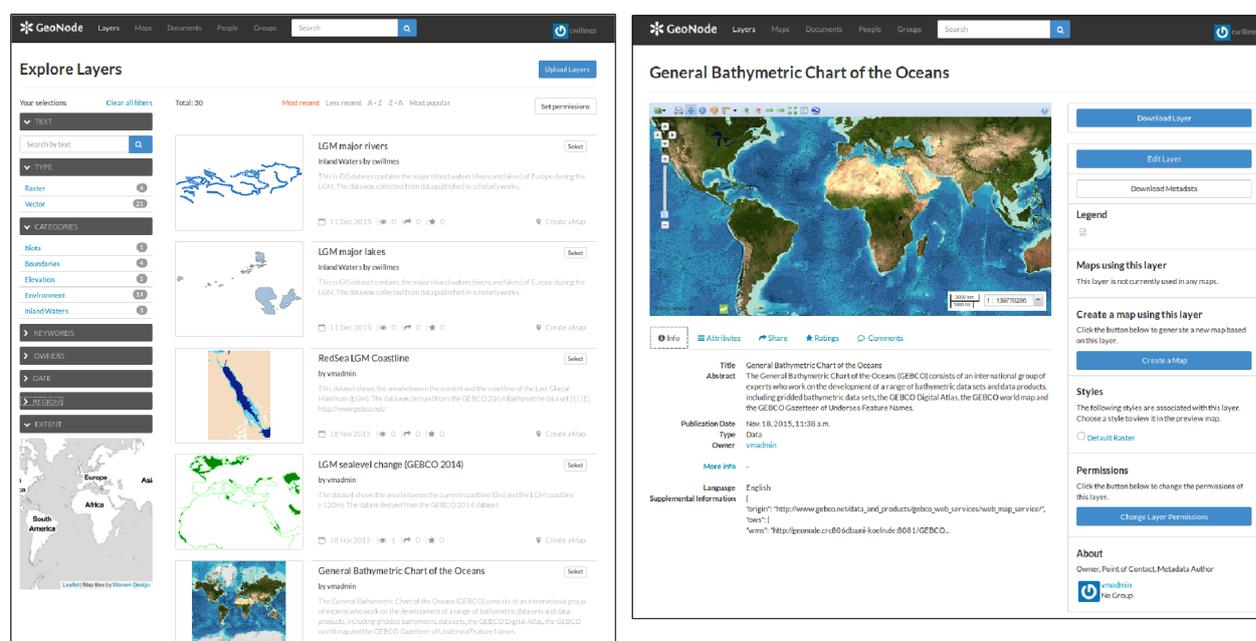


Figure 10.7.: GeoNode web application. Dataset list on the left, detail dataset view on the right. Source: Screenshot.

As the framework for displaying the interactive maps, GeoNode uses the Open GeoExplorer framework (Boundless Inc. 2015). That is an UI to access OWS based spatial data services, and build WebGIS applications using the GeoExt framework (GeoExt Contributors 2014), that integrates the ExtJS library with the OpenLayers (OpenLayers Contributors 2015) library.

This web based UI allows the creation and management of an SDI, also for people without elaborate technical skills, that are normally necessary to configure a MapServer or a stand alone GeoServer instance without the web interface. This allows a stream-lined and integrated geospatial data service production process, as previously described in more detail in section 7.1.2.

GeoNode even offers to manage documents, like Portable Document Format (PDF)s or Spreadsheets, that can be annotated with metadata and can be related to the geospatial datasets of the GeoNode repository. But for the CRC806-Database it was decided to not use this feature, but to implement to integrate with the data catalog and publications repositories of the CRC806-Database, as described in section 6.1.7.

Another feature of GeoNode, that is not (yet) facilitated by the CRC806-SDI, is the Map feature, that allows to create and publish maps. The Map application offers a WebGIS based UI to select and combine layers of the GeoNode repository to create an interactive map consisting of several

layers. The maps are then published in a similar way as the single data layers, in a catalog style repository.

11. CRC806-KB

For the management, organization and collaborative editing of data and information concerning the research of the CRC 806, the CRC806-Knowledgebase (CRC806-KB) application was developed. As described in chapter 8, the CRC806-KB application is based on MW and SMW technology and offers customized and comprehensive UIs. The web application is publicly accessible for viewing and browsing the data, information and knowledge, handled by the CRC806-KB. Even the export of metadata is possible for external users without login, using the ASK API or the *Semantic Search* interface, but editing is only allowed for registered users of the wiki. Any member of the CRC 806 is eligible for an account, and can contribute to the collaborative KB.

To build and maintain collections, consisting of information about data, and information about information, or knowledge (see section 2.1.3), an infrastructure solution was requested within the CRC 806 project. On the basis of SMW a collaborative knowledgebase was developed (see chapter 8). Considering the demands by the DFG, as described in section 3.1.2, the CRC806-KB implements the demand for collaborative research environments, that are suggested and wanted, but not demanded by the funder.

The result of this research was the main wiki of the CRC806-Knowledge base, as described in section 11.1. Besides integrating all kinds of CRC 806 knowledge into one central integrated KB application, it appeared to be more functional to build several KB for specific domains, as described in section 11.2. For the combination of archaeological records and paleoenvironmental, as well as geoscientific data, the *Contextual Areas* KB was created, as described in section 11.2.2. And for the collection of data about ostracodes and foraminifera (Horne et al. 2012) in combination with according bibliographic information the Afriki KB was developed, as described in section 11.2.1.

The basic wiki web interface and its appearance is well known from the standard MediaWiki (MW) appearance (see screenshots for example in figures 8.3, 8.4 or 8.8), as it is prominently known from the Wikipedia web site, that is based on the MW software too.

11.1. Paleoenvironment GIS data collection

Within the GIS data management group of the CRC 806 project Z2, an internal solution for indexing available data sources and according bibliography was demanded, to build a comprehensive knowledge base for supporting GIS applications concerning the CRC 806 research questions. Some example applications of this approach are described in section 12.5.4.

The GIS data collection, under the working title "*Paleomaps*", a sub-system of the CRC806-

Knowledgebase, is currently accessible from the following URL:

<http://www.sfb806db.uni-koeln.de/wiki/>

The main purpose of this application is to collect paleoenvironmental data sources, that can be used as input data for GIS analyses and map creation in the context of the CRC 806. A second purpose of the application, is to foster collaboration and enable synergy effects within the CRC 806 project. This is reached, by the possibility to share not yet published data and knowledge with project partners, that allows to identify possible intersections in research, applied methods, and so forth. Some first research results, that were facilitated by the KB application, are discussed in section 12.5.4. Much potential for more applications in that, and also additional research directions are possible with a growing, functioning and accepted CRC806-KB system.

11.1.1. Data and knowledge domain

A collection of available data sources, in form of datasets and resources in an academic context, naturally needs to handle bibliographic informations. Because the CRC 806 core research questions are of spatio-temporal nature, see section 1.2, a model to handle spatial as well as temporal information needed to be implemented. This is enhanced with annotations of topic categories, as described in section 8.1.

Of course a model for storing CRC806-Database resources in the SMW based KB was implemented, to integrate them as well. Because a CRC806-Database dataset basically is not different from any other dataset, this is handled by a simple annotation, called *SourceType*, which defines if the dataset is *Internal* (not published), *External* (published), or *CRC806-Database* (internal but published through the CRC806-Database).

Paleoenvironmental data, like climate simulations, glaciation extents, landmasks for previous sea levels or biome, pollen and other climate proxy datasets from geoarchaeological records are stored as geospatial datasets in the CRC806-SDI. For those geospatial datasets, a metadata model for describing content, extent and spatial representation type is implemented in the SMW based data model as well (see section 8.1).

The datasets are annotated with spatial regions or sites and temporal periods or events, along which can be queried within the SMW based KB. For each dataset, the URLs to their according OWS endpoints, as well the links to the according web based visualization and download page within the CRC806-SDI *Maps* frontend are annotated.

11.1.2. Data sources

The four basic types of data sources, that are most commonly collected in the CRC806-KB, apart from bibliographic data, are described in the following of this sub-section. Until now, the main use case for the CRC806-KB is to collect GIS data sources, that can be reused for the compilation of GIS based paleoenvironmental maps or for GIS analysis. Figure 11.1 shows a basic catego-

rization of the input data, along two axes of structured or unstructured and intrinsic or annotated spatial information of the datource.

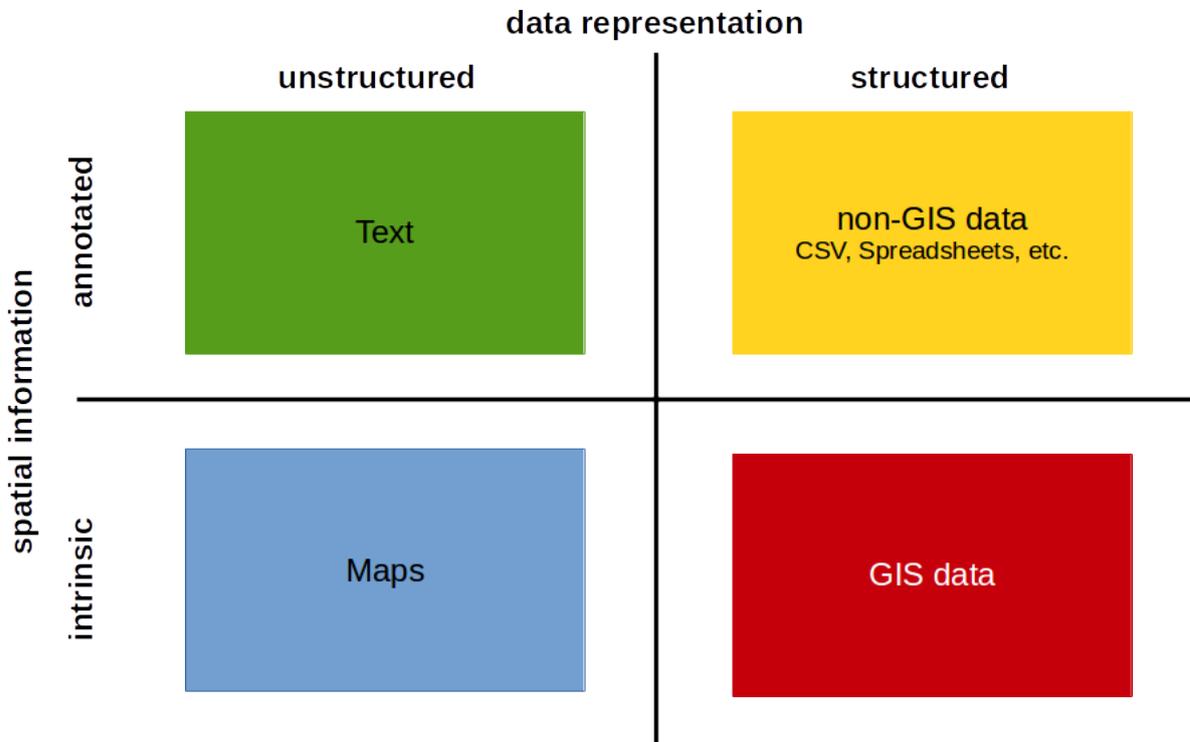


Figure 11.1.: Types of geodata sources. Source: Own work.

Unstructured spatial information

Here the term "unstructured" refers to all data sources, that are not given in a computer process able form, in the sense of a spreadsheet, GIS data, an image of produced by a sensor, or some form of data, that can be used as an input for further computational and algorithmic processing.

Published maps Maps published as prints on paper or in a PDF as an image in non-GIS data format, need to be digitized, into spatial features, by georeferencing and eventually also digitizing them.

Alphanumeric Information Spatial features or locations described in floating text, for example in a scholarly paper, can be modeled in a GIS, according to the given alphanumeric information as, as spatial features. An example would be a glaciation extent, that is described as an area between well known spatial features, e.g. "the maximum extend of the ice shield was marked by the line between town X and lake Y".

Structured spatial data

In the case of digital data, the re-use ability is much more straight forward. Even if it is not available in GIS format, if a spatial context is somehow present in the data, it is possible to transform these data into GIS formats for further use in the presented context.

non-GIS data As non-GIS data we understand, data given as for example as CSV or spreadsheet, that contain an informal spatial reference, which allows to create a GIS dataset from the given information, mostly by formulating a script to read in the CSV and spread sheet and write it out after some scripting foo as a proper GIS dataset.

GIS data As GIS data sources, we understand all kinds of data sources, that can be directly read and displayed in a desktop GIS like QGIS (QGIS Development Team 2015).

For example, topography data is available in abundance for present day spatial context. For paleoenvironments this is much more scarce of course. Because it almost always has to be derived a certain model, for representing the paleotopography. An example would be the coastline representing the sea level during the LGM, which was 120 meters below the level of today.

Bibliography

As described in section 8.1, the bibliographic model of the integrated database is mostly inspired from the BibTeX (Patashnik 1988) model for handling bibliographic resources in the LaTeX environment. The BibTeX model was mainly chosen, because we wanted to stick to an existing implementation for bibliographic data management. This brings the advantage to offer interfaces to a well known format, for data import and export. SMW offers also a BibTeX export format, via the SRF extension. This allows to offer BibTeX export via Semantic Search and ASK API queries. This makes it possible, to automatically export all bibliographic resources annotated with a certain topic, temporal and spatial property. A search for all publications, concerning the Zaffaraya site, during the LGM and sedimentological analyses, can be yielded as a BibTeX file.

11.1.3. Example knowledge item

A good example to illustrate, what the KB application is about, is the knowledge item page about the "High resolution Köppen-Geiger classifications of paleoclimate simulations" publication (Willmes et al. 2016). Figure 11.2 shows the bibliographic record page of this article, as rendered in the KB application.

A bibliographic record is handled as a wiki page. The title of the page, in this case "Willmes2016" is the reference, consisting of first author surname and year of publication of the bibliographic record. This has the advantage, that the bibliographic item can be linked (or referenced) by its actual bibliographic reference (in this case also the page title, that is also the link to a wiki page).

Willmes2016

- BibType:** article
- Title:** High resolution Köppen-Geiger classifications of paleoclimate simulations
- Author:** Christian Willmes, Daniel Becker, Sebastian Brooks, Christoph Hilt, Georg Bareith
- Year:** 2016
- Journal:** Transactions in GIS
- URL:** http://www.sfb806db.uni-koeln.de/wiki/images/2/20/Willmes_KG_TGIS.pdf#
- Topic:** Paleoclimate; Köppen-Geiger
- Temporal:** LGM; Mid-Holocene; Pre-Industrial
- Spatial:** World

Referencing Datasets:

	Source	Abstract	Temporal	Spatial	Topic	DatasetURL	Resources	Bibliography
Köppen-Geiger classification of MPI-ESM-P LGM simulation	CRC806-Database	This geospatial dataset, in raster and vector format, is a Köppen-Geiger climate classification of the MPI-ESM-P Last Glacial Maximum (21k yBP) r11p1 model simulations according to the PMP III 21k experiment. The classifications were computed using the Python pyGRASS library and GRASS GIS.	LGM	World	Paleoclimate Köppen-Geiger	http://crc806db.uni-koeln.de/datasets/show/koppen-geiger-classification-of-mpi-esm-p-lgm-simulation/	LGM Köppen-Geiger Shapefile.zip MPI-ESM-P LGM r11p1 Köppen-Geiger classification.tif	Willmes2016 Willmes2014a
Köppen-Geiger classification of MPI-ESM-P Mid-Holocene simulation	CRC806-Database	This geospatial dataset, in raster and vector format, is a Köppen-Geiger climate classification of the MPI-ESM-P Mid-Holocene (6k yBP) r11p1 model simulations according to the PMP III 21k experiment. The classifications were computed using the Python pyGRASS library and GRASS GIS.	Mid-Holocene	World	Paleoclimate Köppen-Geiger	http://crc806db.uni-koeln.de/datasets/show/koppen-geiger-classification-of-mpi-esm-p-mid-holocene-simulation/	MPI-ESM-P Mid-Holocene r11p1 Köppen-Geiger classification.tif MPI-ESM-P Mid-Holo Shapefile.zip	Willmes2016 Willmes2014b
Köppen-Geiger classification of MPI-ESM-P Pre-Industrial simulation	CRC806-Database	This geospatial dataset, in raster and vector format, is a Köppen-Geiger climate classification of the MPI-ESM-P Preindustrial r11p1 model simulations according to the PMP III 21k experiment. The classifications were computed using the Python pyGRASS library and GRASS GIS.	Pre-Industrial	World	Paleoclimate Köppen-Geiger	http://crc806db.uni-koeln.de/datasets/show/koppen-geiger-classification-of-mpi-esm-p-pre-industrial-simulation/	MPI-ESM-P Preindustrial r11p1 Köppen-Geiger classification.tif MPI-ESM-P Preindustrial Shapefile.zip	Willmes2016 Willmes2014c

Referencing Resources: [edit]

	ResourceLocation	ResourceType	Format	Abstract	Topic	Bibliography
LGM Köppen-Geiger Shapefile.zip	http://crc806db.uni-koeln.de/datasets/getResource?x_unikoelnrcrdata_crcrdata[id]=078cd3ab-90d5-4411-a164-23232e256c6e&x_unikoelnrcrdata_crcrdata[typ]=806&Hash=4b2482661cc1ee63a0c7dc4b02be182j	GIS Data	Shapefile		Paleoclimate	Willmes2014a Willmes2016
MPI-ESM-P LGM r11p1 Köppen-Geiger classification.tif	http://crc806db.uni-koeln.de/datasets/getResource?x_unikoelnrcrdata_crcrdata[id]=4c08f6c3-6505-452e-989e-723cd76d507e&x_unikoelnrcrdata_crcrdata[typ]=806&Hash=05a912b2c93018e1d9a9f9d330cb491j	GIS Data	GeoTIFF		Paleoclimate	Willmes2016
MPI-ESM-P MidHolo Shapefile.zip	http://crc806db.uni-koeln.de/datasets/getResource?x_unikoelnrcrdata_crcrdata[id]=a440b328-2a86-4031-998c-1bd6f5a1802&x_unikoelnrcrdata_crcrdata[typ]=806&Hash=ac8d2ada81ec1154a8d00ad176c52j	GIS Data	Shapefile		Paleoclimate	Willmes2014b
MPI-ESM-P MidHolocene r11p1 Köppen-Geiger classification.tif	http://crc806db.uni-koeln.de/datasets/getResource?x_unikoelnrcrdata_crcrdata[id]=e830374e-2b62-4a8f-af42-70507d7380da&x_unikoelnrcrdata_crcrdata[typ]=806&Hash=05d9f504e37a2aeed2f3c2604e7a78j	GIS Data	GeoTIFF		Paleoclimate	Willmes2014b
MPI-ESM-P Preindustrial Shapefile.zip	http://crc806db.uni-koeln.de/datasets/getResource?x_unikoelnrcrdata_crcrdata[id]=be612105-a648-4aa4-836d-a981639a9e80&x_unikoelnrcrdata_crcrdata[typ]=806&Hash=f41817cb647dc3396c52e2da33c3a1e1j	GIS Data	Shapefile		Paleoclimate	Willmes2014c
MPI-ESM-P Preindustrial r11p1 Köppen-Geiger classification.tif	http://crc806db.uni-koeln.de/datasets/getResource?x_unikoelnrcrdata_crcrdata[id]=2c08f9d-cf3a-4e56-bb16-5fa7c08780f8&x_unikoelnrcrdata_crcrdata[typ]=806&Hash=9b9f9a3a3b1c0dc706a6155d006c1j	GIS Data	GeoTIFF		Paleoclimate	Willmes2014c
MPI-ESM-P r11p1 20120602 LGM pr	Z:/Data/KoepenGeiger/	GIS Data	NetCDF	LGM Precipitation Simulation	Climate Simulation Paleoenvironment Precipitation	Willmes2016
MPI-ESM-P r11p1 20120602 LGM tas	Z:/Data/KoepenGeiger/	GIS Data	NetCDF	LGM Temperature Simulation	Climate Simulation Temperature	Willmes2016

Figure 11.2.: Wiki page of a Bibliographic record with annotated resources. Source: Screenshot.

The SMW template for bibliographic records has two queries, one for *Datasets*, that are annotated with the given bibliographic record, and another query for *Resources*, that are annotated with this bibliographic reference.

In this example, the three GIS datasets of the Köppen-Geiger classifications of paleoclimate simulations, that are the results of this publication, are annotated to the bibliographic record. The user can now follow the links to the according dataset pages, and will find links to the datasets for direct access of the data.

Furthermore, the referenced GIS datasets consist of two resources each. A vector data representation and a raster data representation of the GIS dataset. These resources are also listed under the *Resources* query, and can be accessed accordingly. Additionally, the source climate simulation Resources are linked to the bibliographic record, because they were also inserted into the wiki and related with the reference to this bibliographic record.

11.1.4. Categories

Currently, as described in section 8.1.1 the wiki organizes its data in the following categories:

Dataset Datasets are the main knowledge items, they are annotated with resources, temporals, spatials and bibliography. CRC806-RDM datasets can be mapped 1:1 to the CRC806-KB dataset, and are in this way also available from the KB system.

Resource Hold single data files or links, that are part of a dataset. The concept is similar to the resource concept of the CRC806-RDM system.

Temporal Temporals define events or intervals (time periods), and can be used as annotations of knowledge items.

Spatial Spatials define places (sites) or regions (bounding-box), and can be used as annotations for knowledge items.

Bibliography Bibliography knowledge items hold the bibliographic informations of scholarly publications. The bibliographic records can be used as references (annotations) for datasets, resources, temporals and spatials. Bibliography items can also be linked (annotated) with temporals and spatials, as well as with datasets and resources.

These categories are a first entry point for browsing through the information stored in the KB.

11.2. Domain Knowledgebases

In the development process of the KB system, questions about how to handle the many sub-domains of the CRC 806 in regards to the overall knowledge base came up. Building integrated data models across domains is a complex and difficult task, thus it was decided, to build own sub-domain KB's. This allows to experiment more freely with the data model design and development, without interfering other data collection approaches. For implementing the sub-domain KB, the SMW based concept developed for the main CRC806-Knowledgebase was additionally applied to two sub-domain KB's, as described in the following of this section.

11.2.1. Afriki

The Afriki wiki was developed to assemble primary data of already published results from East Africa (Nile valley - African Rift valley) in the Late and Middle Pleistocene (0.012 to 0.78 Ma OR MIS 19). Since several decades archaeologists and geoscientists explore suitable sites to answer related questions. Simultaneously, analytical methods applied to the archives improved in their sensibility or resolution over the given time. The amount of published scientific data is enormous, but has to be carefully checked on their robustness compared to modern standards. Therefore, it is necessary to compile datasets and excerpt the given data that are source of scientific interpretations (e.g., palaeoenvironment, palaeoclimate, evolution patterns, time models etc.). In addition, the names of the study sites in East Africa are often transcribed from different languages or hold several synonyms for various reasons. Thus, we decided to use a semantic wiki to have the advantage of query-able structured data combined with the ability of web based frontend for collaborative editing of the content (Viehberg et al. 2015).

The Afriki KB is accessible from the following URL:

<http://www.sfb806db.uni-koeln.de/afriki/>

Details of published and unpublished archaeological and geological sites/localities in East Africa are collected in the presented wiki including their bibliographical reference. For example from sediment records, results from available sedimentological/chemical/biological proxy data (e.g., grain size, total organic carbon, stable isotopes, diatoms, ostracods, magnetic susceptibility) are copied into the database including their spatial resolution. Related dating samples (i.e., ^{14}C , OSL, TSL) are also included with their metadata and lab-codes (Viehberg et al. 2015).

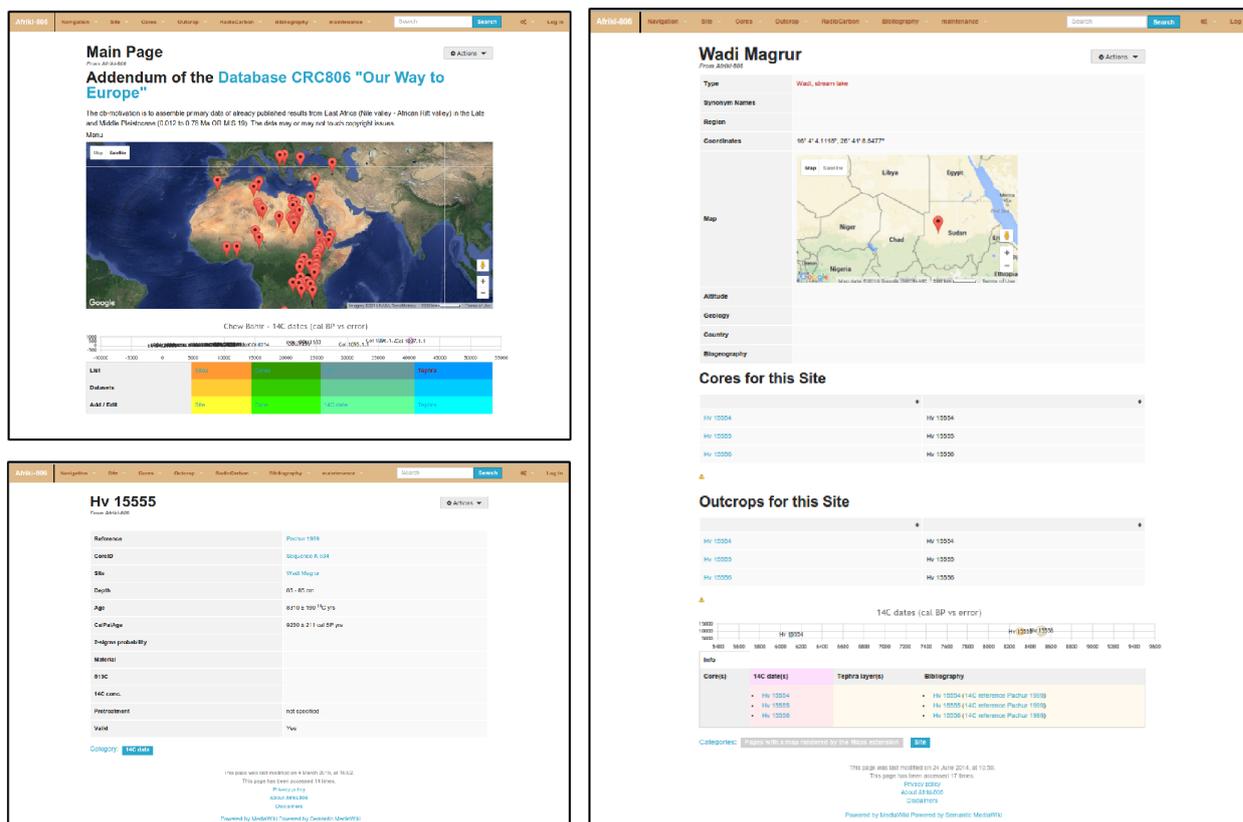


Figure 11.3.: UI of the Afriki KB. Source: Screenshots.

Technically, this wiki instance was enhanced with a custom theme (i.e. layout), as shown in figure 11.3. The theme was developed and customized by the project partners of the A2 project under the lead of Dr. Finn Viehberg. The data collection in this KB instance, was also completely carried out by the A2 project. The assistance by Z2, was in providing the wiki infrastructure and help in developing the data model and data queries including according visualizations. The data model of the application consists of 8 classes:

- Bibliography,
- Site,
- ^{14}C Data,
- Core,
- FaunTaxon,
- OstracodTaxon.

- Outcrop,
- and TephraData.

A feature that makes this KB instance interesting, is the application of temporal visualizations (see figure 11.3), that allows to visualize dates on an interactive zoom-able timeline. Map visualizations of spatial annotated data was also implemented for this wiki.

11.2.2. Contextual Areas

The *Contextual Areas* wiki, was developed for project partners of the B and C clusters, to gather spatio-temporal archaeological informations in one data base, to identify so called *Contextual Areas* in time and space.

The *Contextual Areas* wiki is accessible from the following URL:

<http://www.sfb806db.uni-koeln.de/context/>

For this KB a data integration approach, as technically described in section 8.3, was developed and applied. Thus, for each of the datasets, a custom Python script, that generates DataTransfer XML was implemented. The XML was then imported into the wiki, using the DataTransfer extension. See table 11.1 for an overview of the integrated archaeological datasets.

Table 11.1.: Integrated published archeological databases.

Database	# Datings	# Sites	# Properties	Data format	Source
INQUA DB	21500	7238	54	MS Access	(Vermeersch 2011)
PACEA	6021	1209	26	CSV & MS Excel	(d'Errico et al. 2011)
Stage3 Arch	1896	380	20	MS Excel	(Andel et al. 2003)
Stage3 Faun	1912	502	24	MS Excel	(Andel et al. 2003)
CONTEXT	2874	441	31	MS Excel	(Böhner et al. 2006)

The model was developed in a prototyping approach (see section 4.5), by extending and adapting the data model from the properties of each of the imported datasets, as listed in table 11.1. The data model of the Contextual Area KB consists of eight classes:

- Artefact,
- Bibliography,
- Dataset,
- DatedAge,
- Layer,
- Region,
- Site,
- and TimePeriod.

11.3. Integration

A next step in the development of the CRC806-KB is the integration of knowledge from the shown KB instances into a single knowledgebase.

As can be seen from the three presented KB instances, substantial semantic overlap of the individual data models is given. This allows to integrate these KB instances to a certain degree.

The main properties for integration are:

- Space,
- Time,
- Topic,
- and Bibliography.

Along these axes, an integrated KB interface can be implemented. In a first step, the data models of the two domain knowledgebases, Afriki (see section 11.2.1), and Context (see section 11.2.2), would have to be transferred to mobo. Then, the combination and integration of the three models into one KB is feasible.

Part V.

Synthesis

12. Discussion

In the presented work, the design and implementation of a semantic e-Science infrastructure for an interdisciplinary research project was conducted. This chapter is about discussing the design and implementation decisions that were taken and realized.

Implementing a comprehensive e-Science infrastructure and web application like the CRC806-Database is an ambitious task. Many choices need to be taken and then realized. There are decisions between developing a system from scratch in house or integrating existing implementations, that solve sub problems of the proposed system, between applying Open Source or proprietary software solution, or a combination of both. Additionally, we face constraints on resources (infrastructure, workload, funds, etc.). The here presented implementation is the result of a process of finding answers and a solutions for these questions, decisions, demands and constraints.

In the following of this section, we start with an overall more technical discussion of the CRC806-Database semantic e-Science infrastructure (see section 12.1). The demands, that were defined in section 3.1 are evaluated and discussed in section 12.2. A discussion of the applied licensing and data policies is conducted in section 12.3. The implementation of the Repository Lifecycle, as introduced in section 3.2.3, including a review of the 2014 major update of the CRC806-Database (see section 12.4.1), and the Research data lifecycles, as introduced in section 3.2.2, are discussed in section 12.4. How the CRC806-Database implements Open Science, Open Access and Open Data principles is discussed in section 12.5.

This is followed by a look at the actual data, that is stored and managed by the CRC806-Database at the time of writing this thesis, in section 12.6.

12.1. The CRC806-Database semantic e-Science infrastructure

The CRC806-Database is a comprehensive system, consisting of a number of applications covering the grounds of RDM, SDI and KB. This section critically discusses the overall system architecture, the design, its features, capabilities, its organization, its purpose and its internal integration. This discussion will explain advantages and disadvantages of the components and its technology, and then explain how and on what grounds the decision to use a certain technology was chosen.

For implementing the whole infrastructure, existing open source software applications were adapted (see section 5.2). CKAN as backend for the data catalog metadata management. GeoNode as the backend for the SDI, and Semantic MediaWiki as framework for the KB. And the

integrated frontend is based on the open source CMS Typo3.

By implementing the concepts of semantic e-Science, the CRC806-Database aims to contribute to the prevalent developments in science and society, that are triggered by the computational revolution and transition into a knowledge based economy and society, known as the information revolution (Hey et al. 2005), and as defined and projected by the definition of the fourth paradigm (Hey et al. 2009). The CRC806-Database semantic e-Science infrastructure is one additional answer on how to support scientific research through infrastructure and technology in times of the information revolution.

Here the term *semantic* shall be emphasized, because now that syntax-only (metadata) interoperability is not anymore state of the art (Fox et al. 2009), SWT is the logical next step, and in few infrastructures already state of the art.

The implementation of SWT within the CRC806-Database is by far not complete and sufficient. It will be further enhanced and developed in the future to meet the increasing demands for true semantic interoperability and integration.

12.1.1.1. Architecture, Structure and Technology

As outlined in section 3.3, during the development and implementation of the CRC806-Database semantic e-Science infrastructure, three main building blocks emerged, that were build mostly independent from each other. The CRC806-Database is the name for the overall infrastructure system, and it consists of three major components:

- CRC806-RDM
- CRC806-SDI
- CRC806-KB

These components are summarized and discussed in the following of this section.

CRC806-RDM

At the beginning of the CRC 806 and the Z2 project in late 2009 and early 2010 (see section 1.2), there was a discussions on how to implement the web based infrastructure. First idea was to transfer the concept of the TR32DB (Curdt et al. 2009), that were built since 2007, in a similar DFG INF project setting in the frame of the Trans Regio 32¹, also at the GIS & RS working group of the Institute of Geography (Bolten et al. 2010). This plan was withdrawn, because the TR32DB was and still is a custom developed system for the TR 32 project, consisting of a custom developed PHP & MySQL web application, and a custom metadata schema (Curdt 2014a). It was concluded, that the application of an available CMS would have more or less the same trade off's in terms of adapting the system to the custom needs of the CRC 806. Though, the benefits of a CMS that is supported, developed and improved by a developer community from different backgrounds (industry, academia, public sector, etc.) destined the decision for choosing a CMS

¹<http://tr32.de/>, accessed: 2016-02-20.

approach. This initial design was described and discussed in (Willmes et al. 2012a; Willmes et al. 2012b).

The decision for applying Typo3 as CMS, was mainly influenced by the fact, that the RRZK offered hosted and managed instances of Typo3. As explained in section 5.2.1, the RRZK took care of updating and patching the CMS, as well as backup and server availability and security. The CRC806-RDM system consists of the main Typo3 based website of the CRC806-Database including the data catalog (see section 9.2), the according data repository (see section 6.1.1), and the publication DB (see section 9.3). The SDI, that is integrated by its Maps frontend (see section 10.1) into the main website is not directly part of the CRC806-RDM, but of the CRC806-SDI. Though, in this overall context of the main web site, the CRC806-SDI is integrated with the CRC806-RDM, in a user-friendly solution.

To integrate the different applications and backends of the CRC806-Database, as well as to extend the the Typo3 CMS with custom functionalities, such as the members management (see section 9.4) or the News & blog interface (see section 9.1), we started to develop custom extensions. For the development of the extensions the decision of applying *Extbase & Fluid* was taken (see section 5.2.1), because it was the most promising and well thought through technology available for Typo3 at the time (and still is the most advanced Typo3 extension framework). This decision proofed to be successful, because the custom developed extensions are very robust and good to maintain. Code developed for one of the extensions, can be reused in another extension, this feature saved a lot of work and improved the internal integration of the extensions.

CKAN (see section 5.2.2) was chosen as an out of the box open data and research data management application, for implementing the metadata management of the CRC806-Database RDM infrastructure. CKAN is well adapted, especially by the British RDM community (Winn 2013). The application is very robust, and well proven in many large installation, such as `data.gov.uk` or `data.jrc.ec.europa.eu`. Additionally, it is developed by a sustainable non-profit organization, the OKFN.

The technology chosen for the secure file backend, was based on previous good experiences with this technology by members of the working group, especially in the frame of the TR32BD project (Curdt 2014b; Curdt et al. 2015), as well because it was offered by the RRZK at the time. The AFS infrastructure is well established and proved secure and stable since long time at the RRZK (see section 5.1), for an in depth explanation of the AFS implementation at the RRZK.

A relatively weak spot of the current system is the AFS backend, that consists of volumes allowing a maximum file and folder size of 8 GB. In current days this is already sometimes not sufficient, and will certainly be insufficient in the future. The AFS technology itself is quite old, and better solutions exist. Though, the technology proofed to be very robust, and it apparently does its job very well. In this regard, there is no need from our side, to change the file system backend, but before the end of the project funding, the AFS technology will be evaluated, if it is suitable for the black box post-project phase static web application (see sections 3.2.3 and 14.2). It is considered to move the web application and all data into one virtual container after the project. This would not allow to further use the AFS, that is accessed through an external

endpoint as described in section 5.1, but a solution that further uses the AFS is possible too. A solution including the AFS would increase the complexity of the post-project application. Maybe a solution, that lets the AFS storage as is, but uses a copy of the AFS backend from within the virtual container is most suitable, but this has to be evaluated carefully.

CRC806-SDI

For the handling and management of geodata, and providing open standards based spatial data web services, also known as OWS, the CRC806-Database infrastructure maintains a SDI component.

The CRC806-SDI is based on GeoNode technology (GeoNode Contributors 2014) (see section 7.1.1) as main backend for OWS and general spatial data management. GeoNode (see section 5.2.3) was not applied from the start of the CRC806-SDI as its OWS backend component. In the first version of the SDI, as described in Willmes et al. (2014e), it was based on MapServer and MapProxy, that was integrated into an OpenLayers based WebGIS, no repository or CMS style management of the spatial data services was implemented at this time. A more detailed review of this update is also given in section 12.4.1. The integration with QGIS (see section 7.1.2), that allows to upload and edit data layers in the GeoNode system from the QGIS application in a graphical user interface is also a big plus for GeoNode. Since the SDI was switched to GeoNode, a lot of additional features, and also a significant improvement of usability and maintainability of the SDI was achieved.

Integration into the main website is facilitated by a custom developed Typo3 extension (see section 7.3), that interfaces the GeoNode application. This integration onto the main website also facilitates a seamless integration with the datasets and resources of the CRC806-RDM, including the main data catalog, as well as the publications DB.

The MapProxy (Tonhofer et al. 2014)(see section 7.2.2) and MapServer (MapServer Contributors 2014) (see section 7.2.1) instances, that were the backend of the SDI in version one, are still in use, as described in section 5.2.3.

But another, not yet addressed reason for choosing Open Source SDI infrastructure in direct comparison to the marked leader ArcGIS for Server application (esri 2016), that would also be accessible to the Z2 project, is that the deployment of open source does not need renewal of licenses, that would be needed for the esri product (every year). In the light of plan for *black-boxing*, securing and hardening of the system for the post-project phase (see section 14.2), a license renewal would need to be taken care of after the project, which would be questionable to guarantee.

CRC806-KB

Several demands, that could not be directly solved by the CRC806-RDM or CRC806-SDI systems, came up during the first phase of the CRC 806. These demands were for a data base to collect external data sources, that are of interest to the research of the CRC 806. Approaches from the

use of BSCW (RRZK 2016a), or simple cloud based internal file sharing of MS Access and MS Excel data bases, were the state-of-the-art in the CRC 806 project, at this time.

The decision to build the CRC806-KB (see chapter 11) on the basis of the Semantic MediaWiki framework (SemanticMediawiki Contributors 2015) (see section 5.2.7) had several reasons. As described above, an application for collaboratively editing an internal database was requested from some project participants of the CRC 806. The database should hold information about the following topics:

- Bibliography,
- Paleoenvironment (e.g. climate reconstructions and simulations),
- Archaeology (e.g. Sites and dated artefacts),
- Geoscientific information (e.g. sediments and cores).

The main aim was to annotate existing published datasets and papers with spatio-temporal, as well as topic based informations, to facilitate queries along these annotations.

After some initial discussions about what would be helpful and how this VRE could be facilitated, a set of criteria emerged, which set the constraints for the search of sufficient software applications that could be applied. These criteria and demands for the KB application were the following:

- web based application,
- supportive for collaborative work,
- open source,
- active development and maintenance of the codebase,
- easy to setup and maintain (maybe even facilitated by RRZK).

All these criteria and demands were met by SMW, additionally the very supportive and responsive community of SMW played a role to decide for this technology.

A system to generate mainly internal overviews of what is available in form of digital information, and not only published data and publications, within the CRC 806 project was demanded. The ability to, in some sense, crowd source the development of this knowledgebase to the project participants was also needed. Thus the decision for a wiki based approach was taken.

Apparently, the approach of first implementing several smaller KBs of narrow model scope, instead of one big KB that implements all scopes of the CRC 806, is again an implementation of the bottom-up approach.

12.1.2. System Integration

In section 3.4, the design for integrating the components of the CRC806-Database semantic e-Science infrastructure were laid out, which were then implemented and described from a technical point of view in chapters 6, 7 and 8. Although, the three components could each be a stand alone application, they integrate in several ways and on several technological as well as content based levels.

The most apparent integration layer for the three components is the main website of the CRC806-Database, as described in detail above, the two components of the CRC806-RDM and CRC806-SDI are integrated into one Typo3 based web frontend. The CRC806-KB is not yet integrated fully into the main web frontend, but it is integrated in many ways with the two CRC806-RDM and CRC806-SDI components.

To integrate the data of the CRC806-RDM into the CRC806-KB, the metadata of datasets stored in the CRC806-RDM are also stored in the CRC806-KB, to allow queries also on these data. Temporals and Spatial that are originally defined in the CRC806-KB for annotating data, are also available in the CRC806-RDM and CRC806-SDI for annotating datasets (see section 9.2, 10.1). These temporals and spatials, can be automatically imported into the CRC806-RDM and CRC806-SDI systems using the *Temporal* and *Spatial* import functionality, as described in section 9.8. Datasets stored in the CRC806-SDI are integrated into the CRC806-RDM through the related data feature, as described in section 6.1.7. The metadata of the CRC806-SDI datasets, resources and web services are also stored in the CRC806-KB to allow for integrated queries also on the SDI resources.

This shows, that the three major components of the CRC806-Database semantic e-Science infrastructure are well integrated in several ways.

12.1.3. Development methods

For designing and developing the CRC806-Database a set of different methods were applied, as introduced in chapter 4. Apart of being the methodological basis for implementing the presented semantic e-Science infrastructure, the applied methods have in common, that they support an organic, emergence based bottom-up and iterative approach to application engineering and development.

The DDD approach, as introduced in section 4.1.1, applied for development of the Typo3 extensions also embraces the organic bottom-up approach, by developing applications from the domain model up (Rau et al. 2013).

Prototyping as introduced in section 4.5, that is primarily applied for data model development, also emphasizes the iterative bottom-up approach.

The consideration of choosing a bottom-up in contrast to a top-down approach was and is, that there should always be the possibility to develop, extend and enhance the infrastructure further. Implementing top-down approach, can also have certain advantages, because if designed well it will facilitate state-of-the-art technology in a well thought out and designed application.

But top-down is by definition inflexible, because it is well defined beforehand, which of course can be beneficial and preferred in some contexts. In the case of top-down, changes to the path of development are more difficult or even in some cases impossible to implement along the way, also further improvement of the application is limited compared to a bottom-up designed application.

The nature of science and research is mostly bottom-up, and if the development of a semantic e-Science infrastructure shall be regarded as research, it needs to apply the emergence based bottom-up approach, to at least try to find out something new and add to the improvement of the

current state of the art.

12.2. Discussion and evaluation of the demands

In this section, the demands, that were introduced in chapter 3 are evaluated and discussed, with focus on the question if and how the demands were addressed and implemented in the CRC806-Database.

12.2.1. Semantic e-Science Infrastructure

In section 3.1 the demands for e-Science and RDM infrastructures were introduced. It was pointed out, that the demands are manifold and not only of technical but also of social and even political nature.

The technical demands were structured in a list, and they will be answered and discussed, as implemented, point by point here in following that same format.

Data storage The demand for secure long term storage of the research data, as demanded by the DFG, was implemented as addressed in section 6.1.1.

Data exchange Data exchange is facilitated in many ways by the CRC806-Database infrastructure. Through i.) data publication using the CRC806-RDM (chapter 9), ii.) Spatial data services can be published as well as integrated from external publishers via the CRC806-SDI (chapter 10), and iii.) External and internal data can be exchanged via the CRC806-KB (chapter 11).

Data context The data managed by the CRC806-Database is set into context through spatial, temporal and topic annotations, as well as through standardized metadata schemas and vocabularies, as discussed in section 12.1.2.

Data reuse Is also facilitated in many ways by the CRC806-Database. i.) Through the implementation of Open Science, Open Data and Open Access principles. ii.) By the application of open and legally secure licenses for the published resources. iii.) By encouraging to avoid access restriction to published resources. iv.) By providing open interfaces and using well defined formats and standards. This aspect is also discussed in more detail in section 12.5.

Usability Was increased by delivering one combined web interface for the public resources, through the CRC806-Database website. As well as allowing to upload and publish data in a one-stop web form, with no need for additional steps, like data upload through FTP, or application for institutional accounts, etc., see chapter 9.

Additionally some more social, cultural and ethical demands were defined, that are not so easy to answer like the pure technological demands in a list as above.

The first such demand is the facilitation of project internal, as well as external and interdisciplinary collaboration. This was mainly addressed by implementing the CRC806-KB as an VRE

implementation. But this demand is also catered by the CRC806-RDM and CRC806-SDI components, by allowing to publish and share data and research results and ideas through them.

Technologically, were possible SWT was applied (see sections 2.2.4, 6.1.5, and 6.1.6 for example) to ground the interoperability of resources in semantic technology. This approach is also responsible for the prefix *semantic* before *e-Science infrastructure* in the title of this thesis.

Though, the most important demand are the ethical considerations. The in some sense ten commandments for RDM and for conducting research and science in general are the *Proposals for good scientific practice* (DFG 1998), that were extended and renewed as more formal demands for RDM, as *Leitlinien zum Umgang mit Forschungsdaten*, in 2015 by the DFG (DFG 2015). The implementation of these RDM demands are discussed in the next section in detail.

Research Data Management

In section 3.1.1 a list of 21 general demands for successful RDM were given in table 3.1. This table also had a note about how each demand could be implemented. This resulted in 10 implementation components, that are discussed here in table 12.1

1	Code of good research practice	Adapted from and provided by (DFG 1998).
2	Data catalog	As described in chapters 6 and 9
3	Data Governance	For the CRC806-RDM this partly the responsibility of the dataset author, the maintainer and the Z2 project staff.
4	Active data storage	Provided by the RRZK through SoFS (RRZK 2016b) and/or Sciebo (Sync & Share NRW 2016).
5	Data repository	As described in section 6.1.1.
6	Secure data access	Implemented through URM, as described in section 9.4.
7	DOI minting service	As described in section 6.1.4.
8	RDM policy or aspirational statement	Application of funders proposals for good scientific practice and general laws like copyright and the german constitution (freedom of art and science) article 5, paragraph 3. No custom policy was implemented.
9	Open Access policy	As described in section 3.4.3 and 3.4.5.
10	RDM support service	Yes, the CRC806-Database staff and members of the Z2 project can be contacted for support by the project participants.

Table 12.1.: Discussion of RDM demands.

The first demand, a *code of good research practice*, is implemented by following the recommendations and demands of the DFG *Proposals for good scientific practice* (DFG 1998), as described in previous sections. The probably most basic demand for a data catalog implementation (demand 2 of table 12.1), was delivered with great care, as described in chapters 6 and 9. Taking sufficiently care of Data Governance (demand 3 of table 12.1), is an obvious, but complicated demand, because it is not clearly defined, what is exactly asked. Within the CRC806-Database, the responsibility for data governance, which entails things like, access rights, data license, responsibility for correcting potential errors, and further things in this direction, are shared be-

tween the dataset authors, the dataset maintainers and the Z2 project staff. Active data storage, meaning storage that can be used during the data is actively (collaboratively) edited, is provided through three services that are provided by the RRZK:

- Sciebo (Sync & Share NRW 2016),
- SoFS (RRZK 2016b),
- and Basic Support for Cooperative Work (BSCW) (RRZK 2016a).

Sciebo (Sync & Share NRW 2016), an abbreviation of "*science box*", is a non commercial, academic, cloud based, file hosting service, based on open source ownCloud (ownCloud Contributors 2016) technology. The service offers 30 GB of cloud based storage, on request this can be extended to up-to 500 GB. Sciebo allows to share folders between several accounts to collaboratively edit data documents. Additionally, it is possible to share files from Sciebo as online resources link-able via an URL. These shares can be open access, or protected by a password.

SoFS provides 10 GB of online storage for members of the university of cologne, additionally projects can get shared storage spaces of negotiable storage capacity, though it is only possible to access this storage with a University of Cologne IT account. Consequently only members of the University of Cologne can have access to this resource. Although, it is possible to issue guest accounts for external researchers, this account needs to be renewed after short time periods, which makes this approach not very feasible, for longer term projects like the CRC 806.

BSCW is a Group ware system provided by the RRZK to facilitate web based collaborative work (RRZK 2016a). The platform offers several functionality, like shared calendars, ability to collaboratively edit documents and also an storage folder system to share data files, including a comprehensive URM and access rights implementation. This application can be used by all members of CRC 806 on invitation of a project admin.

Demand 6 as given in table 12.1, a data repository, implements the demands on secure long term data storage. The CRC806-Database implements this demand, by providing a securely redundantly backed up file system based solution. The AFS based secure data archive implementation was described in detail in section 6.1.1. As described in section 6.1.4, the CRC806-Database implements demand 7, as given in table 12.1, by providing the possibility of minting DOI's for datasets stored and published via the CRC806-RDM system. Demand no. 8 is not implemented specifically via a custom formulated statement, but it is implemented, by abiding the funders proposals for good scientific practice (DFG 1998), as well as general German, EU and international laws concerning topics like copyright, and the constitutional freedom of art and science. An Open Access policy (demand 9), is implemented with high priority in the concept of the CRC806-Database e-Science infrastructure, as described in sections 3.4.3 and 3.4.5. Demand 10, a dedicated RDM support service is not formally implemented, but practically such a service is available and delivered by the Z2 project staff and the CRC806-Database administrators.

Knowledgebase and VRE

The CRC806-KB was developed to address the demands for the facilitation of collaborative research environments and information sharing between the project participants. To provide such a platform a wiki based approach was chosen and implemented, as described in chapters 8 and 11. An internal platform to gather and exchange ideas, data and knowledge is the result of this approach. Any CRC 806 member can request an account for the wiki to be able to contribute to the collaborative knowledgebase.

The CRC806-KB in its current form is functioning since December 2015, thus the application is at time of writing this thesis just a few month old and at its very beginning, and will without doubt grow substantially in terms of knowledge items maintained, in knowledge scopes, meaning that the data model will grow, and also in terms of functionality, by adding for example time lines and bounding-box queries to SMW.

It is clear that the research within the CRC 806 has a truly interdisciplinary research setting, and the main questions are of spatio-temporal nature, As described in the introduction (see section 1.2). This entails, that most of the research questions asked, can be at least partly, answered by analyzing spatio-temporal patterns in the known and available data. A database, or in this case a KB to collect and maintain such a spatio-temporal database of published available data sources, is developed to support the research in the project. Where the main CRC806-RDM system is designed to host and publish data generated within the CRC 806, the CRC806-KB system is designed to collect data that is not (yet) published by the project and external data published in the research domains of interest to the CRC 806.

The approach has some similarities to the NESPOS platform, see section 12.7.2. NESPOS is based on the enterprise wiki Confluence (Atlassian Software 2014), and extended with custom extensions to facilitate the handling of structured information, in some sense similar to the SMW (SemanticMediawiki Contributors 2015) technology.

SDI open geospatial webservice publication

The GIS datasets are also published as OGC Open geospatial Web Service (OWS), to enable the networked integration of the data into client desktop GIS and WebGIS applications. The Spatial Data Infrastructure (SDI) consists of a GeoNode (GeoNode Contributors 2014) based backend, and an OpenLayers (OpenLayers Contributors 2015) based frontend (see figure ?? for a screenshot of the interactive WebGIS SDI frontend, showing the Köppen-Geiger classification (Willmes et al. 2015) of an LGM climate simulation dataset). The publication of OGC conform web services from the Desktop GIS into the CRC806-Database SDI is facilitated by using the OpenGeo Geo-Explorer Plug-in for QGIS (Boundless Inc. 2014). This plug-in allows a user friendly publication of GIS maps and datasets into the GeoNode, and thus GeoServer (GeoServer Contributors 2014) based SDI, by entering some few metadata information and some few mouse clicks from the GUI.

12.2.2. Funders demands

RRZK As introduced in section 3.1.2 the funders have certain demands for the conduct of RDM and e-Science infrastructure. Six separately formulated demands were identified in section 3.1.2, that are addressed here in table 12.2.2.

1	Development of a database to centrally store the research data produced by the CRC	This was addressed by implementing the CRC806-RDM,. as described in chapters 6 and 9.
2	Development of techniques and methods of handling (care, allotment, referencing, and linking) the data	This is addressed throughout the complete thesis and implemented in many ways in the CRC806-Database e-Science infrastructure. For example Linked Data (sections 6.1.6 or 4.6), Semantic Web (see section 2.2.4).
3	Support and facilitate the reuse of the data, through enabling interoperability with external repositories and data bases	This is also addressed through the application of DOI (see section 6.1.4) or the application of Semantic Web Technology (see section 3.4.4).
4	Development of a VRE, to facilitate collaboration and reuse of the data within the CRC	To address this demand, the CRC806-KB, as described in chapter 8 and 11, was implemented.
5	Facilitation of "interoperable components", such as Wikis, project management software or VCS	This was partly implemented by the CRC806-KB, as well through the GitLab instance, as described in 5.2.5, that is ac VCS, Wiki and Bugtracker.
6	Adaptation and implementation of state of the art technology, for electronic publication, identity management or virtual organizations	This was implemented by the Publication DB, the possibility to assign DOI's, the URM implementation, as well as through the overall CRC806-Database e-Science infrastructure.

Table 12.2.: Addressing the funders demands.

The first demand, for development of a database to centrally store the research data produced by the CRC, was addressed throughout almost the whole thesis, and implemented by the CRC806-RDM system, as described in chapters 6 and 9. Development of techniques and methods of data handling (demand 2), are also key focus of this whole presented work and are addressed through the here described design and implementation of the CRC806-Database semantic e-Science infrastructure. Support and facilitation for reuse of the data stored and published through the CRC806-Database, through enabling interoperability with external repositories and databases (demand 3 in table 12.2.2) was addressed in several ways. Most importantly through the implementation of SWT, as well as through the implementation of Open Data and Open Science principles. Another important feature of the CRC806-Database that supports the reuse of data is the ability to provide citations for the published data, through the application of DOI's, see section 6.1.4. The implementation of open standards for metadata, as well as for the data itself, most advanced in the CRC806-SDI, see chapters 7 and 10, through the implementation OWS

is also facilitating reuse of the data. Demand 4 (see table 12.2.2), was addressed through the implementation of the CRC806-KB, as described in chapters 8 and 11. The demand for facilitation of interoperable components (demand 5 in table 12.2.2), such as wikis, project management software and VCS, is also completely addressed and delivered by the CRC806-Database system. Wikis are applied by the CRC806-KB systems, project management software is used for developing and planing the implementation of the CRC806-Database, in particular the bugtracker and wiki provided through the GitLab instance (see section 5.2.5) implement this. A VCS is also provided by GitLab and used, to manage the code base of the CRC806-Database semantic e-Science infrastructure application. Adaption and implementation of state of the art technology, for electronic publications, identity management or virtual organization (demand 6) was addressed and mainly implemented by the CRC806-RDM system, as described in chapters 6 and 9, but also by the two other sub systems CRC806-SDI and CRC806-KB, although not as complete as in the CRC806-RDM system.

The close cooperation with the RRZK is also partly due to the demand by the DFG to cooperate with local institutional partners. The RRZK provides almost all infrastructure that is used by the CRC806-Database, as described in section 5.1.

12.2.3. CRC 806 demands

The project demands, as defined in the initial project proposals were given in section 3.1.3. In this proposal it was stated that *"the aim of this project is to solve the main problems of data storage and exchange within an interdisciplinary project by using a complex spatial database that allows the management of heterogeneous spatial and attribute data"* (Bareth et al. 2009).

This main demands were implemented by the CRC806-RDM and CRC806-SDI system components, with grate care as described in detail before. But this basic demands, were extended during the project runtime. Thus, further demands were expressed by the project partners after kick off of the CRC 806. The most important two demands were implementation of:

- Publication DB,
- and VRE.

That were implemented accordingly, as described in chapter 9 and 11. The main reason to develop the Publication DB was to deliver up-to-date publication lists for the CRC 806, its clusters, projects and researchers. This was additionally provided for the website of the IRTG graduate school of the CRC 806. Also, an overall publications catalog of the CRC 806 was implemented that way. This is and will be a useful tool for people interested in the research conducted in the course of the CRC 806. It will also be of much value for the reviewers of the next major CRC 806 evaluation for funding the proposed third phase of the project.

The VRE is still in its infancy, because it is in operation for just some month, since December 2015. It is relatively sure, that the application will grow and also change in many aspects, as explained in section 12.2.1.

12.3. Licensing and data policy

As introduced in section 3.5 the CRC806-Database applies the well known CC licenses (Creative Commons 2015), for data published through the CRC806-RDM data catalog. The CRC806-Database has no custom data policy, the policies applied are given by the funders policies, through general law like copyright, as well as by Open Science, Open Access and Open Data principles.

Many research projects develop and apply custom data policies, that include copyright agreements. An example for this approach would be the custom data policy implementation of the TR32DB (Curdt et al. 2015). This policy is setup to clarify and regulate the data provision and the data (re-)use, using the following paragraphs (Curdt 2014b):

Data provision:

- All data collected or created within the framework of CRC/TR32 must be made available to the TR32DB in a timely manner though at the latest within the current granted funding phase.
- The datasets must be sufficiently documented including data format, structure and quality to allow for easy access and use, if possible adhering to existing standards.
- Datasets submitted to the TR32DB must be accompanied by metadata. The metadata are open access.

Data use:

- Users have to inform the owner of the data of their intended use, and, in case of planned publication, the owner of the data has to be involved.
- Users agree not to distribute datasets of the TR32DB to others (third persons) without prior consent by the owner of the data.
- Users agree to use TR32DB datasets only for non-commercial scientific purposes.

This custom policy has the advantage, that it has some very clear and simple rules that are applied to all data stored in the TR32DB and in effect for every TR32 member, who signed that agreement. On the other hand, there is no experience with its applicability in a legal dispute in front of a legal court.

Application of general copyright law and implementation of constitutional freedom of art and science, is on the other hand legally safe (see section 3.5). A valid concern is, that the data authors who want to publish their data as Open Data, as defined by the Open Definition (OKFN 2016), have no option to do so via the TR32DB data policy. The above agreement formulates a non-open or closed access license, because of the limitation of non-commercial use, the demand for consent of the authors before further use of the data, as well as the demand for involvement of the data owner, are not compatible to and partly contrary to the Open Data and Open Science principles, as described in section 2.5.3.

Another valid critique is the mandatory upload of any data collected or created within framework of the CRC/TR32. Because this seems not to be feasible to enforce. The simple question

would be, at which point does a created dataset fall under this demand, and for what kind of data is this not applicable? Is a dataset, that was recorded in the early phase of the project, that was entailed errors in data capture or other failures, that disqualify the data for further use, mandatory for upload into the data base too? If not, then at what point is it? As stated on the policy, this will be decided by the TR32 steering committee, but this could result in many decisions that has to be made. Here the next question, about from which stage of completeness of the dataset under question, should be proceeded to the steering committee for decision whether it should be uploaded or not?

The CRC806-Database choose another approach because the researchers should be able to decide what they want to publish under their name and what they prefer not to publish does not have to be published. The researcher has to voucher with his name and credibility for his publications, it should not imposed upon them to publish data he or she does not want to be accountable and citable for. Though, data that is cited and applied in the course of an publication has to be available, also in the frame of the CRC 806, according to the demands of the DFG.

12.4. CRC 806 Repository lifecycle

In the IT realm it is common to describe applications and products along so called lifecycle models. Such a lifecycle model was also applied to the CRC806-Database application. As described in section 3.2.3, a RLC model was designed for the CRC806-Database RDM infrastructure, to formalize the three major phases of the RLC:

- Pre-project phase,
- Project phase,
- Post-project phase.

This concept was developed to address the demand of the DFG to guarantee the secure storage of the deposited research data, at least 10 years after the end of the project. To guarantee this, some measures need to be taken into account to plan and facilitate a solution to this demand.

As said, the CRC 806 RLC model defines the three major phases of a research project. The most crucial phase is the post-project phase, because in this phase, no adjustments to the implementation can be conducted.

The plan for the post-project phase is to remove all server side application vulnerabilities and reduce the risk by reducing the complexity of the application and transform the project into a static web page, see table 12.4 for an overview of how these vulnerabilities are addressed.

A major vulnerability of the CRC806-Database web application will be the user login and the according URM. In the post project phase the user login will be disabled completely. This ensures, that no account can be hacked to compromise the CRC806-Database in any way. The data upload functionality via PHP will be disabled too. The upload endpoint, as described in 6.1.1, poses a risk for possible attacks. If this application would be compromised, hackers could get the possibility to upload unwanted data into the AFS backend. The functionality for request

Vulnerability	Post-project implementation
User Rights Management	will be disabled, including user login
Data upload	will be disabled
Access requests	eMail to dataset owner
Dynamic queries	will be disabled
SDI and Maps frontend	will probably be disabled, this is not finally concluded yet

Table 12.3.: Application vulnerabilities and their post-project handling.

access to request restricted resources, will be disabled too, but an email contact will be given instead. The interested users can contact this address to ask for access to the data in the post-project phase. The access itself can be managed through an Apache `.htaccess` configuration, that asks for a password before delivering a resource. Dynamic queries will also be disabled, because they are triggered via PHP on the MySQL backend, that both will not be available after the application is transferred into a static web project. Maybe, a solution based on a static index and client-side JavaScript can be implemented to allow rudimentary searches of the data base, but this is not yet worked out and can't be promised so far. The SDI will most probably shut down after the project phase, because there is no plan if and how the SDI server infrastructure can or will be maintained. This shut down of the SDI would include the Maps frontend.

As part of the iterative Project phase, of the CRC 806 RLC, a major update including architecture and technology adjustments was conducted during the second project phase, as explained in detail in the following section.

12.4.1. The 2014 major update of the system

Because of the disadvantages, shortcomings, bugs and problems of the first version of the CRC806-Database, as described in Willmes et al. (2012a), it was decided to conduct a major redevelopment of the CRC806-Database web portal. This redevelopment was also described in detail in chapter 6. However, this major redesign did not affect the underlying CKAN based data catalogue and the AFS based long-term storage of data files and documents. The data archival and preservation aspects of the CRC806-Database were not affected by the changes to the overall system and remained as already described in (Willmes et al. 2014e), and laid out in detail in sections 5.1 and 6.1.1. Furthermore, it was taken care of preserving all URLs of dataset landing pages. This guarantees, that all links from other websites or even from already existing publications to content and especially data set landing pages remain unaffected. What actually did change, were the URLs of the OGC (WMS, WFS, WCS, and CSW) SDI services.

In table 12.4 the updated components are summarized and compared by their technology previously of the update and after the update.

The approach to replace all functionality implemented in client side AngularJS (Google Inc. 2014) technology with *Extbase & Fluid* Typo3 v6 core server side technology, results in more robust and flexible capabilities at the same time. The integration is tighter, because it could

Component/Feature	Before update	After Update
CMS	Typo3 v4.5	Typo3 v6
Extensions	Typoscript	Custom Extbase & Fluid
WebGIS	Custom GeoExt	Custom Extbase & Fluid + OpenLayers
SDI	MapServer & MapProxy	GeoNode
News & Blog	TT News Extension	Custom Extbase & Fluid
Publications	n/a	Extbase & Fluid

Table 12.4.: CRC806-Database major update.

increase the maintainability and resilience of the system by avoiding a complete layer of complexity and possible failure, by building all MVC functionality directly using the Typo3 technology of *Extbase & Fluid*. The switch to *Extbase & Fluid* technology also provided better development capabilities for the maintainability of the system and its development.

But not only the technological and user interface side of the CRC806-Database was improved, there were also some considerable new features developed. The feature, that provides possibly the most useful is the *related* feature (as described in section 6.1), that integrates the data catalogue, the publication database and the maps interface, by listing related entries for the resources in the according directories. Another useful new feature that needs to be mentioned, is the temporal filter, see also sections 6.1 and ?? for details.

The switch from the RRZK maintained Typo3 instance, to a self administered and maintained instance is an important step to guarantee long term availability of the CRC806-Database, after funding for the data management project of the CRC 806 is terminated, which is the case at the latest in summer 2021, if a third phase of the CRC will be funded. The DFG demands in its funding guidelines, that data produced by CRC members have to be accessible at least 10 years after the funding terminated. And to meet this requirement, a system that can stay working without maintenance on the application side, must be implemented.

Summarizing the main lessons learned from this architecture improvements, are for one the User Interface maintainability and its stability (rendering behavior) considerations for choosing the right technology. For two, it is crucial to break as less as possible (functionality, URLs, known interfaces, etc.) to preserve as much backward compatibility as possible. And third, the most important lesson, is to only implement changes in the backend, if they are not avoidable apart from very good reason. In conclusion, we can say that the effort of redesigning the web portal on the basis of *Extbase & Fluid* technology was worth the effort (Willmes et al. 2015).

12.5. Open Science

This section evaluates and discusses, how the CRC806-Database e-Science infrastructure implements the Open Science principles, as introduced in section 2.5. The design for implementing the principles was laid out in section 3.1. The implementation itself was described in the according infrastructure components in Part III (Implementation), in the chapters 6, 7 and 8, of this

thesis.

Open Science is a development in the academic landscape to make scientific research, data and dissemination accessible to all levels of an inquiring society, amateur or professional. The principles of Open Science are now also encouraged by the major research funders. For example, the European-funded project Facilitate Open Science Training for European Research (FOSTER)² has developed an open science taxonomy (Pontika et al. 2015) as an attempt to map the Open Science field.

The paramount Open Science feature of the CRC806-Database, apart from implementing Open Data, Open Access and Open Source principles, is the application of DOIs to datasets published via the CRC806-Database. This allows researchers to scholarly cite publications by the CRC806-Database in their scientific works, as described in 6.1.4.

In the following of this section the realizations of how the single sub principles of Open Science, i.e. Open Data, Open Access and Open Source, were realized within the CRC806-Database infrastructure, are discussed.

12.5.1. Open Data

To support the aims of the Open Data concept is an important aspect for the design and implementation of the CRC806-Database (see section 3.4.5). Dataset authors are encouraged to provide their data as open data, meaning by assigning them an open license and don't lock the data behind access constraints (see section 3.5). By implementing the five star open data badges, see section 2.2.4 and 6.1.5, a guidance and also an incentive for members to follow the open data principles is provided. Additionally, it works as a reassurance for the dataset authors to immediately see, if they maybe by mistake closed the access and re-usability of the data. By the implementation of the five star open data badges, the datasets are automatically checked, if:

- the data is open accessible,
- the dataset provides an actual data resource,
- the data resource is delivered in an open format,
- the dataset is assigned an open license,
- the data data is annotated with metadata,
- the data is linked in the sense of Linked Data and the Semantic Web.

The count (0 to 5) of open data stars and according badges, as shown in table 6.4 are assigned according to how many of the criteria are met by the given dataset.

To provide and implement Open Data is not only an act of friendliness and general openness, some maybe even would say naivety, it increases the chances of reuse and thus potential for citations of the according research. If published with a DOI it can be even directly cited, and generate scientific credit for the authors.

Basically, it should be the goal of every researcher, that his research results are noticed, discussed and at best reused and improved. Open Data is at the core of facilitating this in the age

²<https://www.fosteropenscience.eu>, accessed: 2016-03-10.

of the fourth paradigm.

On the other hand, if data that can't be public for reasons like privacy or other possible sensitive informations, it should not be published in the first place. To also fulfill the funders demands, for depositing all data that are the basis to published research for at least ten years, in this instance the data is only deposited for archival, and access is restricted accordingly. The metadata itself will be public though, because if the dataset was used to produce research results, at least the description of its contents has to be public if it was cited in an according publication.

12.5.2. Open Access

Open Access principles were implemented in several ways, they overlap to a considerable degree with the Open Data and Open Source principles, as explained in sections 2.5.3 and 3.4.5. Basically, Open Access applies the more general Open Data principles for scientific publications and research data, including the definition of different publishing models, as described in detail in section 2.5.3.

Apart from these formal definitions of certain publishing models, the implementation of open principles was and is of higher concern than just deciding a certain implementation. Nowadays, where the cost of publishing and distribution is not much more than running a website and some editorial and administrative staff, the times of high cost for printing and distribution of scientific knowledge are over, this leads to the fact, that the arguments for implementing Open Access are objectively convincing. In this regard the first paragraph of the *Open Access Manifesto* by Swartz (2008) is quoted here:

Information is power. But like all power, there are those who want to keep it for themselves. The world's entire scientific and cultural heritage, published over centuries in books and journals, is increasingly being digitized and locked up by a handful of private corporations. Want to read the papers featuring the most famous results of the sciences? You'll need to send enormous amounts to publishers like Reed Elsevier (Swartz 2008).

This quote is still true, eight years after the manifesto was written, and three years after the tragic death of its author. This circumstances makes clear that it is not a pure technical choice, but it is more a political, if not even ethical choice, that each publisher has to decide. Here is Swartz (2008) argument regarding this choice:

It's called stealing or piracy, as if sharing a wealth of knowledge were the moral equivalent of plundering a ship and murdering its crew. But sharing isn't immoral — it's a moral imperative. Only those blinded by greed would refuse to let a friend make a copy (Swartz 2008).

Thus, Open Access is implemented in the CRC806-Database where possible. But the choice to grant Open Access to the publications is still in the data creators and authors hands. In the long

run, this is sure, authors who choose the open options will gain the most, in terms of citations and in terms of profit (not only monetary) from contribution to the greater good.

12.5.3. Open Source

As defined in section 2.5.3 OSS facilitates reuse of a diverse set of software code and applications for implementing new independent applications. OSS is a more decentralized model of production, in contrast with more centralized models of development such as those typically used in commercial software companies (Raymond 2001). The CRC806-Database is completely built on Open Source Software, as described in section 5.2. The use of open source software for implementing the CRC806-Database has several reasons.

The first and most banal as well as obvious reason is that it reduces the costs, because Open Source software comes without a mandatory monetary cost. Though, that the application of Open Source would be without cost at all would be false. There is significant cost in terms of human resources, skill and time involved. Another reason for building upon OSS, is that the software does not have problems with license renewals to keep the infrastructure running. A practical example for problems in this sense, are esri ArcGIS server based SDI implementation, that stop working, if a license is expired, that has to be renewed yearly. This apparently introduces an additional cost for maintenance. We also applied certain OSS products, because they were offered and maintained by the RRZK. This is true, for all web based applications running on the Apache Webserver (Apache HTTPD Contributors 2015) based web projects, or the MySQL (Oracle Inc. 2015) based DB backends, the Typo3(Typo3 Contributors 2014) based CMS, and many more applications, as explained in section 5.2.

The fact, that many OSS products are well supported by commercial companies, and developed by large developer bases, as well as peer reviewed and improved by large user bases, makes OSS a very sound choice. The quality of the software product, as well as its security and support in case of failure is for many large OSS superior to many proprietary software products (Christl 2014). And finally, but most important for the scope of the here presented infrastructure, by applying OSS the use of proprietary black boxes are avoided. For software applications in a scientific context, such as in the GIS realm, it is important to be able to understand how exactly a certain tool or algorithm is implemented. This is only possible, if the source code is accessible for review, as ensured by the application of OSS.

12.5.4. Open Science in practice

In this section, two example applications, that emerged from the CRC806-KB (see chapter 8 and 11) information collection are explained. Both applications apply data collected in the CRC806-KB for creating new datasets and contributing additional information to the published record.

The applications both cite the underlying GIS datasets with own DOI's that were issued for this purpose by the CRC806-Database. The open access publication of the GIS datasets with according DOI's is the key feature of the CRC806-Database Open Science approach, as shown in

the following two examples.

This practical approach to the implementation of Open Science principles is the result of six years of developing an e-Science infrastructure, it was not planned from the onset of the CRC806-Database, that this kind of application would emerge. Apparently it did, and this is also to a significant degree the result of applying a continuing organic always developing and improving bottom-up approach to the implementation of the presented system.

Paleoclimate Köppen-Geiger classifications

The first research application, that emerged from the CRC806-Knowledgebase approach was a Köppen-Geiger climate classification of paleoclimate simulations. The creation of the Köppen-Geiger classifications was described and published in a Journal paper (Willmes et al. 2016), but the resulting GIS datasets were published as open data in the CRC806-Database, as shown in figure 12.1.

The paleoclimate simulations are originally produced by the Max-Planck-Institute for Meteorology in Hamburg, Germany. In the context of the Paleoclimate Modelling Intercomparison Project (PMIP) III project, three time slices were modelled:

- Pre-Industrial (1800 AD),
- Mid-Holocene (6k BP),
- LGM (21k BP).

The model simulations were conducted by several institutions. We decided to use the MPI-ESM-P (Max-Planck-Institut – Earth System Model – Paleo Version) model simulations (Giorgetta et al. 2013).

Based on the discovery of these paleoclimate simulations, in the course of research for potential data sources for the CRC806-Knowledgebase, an idea to compute climate classifications to reconstruct paleoenvironments emerged. This idea was successfully executed and the resulting classification maps and GIS datasets were published in the CRC806-Database:

- Pre-Industrial (Willmes et al. 2014c),
- Mid-Holocene (Willmes et al. 2014b),
- Last Glacial Maximum (Willmes et al. 2014a).

The Python script that implement the computation algorithm are also published as open data (Willmes et al. 2014d). The paper (Willmes et al. 2016), that was published in the Journal *Transactions in GIS*, was written under the aim to realize Open Science principles, by publishing all results of the work open access. The GIS datasets were published in the CRC806-Database including DOI, see the citation in the list above, as well as in the CRC806-SDI as OWS.

The Köppen-Geiger climate classification is based on vegetation-referenced temperature and precipitation thresholds, which differentiate in their combinations. According to the here applied updated classification scheme of Peel et al. (2007), six climate zones or main groups (first letter)

The screenshot displays a web application interface for the CRC806-Database. The main content area is titled "High Resolution Köppen-Geiger Classifications of Paleoclimate Simulations" and is maintained by Christian Wilmes. It features an abstract, a list of resources (PDFs), and a grid of related spatial datasets including Köppen-Geiger LGM, Pre-Industrial, and Mid-Holocene classifications in various formats (Vector, Raster, Clipped). The right sidebar contains metadata such as the project group, authors, license (CC BY-NC-ND), a spatial map, and an interval description for the Last Glacial Maximum. At the bottom, there are links to related datasets and a list of tags.

Figure 12.1.: Köppen-Geiger Paleoclimate classification publication, annotated with the open access GIS datasets in the CRC806-Database web application. Source: Screenshot.

are sub-divided by twelve climate types (second and third letter) based on the combination of temperature and precipitation patterns, resulting in 30 distinct climates. The climate zones are defined by temperature, except for the arid B climates, which are defined by precipitation. The climate types are mainly differentiated by annual precipitation and its yearly distribution, while the climate sub-types are differentiated by its seasonal variability (Willmes et al. 2016).

Time Slice Maps

The second Open Science applications, that is a direct result of the CRC806-KB data collection, are the *Time Slice Maps*. The aim of this project is to provide maps and GIS datasets for a paleoenvironment of a given region. The underlying paleoenvironment information is gathered from heterogeneous published sources.

The approach for collecting the data mainly consists of digitalization of maps (scanning, georeferencing and digitizing), or GIS modelling from textual information, of published paleoenvironmental information, which are not yet available in GIS formats. Consequently, the main focus of this work is on the acquisition of qualitative and analogue paleoenvironmental information, and on producing GIS datasets representing these information. These gathered and created datasets are then combined with already existing GIS datasets in the CRC806-KB, in order to produce comprehensive paleoenvironment geo-datasets and maps. See figure 12.2 for an example *Time Slice Map* of the LGM paleoenvironment of Europe.



Figure 12.2.: Time Slice Map of the LGM paleoenvironment of Europe. Source: (Becker et al. 2015).

The produced GIS datasets and maps are finally published in the CRC806-Database in form of a download-able dataset, including DOI, and via Open Geospatial Consortium (OGC) web services in the Spatial Data Infrastructure (SDI) of the CRC806-Database. In the following of this section,

the publication of this example dataset, the "LGM paleoenvironment of Europe - Map" (Becker et al. 2015), is described.

Dataset DOI publication Geodatasets and according maps are published including a Digital Object Identifier (DOI) minted via DOIDB³ (Ulbricht et al. 2016), the cooperation partner of the CRC 806 for issuing DOI's. Examples of paleoenvironmental GIS datasets, created with the here presented approach, and published in the CRC806-Database are (Verheul et al. 2015) and the here further described (Becker et al. 2015) dataset.

These datasets and maps are published with an appended strictly formalized descriptive document (see fig. 12.3), containing the metadata and some contextual information, citation of the data sources, as well as advice on the further citation of the dataset.

The figure displays three panels of a metadata document for the dataset 'LGM paleoenvironment of Europe - Map' by Becker et al. (2015).

- Left Panel:** Contains the title, authors, abstract, and context. The abstract describes the LGM paleoenvironmental data derived from the Kippen-Grige climate classification. The context section provides details about the data source and the map's resolution. A metadata table is also present.
- Middle Panel:** Contains temporal metadata (2.3), data sources (3), and maps and visualisations (4). The temporal metadata table lists the type, name, and interval of the data. The data sources section lists the datasets used, such as the Kippen-Grige climate classification and the LGM map. The maps and visualisations section includes a map of Europe showing the LGM paleoenvironment and a legend.
- Right Panel:** Contains data resources (5), web resources (5.2), and acknowledgements. The data resources section lists the files available for download, including the map and the metadata. The web resources section lists the URLs for the data and the metadata. The acknowledgements section thanks the funding agencies and the CRC806-Database.

Figure 12.3.: Example CRC806-Database publication metadata document of Becker et al. (2015).

The metadata of the example dataset (Becker et al. 2015) is strictly formalized. The metadata schema contains the following categories:

- Basic Metadata
- Spatial Metadata
- Temporal Metadata
- Data Resources
- Bibliographic References

The basic metadata holds general elements like *Title*, *Authors* and *Year* from DublinCore (Dublin Core Metadata Initiative 2004) and DCAT (Maali et al. 2014), to describe the basic publishing informations of the published dataset. For spatial annotation ISO19115-1 (2014) metadata elements, like *Place* or *Region* are used. Temporal metadata for annotation of events or

³<https://doidb.wdc-terra.org>

intervals is applied in a consistent way according to ISO 8601. Resources are cited like references of scholarly publications, including author, title, etc. and if available a web accessible URL. And Bibliographic References are handled like in any scholarly publication, by compiling and providing a bibliographic reference list.

12.6. Data metrics

In this section a discussion of the overall data basis, that is collected within the CRC806-RDM repository until now, as well as statistics about the use and adoption of the CRC806-Database is conducted. This quantitative approach to analysis of facts, numbers and statistics about an RDM repository, is also known under the term of *data metrics*, as it was already introduced in this work in section 9.6.1, to describe the publicly available data statistics of the CRC806-Database.

12.6.1. Data basis

Here we take a look at the current data basis of the CRC806-RDM component. This includes the datasets and publications stored in the data catalog and the publications database. These resources are managed in the CKAN backend. The statistics shown in this section are based on the evaluation and analysis of simple metrics, such as count of downloads, type of data, and so on. In figure 12.4, the datasets and publication counts, as of March 2016, by clusters and projects are given.

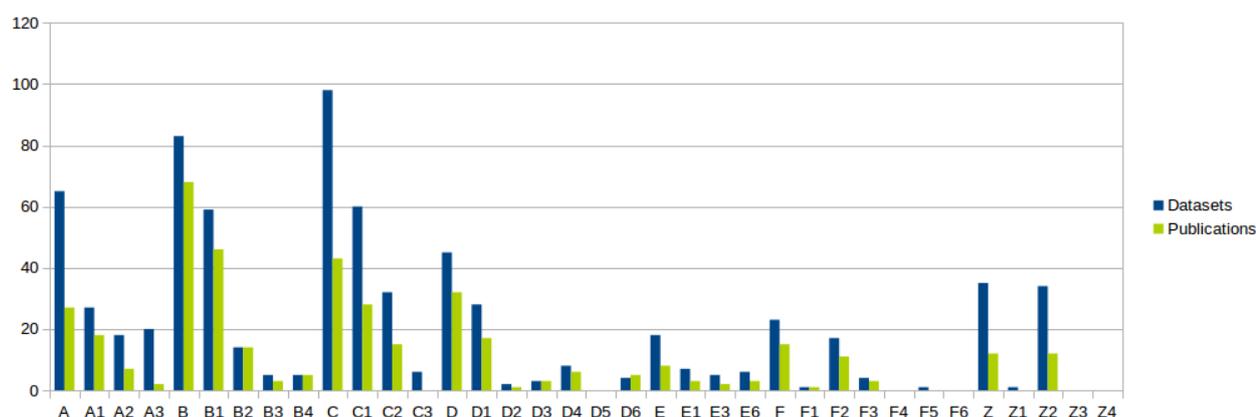


Figure 12.4.: Datasets and Publications by Cluster and Project, as of March 2016. Source: Own work.

Currently, 332 datasets of which 182 are publications, are stored in the CRC806-RDM system, and these are the numbers from which the here shown diagrams were made. Cluster C has the most datasets (98) in the CRC806-RDM repository and cluster B has the most publications (68) in the repository.

12.6.2. Resources and file system statistics

Each dataset or publication can have zero or more resources. Currently 391 resources are stored in the CRC806-RDM system. In figure 12.5, the distribution of different data formats is shown in a pie chart.

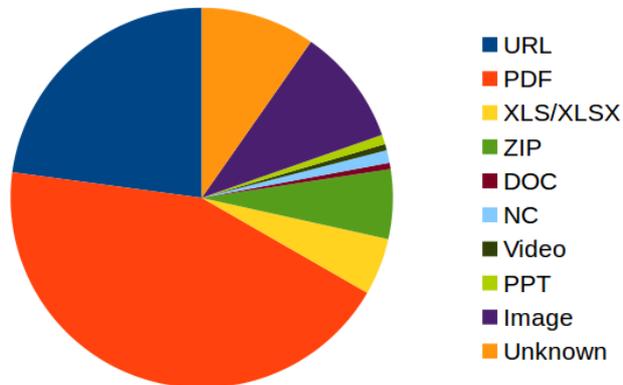


Figure 12.5.: Data format allotment of resources stored in the CRC806-RDM repository, as of March 2016. Source: Own work.

By far the most resources, are PDFs or URLs (link to externally stored resource), this is due to the fact that publications are almost only one or both of these two formats.

Further most common formats are Spreadsheets (XLS or XLSX), zip archives (ZIP), this includes most of the GIS datasets, and images (JPG, PNG or TIF). There is one more larger fraction assigned to unknown, in this cases, the format was not successfully detected during data upload. This is the case if the format is uncommon, i.e. the upload application does not have a category for it, or the file has a corrupted mime type and/or unknown file extension.

Currently, all resources stored in the AFS backend add up to 13.717 GB, as shown by cluster in table 12.5. The resources are stored in the file system in 385 directories and 561 files.

Cluster	Size in GB
A	0.968
B	0.213
C	0.515
D	8.2
E	1.5
F	0.021
Z	2.3
All	13.717

Table 12.5.: Volume of data stored in the AFS backend by cluster, as of March 2016.

12.6.3. Access statistics

In figure 12.6 all resources stored in the data catalog and the publications DB are shown by its number of downloads. Each of the 391 resources is visualized by a vertical line. The lines length shows the number of downloads. The color shows, if the resource is open access (blue) or not (red). This is a typical long-tail graph. There are a few datasets that have many downloads and many that have few downloads.

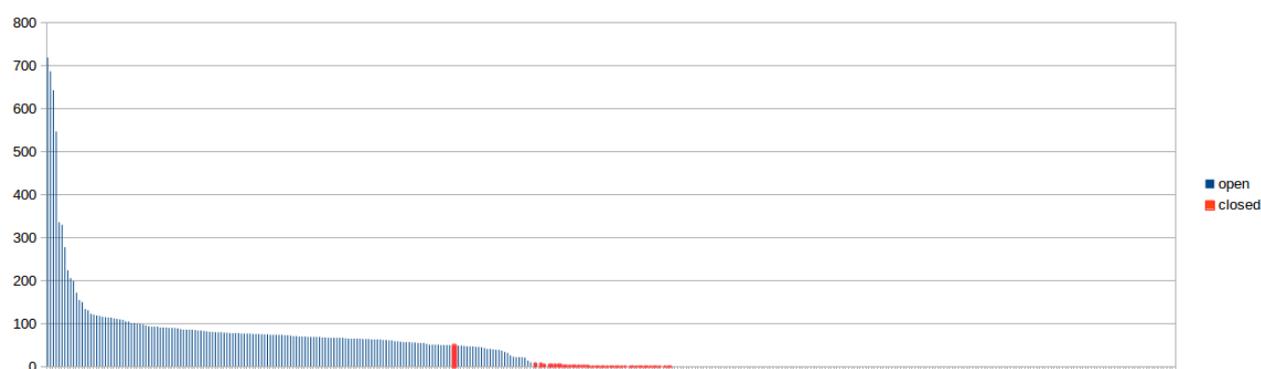


Figure 12.6.: Downloads per Resource, as of March 2016. Source: Own work.

It is clear that Open Access resources are downloaded significantly more often than closed access resources. Added up, all resources of the CRC806-RDM repository were downloaded 15221 times, since late 2014 when the new Typo3 v6 based website went online. The most often downloaded resource has 718 downloads, another 32 resources have more than 100 downloads. The most downloaded closed access resource has 49 downloads, the next most downloaded closed resource has just 8 downloads. 174 closed resources have 0 downloads, only two open resources have 0 downloads.

12.7. Cooperation with related Infrastructures

The CRC806-Database is not the only infrastructure in Germany, that covers the domains of prehistoric archaeology and paleoenvironment. The CRC806-Database aims to cooperate with related infrastructures where possible. In the following two established cooperations with ROAD and with NESPOS are presented.

12.7.1. ROCEEH Out of Africa Database

The Role of Culture in Early Expansion of Humans (ROCEEH)⁴ project concerns questions about how the human species expanded in several waves from its original African homeland to encompass Eurasia and possibly back into Africa (Märker et al. 2013).

Apparently this scope is very closely related to the scope of the CRC 806, which is responsible for the huge potential of a cooperation between the CRC806-Database and ROAD.

⁴<http://www.roceeh.net/>, accessed: 2016-03-07.

The ROCEEH project is funded by the Heidelberg Academy of Sciences and is projected to run for 20 years. The research aim is to reconstruct the spatial and temporal patterns of the expansions of hominins between three million and 20,000 years ago in Africa and Eurasia (Märker et al. 2014).

The “ROCEEH Out of Africa Database” (ROAD)⁵ incorporates a large array of variables from the fields of geology, geomorphology, palaeontology, palaeobotany, palaeoanthropology and archaeology, assimilating these data in vector, raster and text formats. The complex structure of ROAD resulted from in-depth discussions among staff members and with outside experts about objects, contents and relationships of the various disciplines that were then incorporated into the database in a holistic way. The different themes tackled by ROCEEH are translated into objects, object keys, relationships among objects, and descriptive attributes (Märker et al. 2009).

THE RAD DB is based on PostgreSQL & PostGIS for geospatial data management, and on a MapServer instance for its SDI setup. Based on these foundations a collaboration, on the basis of OGC OWS data exchange, between the CRC806-Database and ROCEEH ROAD is setup, to exchange GIS data.

We focus especially on the spatial data available in both systems as well as on the environmental information. Therefore, we examine and test exchange interfaces based on Spatial Data Infrastructure technology (OGC Standards) (Märker et al. 2014).

12.7.2. NESPOS

NESPOS⁶ (Pleistocene People and Places) is a database that primarily handles data about Neandertahl remains, maintained and curated by the Neandertahl Museum e.V. and the NESPOS society (Bradtmöller et al. 2010).

The NESPOS database, as well as the according NESPOS society has overlap with members of the CRC 806 C cluster, and especially with project C1. Prof. G.-C. Weniger is the director of the Neandertahl Museum, as well as the head of the CRC 806 C cluster. Thus, a cooperation between the CRC806-Database and the NESPOS database makes sense and was setup from the beginning of the CRC 806 project.

For the time between 2011 and 2015, the NESPOS server infrastructure was hosted at the RRZK. The server was a RRZK provided RHEL instance, with limited root access for the Z2 project, similar to the `sf806s rv` (see section 5.1). The web application was migrated in 2010 to the RRZK based infrastructure, and maintained and administered for five years until 2015 by the author of this thesis. The NESPOS web application is based on the Confluence web application framework (Atlassian Software 2014). Confluence is an enterprise wiki system, that has some similarities to Semantic Mediawiki. It is also possible to store and handle structured data in confluence, by using according extensions. Confluence is a Java (Oracle Inc. 2016) based web application hosted on an Apache Tomcat server (Apache Software Foundation 2016). Tomcat is an open source software implementation of the Java Servlet, Java Server Pages (JSP), Java

⁵https://www.roceeh.uni-tuebingen.de/roadweb/smarty_road_simple_search.php, accessed: 2016-03-07.

⁶<https://www.nespos.org>, accessed: 2016-03-07.

Expression Language and Java WebSocket technologies, used to host the Confluence Java based web application.

At the beginning of the CRC 806 project it was thought out to closely cooperate, and even integrate the CRC806-Database with NESPOS. One reason this plan was not executed, was due to the closed setup of NESPOS, where only members of the NESPOS society can edit the database. There were also some attempts for creating dynamic web maps from the NESPOS sites collection, of which some first prototypes were functional, but it was not commenced to integrate these datasets into the CRC806-Database, because there was no official demand or mandate for this.

In 2015 the NESPOS system was migrated to a commercial web hoster. During this migration, the Confluence (Atlassian Software 2014) backend was updated to the current LTS version of the system, which was not possible to carry out with the limited resources of the Z2 project.

13. Conclusion

The aim of this thesis was to develop a state-of-the-art, and possibly even one step further, scientific information infrastructure for an interdisciplinary research project. The necessity of useful, sound, and professional RDM for interdisciplinary collaborative research projects is nowadays beyond discussion and thus consensus in the sciences and society, in times of the ongoing *information revolution* (Bernal 1939), the *data deluge* (Hey et al. 2003), and the manifestation of the *fourth paradigm* (Hey et al. 2009).

This thesis aimed to contribute answers to the questions, of how information infrastructure should be implemented, by presenting an implemented approach and, consequently, particular technological answers to these questions. Nobody says that the here given answers, that were found and worked out during this study, are the only correct answers to these questions and demands. But that the here presented infrastructure successfully answers the introduced demands of the funding agency, as well as by the project participants, was shown in this thesis and proofed by the presented CRC806-Database semantic e-Science infrastructure implementation.

Conducting a doctoral thesis about the design, development, and implementation of web-based research infrastructure is not directly what most people understand as a straightforward research thesis project. According to the straightforwardness, they are right. Nonetheless, to design and implement a system like the one presented here is hard work that requires a lot of research for choosing the right technology, being knowledgeable of the state-of-the-art in the field, identifying the best approaches, and focusing on suitable solutions. And most important the realization of these solutions.

In the following of this section, some specific conclusions and observations according to certain sub-topics of this thesis are given.

13.1. Development and adaption of the CRC806-Database

The CRC806-Database was developed from the start of the CRC 806 project in late 2009. At this point in time, it was thought out to simply transfer the TR32DB concept to the demands of the CRC 806, as described in (Bolten et al. 2010). The TR32DB is a from scratch developed web application and RDM system designed to meet the demands of the TR32 project (Curdt et al. 2008; Curdt et al. 2009). The development of the TR32DB started in 2007, thus the project was running since two years, when the CRC 806 project started.

After a careful examination of the technology applied for the TR32DB application in comparison to out-of-the-box OSS systems, it was decided to build the CRC806-Database upon a combination

of several OSS products. As discussed above, this approach has several advantages (use of out-of-the-box features, strong developer community improving the systems, bug-fixing by community, support from the community, etc.) but also some disadvantages (vulnerability by known bugs and exploits, maintenance and keeping the infrastructure up to date, etc.) compared to the TR32DB approach. A major argument for trying a different approach was the demand and wish to conduct research about how to implement a RDM infrastructure and find new way's for supporting a different CRC's RDM concept.

The CRC806-Database had a relatively difficult start in terms of acceptance adaption by the project participants. This is due to the naturally less features and functionality in the beginning of the project, as well as some glitches with the web application in the first several months, that had bad impact on the trust into the technology from the project partners. Also not to be underestimated should be the fear of being forced to publish data and documents that some researchers preferred to not publish in the first place. As discussed in the following section 13.2, this issue had its role in the bad acceptance of the infrastructure as well.

The implementation of the Publications DB, as described in section 6.2, significantly improved the adoption of the infrastructure by the project partners. Because the researchers of the project manage their publication record via the CRC806-Database now, all publications produced in the course of the CRC 806 are at least bibliographically managed in the CRC806-Database. But to be honest, most researcher just enter the metadata of their publication, to get their publications lists on their researchers profile pages on the main CRC 806 website. Often not even an abstract is added, not to speak of a link to the publishers record or even the PDF of the according publication. This use of the CRC806-Database has to, and will be adjusted and improved in the future. We already implemented a warning message, if a new publication record will be created without an abstract or any resources. This warning message has to be confirmed before the record can be published. Though, it was decided to not force the project members to provide information that they do not want to provide. But at least, by the application of this warning message, the users are made aware of their suboptimal contribution, if trying to publish without these informations.

13.2. Data sharing and data reuse

Obviously there are many problems to the practice of data sharing and data reuse. We can identify the following three existing main obstacles for efficient data integration and reuse in inter-domain research efforts like the CRC 806. The first problem is the format in which the data are available. Here, we aim to find ways to facilitate tools and infrastructure to enable sharing in the most compatible ways. To reach the best compatibility with Tim Berners-Lee's Five Star Data Meme (see section 6.1.5) is the goal for this research. The second problem for better data reuse addresses the semantics, used to represent the data. To tackle this obstacle, semantic web technology (SWT) was applied within this thesis. The third problem is the lack of knowledge about the published existent data of potential interest. This addresses the exposition,

the discovery of data, and the interface available for both tasks. A solution to this problem was implemented by the CRC806-KB (see chapter 11).

A further prominent obstacle to successful data sharing, is the (lack of) documentation of the research data. To be useful or reproducible, a dataset must be accompanied by descriptive information (i.e., metadata) (Gray et al. 2002). Preparing documentation is frequently the most laborious step for researchers in taking data from usable within the lab to usable by others, and rewarding this effort is a major impetus for data publication (Kratz et al. 2014).

Also simple technical and administrative problems can hinder the user experience for potential data sharing. Usability is key for the success of an IT application (Nielsen 1999), thus if the user experience is good, probability for better adaption from the potential users is higher. For example if data can only be uploaded via an extra software, for example via SSH or FTP, this can be a considerable constraint for the motivation to share data. In some cases, the decision to share data can be spontaneous. In this cases, if the researcher who just prepared a dataset, is hindered to upload his data into the publication platform, it is possible that he or she will turn this plan down, and later does not follow up on it.

Data can be uploaded into the CRC806-RDM repository directly through the web frontend, using a user friendly web based GUI. No external or limited user accounts for access to data upload or data sharing is needed in the CRC 806. These demands were all implemented through CRC806-Database web applications.

Another major obstacle to data sharing is of cultural and social nature. The understanding of the benefits of open data, open access, open access and openness in general in sharing for the own work and progress, is in some places more prevalent and in others less. In archaeology the culture is more skeptical against openly sharing data, information and knowledge with colleagues or even with the public. There is often a fear of being robbed, of either ideas or facts, like locations of excavation sites, or new insights into findings, like new dated ages for artefacts and the like. In this discipline it is also still the norm to publish vast amounts of knowledge in inaccessible edited collections of few printed items. This kind of publication is known as *grey literature*. In times of the WWW this is indeed not necessary. In most of these cases it is simply not wished for to have these editions and volumes published online, because it is actually wanted to restrict access to the work. This practice could be described as, at least to some extent corrupt and wrong in the sense of what science is about. But it is not the role of the data manager to force the researcher to their better luck. This can only be achieved by the science funders, hardly demanding for more open publication practices. Providing infrastructure, that is able to publish any kind of information scholarly citable, like the CRC806-Database, can only help in this direction, but it is not able to solve the cultural and social obstacles in this regard.

In terms of data reuse, the attitude is more positive. Almost every researcher is keen to reuse published data for his own analyses and research. Under the principles of good scientific practice, it is normally also well adapted to cite and credit the data sources. For example, in archaeology it is common practice to compile a dataset of site locations, that are classified into a common pattern, e.g. occurrences of artefacts from the same cultural time period. These loca-

tions are listed in a table within the (paper) publication, including the according citation of the data source. The CRC806-KB, as described in chapters 8 and 11, was developed to allow the researchers of the CRC 806 share their knowledge about reusable data sources with each other and also with the interested wider community and public. The CRC806-Database further tries to cater into this direction of data reuse, by providing compiled GIS datasets, for example of paleoenvironments for a given time slice, see section 12.5.4 for a detailed discussion of this approach. The datasources for these time slice maps are compiled from queries on the CRC806-KB application. Also, there are initiatives from the wider community to foster data reuse. The Computer Applications & Quantitative Methods in Archaeology (CAA) society, for example, awards a yearly price, the Re-Cycle award, to the best use case or application of successful data-reuse in the domain of archaeology. Initiatives like this help to emphasize the importance and the overall benefits of data reuse for archaeology, as well as for the sciences in general. More on this topic in the context of the application of open science is concluded in the following section 13.3.

13.3. Open Science

As discussed in section 12.5, the application and implementation of Open Science principles and technology is an important aspect of the CRC806-Database semantic e-Science infrastructure.

A main reason for supporting Open Science within the CRC 806 is the conviction of its overall positive impact on the research within the project and its communication to the scientific community. As well as Open Science's positive impact on economic and cultural growth (Jong et al. 2014; Sitek et al. 2014), and to the overall good for the broader public and society.

It is proven, that open science strategies have an overall impact on innovative performance of research projects (Jong et al. 2014). Additionally, it is wished for, or even demanded by the main science societies (Max-Planck-Gesellschaft 2003) and funders in Germany (Engelhardt 2013; DFG 2014; DFG 2015) and also on EU (European Commission 2013) and further international levels. The following official statement of the European Commission, makes the official stance and policy on Open Science and Open Access of the EU quite clear:

The global shift towards making research findings available free of charge for readers, so-called 'Open access', has been a core strategy in the European Commission to improve knowledge circulation and thus innovation. It is illustrated in particular by the general principle for open access to scientific publications in Horizon 2020 and the pilot for research data (European Commission 2013).

The CRC806-Database policy on Open Science is about support and enhancement of the Open Science adoption within the CRC 806. Project participants are not forced to follow Open Science principles, but they are and will be encouraged to do so.

Concludingly, the CRC806-Database supports Open Science in several ways. The support for publishing Open Data, by allowing to grant open licenses (see section 3.5), as well as support for

open formats and open standards, especially for geodata as described in chapter 7, is apparently provided by the CRC806-Database.

The support of scholarly Open Access publishing of datasets and even publications, is provided through the combination of Open Licenses and most important by the application of DOIs for resources published via the CRC806-Database. As described in section 2.4.3, DOI's allow to properly cite web resources in scholarly publications.

And last but not least, the support of VRE as implemented by the CRC806-KB, also helps to foster Open Science, Open Access and Open Data principles within the CRC 806, by simply facilitating collaboration on a data collection and providing an infrastructure for data sharing, and thus its possible reuse.

In the end, it should be the goal of every researcher and scientist to be visible, and to have his or her research published openly accessible by anyone. Not only because it is nice and friendly to be open. To publish open access has many advantages compared to closed access publication (Sitek et al. 2014; Friesike 2014). Visibility is a key success factor in terms of citations and impact, and this is to some extent the capital of any researcher, for building a career or for maintaining and increasing reputation.

13.4. Semantic e-Science

As defined in section 2.5.1, Semantic e-Science is the combination of Semantic Web technology (SWT) and e-Science.

e-Science fills this gap between straight forward research based science, and the research on infrastructure development to facilitate research. Research conducted in the domain of e-Science is thematically broad, from the technological know-how to the essentials of the disciplines considered by the to be designed infrastructure. In the here shown example of the CRC806-Database semantic e-Science infrastructure, the by itself thematically very broad disciplines of Geosciences, Archaeology and cultural Anthropology are the scope.

Right now, it is beyond state-of-the-art but advanced technology to deliver full feature Semantic Web applications from a relatively small project, like the Z2 Data Management Project of the CRC 806, where one PhD student and two student assistants are implementing the whole system. The reason, that SWT applications are more hard to deliver, than state of the art Web 2.0 applications, is the lack of well established, decently maintainable, and user friendly out-of-the-box SWT open source applications. To the knowledge of the author, neither proprietary nor Open Source projects, can deliver what is needed in this realm yet. Though, it would not be to hard to extend existing CMS with fully fledged SWT capabilities like RDFa annotation of content and providing the content in RDF format accordingly. If that would be standard, the addition of an SPARQL endpoint, to make the RDF content query-able would not be to hard either. Any of the existing open source SPARQL endpoints, available for any major programming language or framework, could be facilitated here.

But none the less, the here presented infrastructure aims to add a further step to the overall

progress in creating a better solution for semantic e-Science infrastructures.

14. Outlook

This last chapter gives some outlooks on developments and goals for the near future of the CRC806-Database and the Z2 project. In general, the CRC806-Database will stay in continuous development until it will be transferred into a static web application for post-project operation, as described in section 14.2. Ongoing bug-fixes and feature request, as well as improvements, will keep the Z2 project busy until the project ends.

One of the most important development paths for the future of the infrastructure will be the enhancement of its Open Science feature, that will be motivated by projects like the Time Slice Maps (see section 12.5.4), and further to come paleoenvironment GIS projects in this direction.

Another major development direction for the CRC806-Database will be the improvement of its Semantic Web capabilities. The potential of SWT is massive. If mature user friendly tools are widely accessible, this technology can facilitate a truly distributed communication space, centralized platforms like Facebook and Google will lose power, because what they facilitate, social networks and data lookup (web search), is technologically facilitated since long time through SWT in an open distributed manner. It just needs to be more adapted.

One problematic area in the SWT context are the opportunities to violate privacy. Protecting privacy in the Linked Data context is likely to require a combination of technical and legal means together with a higher awareness of the users about what data to provide in which context (Bizer et al. 2009).

Many existing problems that contribute to the current data sharing and reuse situation, as well as wider problems like closed platforms as information hub, or search engines and social networks for example, are already theoretically solved within the Semantic Web and SWT realm. In this regard, the following quote says a lot:

Whatever the cause, almost everyone can find a reason to support this grand vision of the Semantic Web. Sure, it's a long way from here to there – and there's no guarantee we'll make it – but we've made quite a bit of progress so far. The possibilities are endless, and even if we don't ever achieve all of them, the journey will most certainly be its own reward (Swartz et al. 2001).

Until now, the CRC806-Database infrastructure is not very well known in the wider community, we hope this will change through more and more publications and valuable datasets, that will be added through the endeavors described in section 12.5.4. The ongoing cooperation with ROCEEH and NESPOS will also do their part in increasing the visibility of the CRC806-Database, also by joint projects and publications in the course of these cooperations.

14.1. Data preservation

The problem of preserving the CRC806-Database infrastructure and most importantly its data resources will be solved and implemented in the third phase of the CRC 806 project. The CRC 806 has made an agreement with the RRZK, that sufficient IT server infrastructure will be available for the CRC806-Database infrastructure for at least 10 years after the end of funding of the CRC 806. This was also described in the project proposals (Bareth 2009; Bareth et al. 2013). It is well possible, that the data will be available after the end of the 10 years guaranteed preservation, but can not be guaranteed beyond the 10 years.

Maybe a transfer of the data to a long term data preservation solution, will be conducted, if demanded or requested later on, but this is not yet planned. Central infrastructure for long term data preservation, that does not have to have discipline specific features, would be very beneficial to the German or even European research landscape. A DataCite approach (DataCite Metadata Working Group 2011), with a simple metadata schema for describing datasets, and a well designed long term preservation infrastructure and according strategy, as well as sufficient funding. The EU funded Zenodo¹ infrastructure is a promising implementation of this idea. This could be facilitated by the universities (its computing centres and/or libraries), or by the research funders agencies or by the governments ministry of science and education.

Preserving the data is one thing, to hold the data usable, is a whole other story. It is well possible, that data formats that are today well supported by many software products, are not well supported in the future. This could be for example the case for certain GIS formats, like ESRI Shapefile (esri 1998), or NetCDF (Domenico 2011). But if there exists an open source application to access these data formats, the probability of having a chance to somehow transfer the old data format into a currently used one is significantly higher, compared to if there are only proprietary solutions for access to the given format. The reason for this is clear. The code to access and read a data format is a kind of low level documentation of the data format itself. Even if the code could not be executed anymore, the source code could be read and translated into then current software development frameworks, to implement a tool to access the data.

14.2. Post-Project phase and availability of the systems

As described in section 3.2.3, and 12.4, the CRC806-Database will be transferred to a static web project at the end of the CRC 806 funding. The exact plan of how this will be conducted is still in development, but some constraints are given in the following list:

- Preserve all URL's and links.
- preserve as most interaction as possible, e.g. through JavaScript based clientside processing.
- find a solution for a static site search, e.g. based on a static indexing and JavaScript based search engine.

¹<https://zenodo.org/>, accessed: 2016-02-14.

There are several tools available to create a static web project from dynamic webpages, by systematically requesting any possible site (dynamic request) and pre-defined query results. These web pages will be stored as static HTML documents, including images and client side executed JavaScript content and functionality. This process is also known under the term web caching. The popular Linux tool *GNU Wget* (FSF 2016) offers an open source implementation for this approach.

Part VI.

References

Bibliography

- Aasen, H., Gnyp, M. L., Miao, Y., Bareth, G., 2014. Automated Hyperspectral Vegetation Index Retrieval from Multiple Correlation Matrices with HyperCor. *Photogrammetric Engineering & Remote Sensing* 80 (8), 785–795.
- Abbate, J., 2000. *Inventing the Internet. Inside technology.* MIT Press.
- Abecker, A., Bernardi, A., Hinkelmann, K., Kühn, O., Sintek, M., 1998. Toward a Technology for Organizational Memories. *IEEE Intelligent Systems* 13 (3), 40–48. issn: 1541-1672.
- Ackoff, R. L., 1989. From Data to Wisdom. *Journal of Applied Systems Analysis* 16, 3–9.
- Addis, M., 2015. RDM workflows and integrations for HEIs using hosted services. Tech. rep. (accessed: 2015-09-14). figshare. <http://dx.doi.org/10.6084/m9.figshare.1476832>,
- Adida, B., Birbeck, M., McCarron, S., Herman, I., 2015. RDFa Core 1.1 - Third Edition. (accessed: 2015-09-22). W3C. <http://www.w3.org/TR/rdfa-syntax/>,
- Agafonkin, V., 2015. Leaflet - open-source mobile-friendly interactive maps. (accessed: 2015-08-24). <http://leafletjs.com/>,
- Alamri, A., Bertok, P., Thom, J. A., Fahad, A., 2015. The mediator authorization-security model for heterogeneous semantic knowledge bases. *Future Generation Computer Systems.* issn: 0167-739X.
- Allemang, D., Hendler, J., 2011. *Semantic web for the working ontologist: modeling in RDF, RDFS and OWL.* Morgan Kaufmann Publishers/Elsevier.
- Allianz Initiative, 2011. Definition: Virtual Research Environments. (accessed: 2016-03-24). http://www.allianzinitiative.de/en/core_activities/virtual_research_environments/definition/,
- Alquier, L., McCormick, K., Jaeger, E., 2009. knowIT, a Semantic Informatics Knowledge Management System. *Proceedings of the 5th International Symposium on Wikis and Open Collaboration, WikiSym '09.* ACM, Orlando, Florida, 20:1–20:5. <http://doi.acm.org/10.1145/1641309.1641340>,
- Analytical Graphics, Inc., 2015. Cesium is a JavaScript library for creating 3D globes and 2D maps in a web browser. (accessed: 2015-08-25). <http://cesiumjs.org/>,
- Andel, T. van, Davies, W., 2003. Neanderthals and modern humans in the European landscape during the last glaciation: archaeological results of the Stage 3 Project. *McDonald Institute Archaeological Research monographs*, Cambridge, UK.
- ANSI/X3/SPARC Study Group on Data Base Management Systems, 1975. *Data Base Management Systems Interim Report.* Tech. rep. (FDT (Bulletin of ACM SIGMOD) 7:2). American National Standards Institute.
- Apache HTTPD Contributors, 2015. Apache - HTTP Server Project. (accessed: 2015-08-07). The Apache Software Foundation. <http://httpd.apache.org/>,
- Apache Jena Contributors, 2015. Apache Jena. (accessed: 2015-08-18). The Apache Software Foundation. <https://jena.apache.org/>,
- Apache Software Foundation, 2016. Apache Tomcat. (<http://tomcat.apache.org/>, accessed: 2016-03-07). The Apache Software Foundation.
- Atlassian Software, 2014. Confluence. (<https://de.atlassian.com/software/confluence>, accessed: 2014-12-08).

- Atzori, L., Iera, A., Morabito, G., 2010. The Internet of Things: A survey. *Computer Networks* 54 (15), 2787–2805. issn: 1389-1286.
- Baaser, U., 2010. CampusGIS routing – a web-based LBS for the University of Cologne. *Proceedings of Joint International Conference on Theory, Data Handling and Modelling in GeoSpatial Information Science*,
- Ball, A., 2013. The DCC Disciplinary Metadata Catalogue. *Proceedings of the International Conference on Dublin Core and Metadata Applications, DC 2013*. Lisbon, Portugal.
- Ball, A., 2014. How to License Research Data. DCC How-to Guide. Digital Curation Centre. <http://www.dcc.ac.uk/resources/how-guides/license-research-data>,
- Ball, A., Duke, M., 2015. How to Cite Datasets and Link to Publications. (DCC How-to Guides). Digital Curation Centre. <http://www.dcc.ac.uk/resources/how-guides/cite-datasets>,
- Barabási, A.-L., Bonabeau, E., 2003. Scale-Free Networks. *Scientific American* 288, 50–59.
- Baran, P., 1964. On distributed communications networks. *Communications Systems, IEEE Transactions on* 12 (1), 1–9.
- Barbuti, R., Giacobazzi, R., Levi, G., 1993. A General Framework for Semantics-based Bottom-up Abstract Interpretation of Logic Programs. *ACM Trans. Program. Lang. Syst.* 15 (1), 133–181. issn: 0164-0925.
- Bareth, G., 2009. GIS- and RS-based spatial decision support: structure of a spatial environmental information system (SEIS). *International Journal of Digital Earth* 2 (2), 134–154. eprint: <http://www.tandfonline.com/doi/pdf/10.1080/17538940902736315>.
- Bareth, G., Doluschitz, R., 2010. Spatial data handling and management. In: E.C.Oerke, Gerhards, R., Menz, G., Sikora, R. (Eds.), *Precision crop protection - the challenge and use of heterogeneity*. Springer, Heidelberg, 205–232.
- Bareth, G., Bubenzer, O., 2009. Z2: Data Management and Data Services. CRC 806 proposal for the first funding phase 2009-2013, ed. by W. Schuck, 495–511.
- Bareth, G., Bubenzer, O., 2013. Z2: Data Management and Data Services. CRC 806 proposal for the 2nd funding phase 2013-2017, ed. by W. Schuck, 365–376.
- Bartelme, N., 2005. *Geoinformatik*. Springer.
- Bartling, S., Friesike, S., 2014. *Opening Science – The Evolving Guide on How the Internet is Changing Research, Collaboration and Scholarly Publishing*. Springer International Publishing.
- Bateson, G., 1987. *Geist und Natur: Eine notwendige Einheit*. Suhrkamp Taschenbuch Wissenschaft.
- Battersby, S. E., Finn, M. P., Usery, E. L., Yamamoto, K. H., 2014. Implications of Web Mercator and Its Use in Online Mapping. *Cartographica: The International Journal for Geographic Information and Geovisualization* 49 (2), 85–101. eprint: <http://dx.doi.org/10.3138/cart0.49.2.2313>.
- Baumann, P., 2012. *OGC® WCS 2.0 Interface Standard- Core*. Open Geospatial Consortium. <http://www.opengeospatial.org/standards/wcs>,
- Beath, C., Becerra-Fernandez, I., Ross, J., Short, J., 2012. Finding value in the information explosion. *MIT Sloan Management Review* 53 (4), 18.
- Becker, D., Verheul, J., Zickel, M., Willmes, C., 2015. LGM paleoenvironment of Europe - Map. CRC806-Database. 10.5880/SFB806.15.
- Bendig, J., Bolten, A., Bennertz, S., Broscheit, J., Eichfuss, S., Bareth, G., 2014. Estimating Biomass of Barley Using Crop Surface Models (CSMs) Derived from UAV-Based RGB Imaging. *Remote Sensing* 6 (11), 10395. issn: 2072-4292.
- Bergmann, S., 2016. PHPUnit testing framework. (<https://phpunit.de/>, accessed: 2016-01-23).
- Bernal, J. D., 1939. *The Social Function of Science*. George Routledge & Sons Ltd.

- Berners-Lee, T., Masinter, L., McCahill, M., 1994. RFC1738 Uniform Resource Locators (URL). IETF. <https://www.ietf.org/rfc/rfc1738.txt>,
- Berners-Lee, T., 1989. Information Management: A Proposal. Tech. rep. CERN. <http://www.w3.org/History/1989/proposal.html>,
- Berners-Lee, T., 2000. Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web. HarperBusiness.
- Berners-Lee, T., Hendler, J., Lassila, O., 2001. The Semantic Web. *Scientific American* 284 (5), 34–43.
- Birkenbihl, K., 2006. Standards für das Semantic Web. In: Pellegrini, T., Blumauer, A. (Eds.), *Semantic Web - Wege zur vernetzten Wissensgesellschaft*. Springer-Verlag, 73–88.
- Bizer, C., Heath, T., Berners-Lee, T., 2009. Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems* 5 (3), 1–22.
- Blaha, M., 2010. Patterns of data modeling. CRC Press, Taylor & Francis Group, Boca Raton, FL.
- Blumauer, A., Pellegrini, T., 2006. Semantic Web und semantische Technologien: Zentrale Begriffe und Unterscheidungen. In: Pellegrini, T., Blumauer, A. (Eds.), *Semantic Web - Wege zur vernetzten Wissensgesellschaft*. Springer Verlag, Berlin Heidelberg. 9–25.
- Boddy, D., Boonstra, A., Kennedy, G., 2005. *Managing Information Systems: an Organizational Perspective*. Prentice Hall.
- Bohle, S., 2013. What is E-science and How Should it be Managed? (accessed: 13.07.2015). http://www.scilogs.com/scientific_and_medical_libraries/what-is-e-science-and-how-should-it-be-managed/,
- Böhner, U., Schyle, D., 2006. radiocarbon CONTEXT database. <http://context-database.uni-koeln.de/>, 10.1594/GFZ.CONTEXT.Ed1.
- Bolten, A., Hoffmeister, D., Willmes, C., Bareth, G., 2010. Adapting the Database Concept of the CRC/TR 32 "SVA-Patterns" to CRC 806 "Our Way to Europe". In: Curdt, C., Bareth, G. (Eds.), *Proceedings of the Data Management Workshop. 29.-30.10.2009, University of Cologne. Vol. 90. Kölner Geographische Arbeiten. Geographisches Institut, Universität zu Köln*, 7–12.
- Boslaugh, S., Watters, P. A., 2008. *Statistics in a Nutshell*. O'Reilly Media, Inc., Sebastopol, CA.
- Boundless Inc., 2014. OpenGeo Suite plugin for QGIS. (<http://qgis.boundlessgeo.com/static/docs/intro.html>, accessed: 2015-04-09). Boundless Inc.
- Boundless Inc., 2015. OpenGeo Explorer QGIS plugin. (accessed: 2015-08-24). <http://qgis.boundlessgeo.com/>,
- Braden, R., 1989. Requirements for Internet Hosts – Communication Layers. (Request for Comments: 1122). Internet Engineering Task Force. <http://tools.ietf.org/html/rfc1122>,
- Bradt Möller, M., Pastoors, A., Slizewski, A., Weniger, G.-C., 2010. NESPOS- A digital archive and platform for Pleistocene archaeology. *Proceedings of the data management workshop*, ed. by C. Curdt, G. Bareth. Vol. 90. *Kölner Geographische Arbeiten. University of Cologne*, 13–18.
- Brickley, D., Corlosquet, S., Douglas, J., R.V.Guha, Goto, S., Holland, V. T., Hepp, M., Jiang, C., Johnson, J., Macbeth, S., Mika, P., Mishra, G., Peterson, S., Shubin, A., Tikhokhod, Y., Antonov, E., Nevile, C., Coughran, B., Grossmeier, G., Sandhaus, E., Chopra, A., Swick, R., Cerf, V., Wallis, R., 2016. Schema.org collaborative vocabulary. (<http://schema.org/>, accessed: 2016-01-28).
- Brocks, S., Jungkunst, H. F., Bareth, G., 2014. A regionally disaggregated inventory of nitrous oxide emissions from agricultural soils in Germany – a GIS-based empirical approach. *Erdkunde* 68 (2), 125–144.
- Brown, G. O., 2003. Out of the Way. *PLoS Biol.* 1 (1).
- Brunt, J. W., 2009. Data Management Principles, Implementation and Administration. In: Michener, W. K., Brunt, J. W. (Eds.), *Ecological Data: Design, Management and Processing*. John Wiley & Sons. 25–42.

- Buckland, M., 1998. What is a 'document'? *Journal of the American Society for Information Science* 48 (9), 804–809.
- Budhatoki, N. R., Nedovic-Budic, Z., 2007. Expanding the Spatial Data Infrastructure Knowledge Base. In: Onsrud, H. (Ed.), *Research and Theory in Advancing Spatial Data Infrastructure Concepts*. ESRI Press, Redlands, CA, USA, 7–31.
- Butler, H., Daly, M., Doyle, A., Gillies, S., Schaub, T., Schmidt, C., 2008. The GeoJSON Format Specification. (accessed: 2015-09-22). <http://geojson.org/geojson-spec.html>,
- Callaghan, S., Donegan, S., Pepler, S., Thorley, M., Cunningham, N., Kirsch, P., Ault, L., Bell, P., Bowie, R., Leadbetter, A., Lowry, R., Moncoiffé, G., Harrison, K., Smith-Haddon, B., Weatherby, A., Wright, D., 2012. Making Data a First Class Scientific Output: Data Citation and Publication by NERC's Environmental Data Centres. *The International Journal of Digital Curation* 7 (1), 107–113.
- Carusi, A., Reimer, T., 2010. Virtual Research Environment - Collaborative Landscape Study. Tech. rep. JISC. <http://www.jisc.ac.uk/media/documents/pub/vrelandscapereport.pdf>,
- Chacon, S., Straub, B., 2015. *Pro Git – Everything you need to know about Git*. apress, Mountain View, CA.
- Chalmers, R., Grangel, R., 2008. Methodology for the implementation of knowledge management systems. *Journal of the American Society for Information Science and Technology* 59 (5), 742–755. issn: 1532-2890.
- Checkland, P., Holwell, S., 1998. *Information, Systems, and Information Systems: Making Sense of the Field*. John Wiley & Sons, Chichester, West Sussex, UK.
- Christl, A., 2014. OSGeo - Professionally leveraging Open Source. FIG Congress 2014 - Engaging the Challenges, Enhancing the Relevance, Kuala Lumpur, Malaysia. http://www.fig.net/resources/proceedings/fig_proceedings/fig2014/papers/TS08H/TS08H_christl_7371.pdf,
- Clark, T., 2005. *Storage Virtualization: Technologies for Simplifying Data Storage and Management*. Addison-Wesley Professional.
- Creative Commons, 2015. About The Licenses. (accessed: 2015-09-02). Creative Commons. <http://creativecommons.org/licenses/>,
- Creative Commons, 2016. Creative Commons in Science. (accessed: 2016-01-24). Creative Commons. <http://creativecommons.org/science/>,
- Creative Commons Developers, 2016. Creative Commons Rights Expression Language. (<http://creativecommons.org/ns>, accessed: 2016-01-28).
- Curdt, C., 2014a. TR32DB Metadata Schema for the Description of Research Data in the TR32DB. CRC/TR32 Database (TR32DB). 10.5880/TR32DB.10.
- Curdt, C., Bareth, G., eds. 2009. *Proceedings of the Data Management Workshop. 29. - 30.10.2009*. Kölner Geographische Arbeiten 90. Geographisches Institut der Universität zu Köln, Köln.
- Curdt, C., Hoffmeister, D., Jekel, C., Brocks, S., Waldhoff, G., Bareth, G., 2011. TR32DB - Management and visualization of heterogeneous scientific data. *Proceedings of the 19th International Conference on Geoinformatics, Shanghai, China*.
- Curdt, C., Hoffmeister, D., Jekel, C., Udelhoven, K., Waldhoff, G., Bareth, G., 2010. Implementation of a centralized data management system for the CRC Transregio 32 "Patterns in soil-vegetation-atmosphere-systems". *Proceedings of the Data Management Workshop, 29.-30.10.2010*, ed. by C. Curdt, G. Bareth. Vol. 90. Kölner Geographische Arbeiten. University of Cologne, 27–33.
- Curdt, C., Hoffmeister, D., Waldhoff, G., Bareth, G., 2008. SPATIAL DATA INFRASTRUCTURE FOR SOIL-VEGETATION-ATMOSPHERE MODELLING: SET-UP OF A SPATIAL DATABASE FOR A RESEARCH PROJECT (SFB/TR32). *The International Archives of the Photogrammetry, Re-*

- mote Sensing and Spatial Information Sciences. Vol. XXXVII. Part B4, ISPRS. Beijing, China, 131–136.
- Curdt, C., 2014b. Design and Implementation of a Research Data Management System: The CRC/TR32 Project Database (TR32DB). PhD thesis, University of Cologne.
- Curdt, C., Hoffmeister, D., 2015. Research data management services for a multidisciplinary, collaborative research project: design and implementation of the TR32DB project database. *Program* 49 (4), null. eprint: <http://www.emeraldinsight.com/doi/pdf/10.1108/PROG-02-2015-0016>.
- Curdt, C., Hoffmeister, D., Waldhoff, G., Jekel, C., Bareth, G., 2012. Scientific Research Data Management for Soil-Vegetation-Atmosphere Data - The TR32DB. *International Journal of Digital Curation* 7 (2), 68–80.
- Dallmeier-Thiessen, S., 2011. Strategien bei der Veröffentlichung von Forschungsdaten. In: Bütner, S., Hobohm, H.-C., Müller, L. (Eds.), *Handbuch Forschungsdatenmanagement*. Bock und Herchen, Bad Honnef, Germany, 157–168.
- D’Arcus, B., Giasson, F., 2009. Bibo - Bibliographic Ontology Specification. (accessed: 2015-09-04). The Bibliographic Ontology. <http://bibliontology.com/>,
- DataCite Metadata Working Group, 2011. DataCite Metadata Schema for the Publication and Citation of Research Data. DataCite.
- David, P. A., 2004. Understanding the emergence of ‘open science’ institutions: functionalist economics in historical context. *Industrial and Corporate Change* 13 (4), 571–589. eprint: <http://icc.oxfordjournals.org/content/13/4/571.full.pdf+html>.
- Davies, J., Studer, R., Warren, P., 2006. *Semantic Web Technologies - trends and Research in Ontology-based Systems*. John Wiley & Sons, Ltd, Chichester, UK.
- Davis, R., Shrobe, H., Szolovits, P., 1993. What is a Knowledge Representation? *AI Magazine* 14 (1), 17–33.
- DCC, 2015. What is digital curation? (accessed: 2015-07-31). Digital Curation Centre. <http://www.dcc.ac.uk/digital-curation/what-digital-curation>,
- DDI Alliance, 2015. DDI Lifecycle v3.2 - A metadata specification for the social and behavioral sciences. (accessed: 2015-07-31). Data Documentation Initiative. <http://www.ddialliance.org/>,
- De Dauw, J., 2015. Semantic Maps MediaWiki Extension. (accessed: 2015-08-21). <https://github.com/SemanticMediaWiki/SemanticMaps/>,
- de Geer, S., 1923. On the Definition, Method and Classification of Geography. English. *Geografiska Annaler* 5, 1–37. issn: 16513215.
- de la Beaujardiere, J., 2006. *OpenGIS® Web Map Server Implementation Specification*. Open Geospatial Consortium Inc. <http://www.opengeospatial.org/standards/wms>,
- Dengel, A., 2012. *Semantische Technologien: Grundlagen – Konzepte – Anwendungen*. Spektrum Akademischer Verlag, Berlin, Heidelberg.
- Dennell, R., Roebroeks, W., 2005. An Asian perspective on early human dispersal from Africa. *Nature* 438 (7071), 1099–1104. issn: 0028-0836.
- d’Errico, F., Banks, W. E., Vanhaeren, M., Laroulandie, V., Langlais, M., 2011. PACEA Geo-Referenced Radiocarbon Database. *PaleoAnthropology* 2011, 1–12.
- Devaraju, A., Jirka, S., Kunkel, R., Sorg, J., 2015. Q-SOS—A Sensor Observation Service for Accessing Quality Descriptions of Environmental Data. *ISPRS International Journal of Geo-Information* 4 (3), 1346. issn: 2220-9964.
- DFG, 1998. *Proposals for Safeguarding Good Scientific Practice - Recommendations of the Commission on Professional Self Regulation in Science*. Tech. rep. Deutsche Forschungsgemeinschaft, Weinheim, Germany. http://www.dfg.de/aktuelles_presse/reden_stellungnahmen/download/empfehlung_wiss_praxis_0198.pdf,

- DFG, 2009. Recommendations for Secure Storage and Availability of Digital Primary Research Data. (accessed: 2015-07-31). Deutsche Forschungsgemeinschaft. http://www.dfg.de/download/pdf/foerderung/programme/lis/ua_inf_empfehlungen_200901_en.pdf,
- DFG, 2012. Merkblatt Sonderforschungsbereiche. Tech. rep. (DFG Vordruck 50.06 – 6/12). Deutsche Forschungsgemeinschaft, Bonn. http://www.dfg.de/formulare/50_06/50_06_de.pdf,
- DFG, 2014. Leitfaden für die Antragstellung - Projektanträge, DFG-Vordruck 54.01 - 04/14. Deutsche Forschungsgemeinschaft. http://www.dfg.de/formulare/54_01/54_01_de.pdf,
- DFG, 2015. Leitlinien zum Umgang mit Forschungsdaten. http://www.dfg.de/download/pdf/foerderung/antragstellung/forschungsdaten/richtlinien_forschungsdaten.pdf,
- Domenico, B., 2011. OGC Network Common Data Form (NetCDF) Core Encoding Standard version 1.0. (<http://www.opengis.net/doc/IS/netcdf/1.0>, accessed: 2016-03-16). Open Geospatial Consortium.
- Donnelly, M., 2011. Data management plans and planning. In: Pryor, G. (Ed.), *Managing Research Data*. Facet Publishing, London, UK, 83–104.
- Dublin Core Metadata Initiative, 2004. Dublin core metadata element set, version 1.1: Reference description. <http://dublincore.org/documents/dces/>.
- Dublin Core Metadata Initiative, 2012. *Dublin Core Metadata Element Set, Version 1.1*. (accessed 2015-07-24). <http://dublincore.org/documents/dces/>,
- Duvall, P. M., Matyas, S., Glover, A., 2007. *Continuous Integration - Improving Software Quality and Reducing Risk*. Addison-Wesley.
- ebRIM, 2004. *ISO/TS 15000-3:2004*. International Organization for Standardization. http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=39974,
- Effertz, E., 2010. The funders perspective: Data management in coordinated programmes of the German Research Foundation (DFG). *Proceedings of the Data Management Workshop, 29.–30.10.2010*, ed. by C. Curdt, G. Bareth. Vol. 90. *Kölner Geographische Arbeiten*. University of Cologne, 35–38.
- Elmasri, R., Navathe, S., 2011. *Fundamentals of Database Systems*. Addison Wesley.
- Engelhardt, C., 2013. Forschungsdatenmanagement in DFG-SFBs - Teilprojekte Informationsinfrastruktur (INF-Projekte). *LIBREAS. Library Ideas* (23). Ed. by B. Kaden, M. Kindling, M. Schulz, 106–130.
- Erich, F., Amrit, C., Daneva, M., 2014. Cooperation between information system development and operations: a literature review. *2014 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, ESEM '14, Torino, Italy, September 18-19, 2014*, 69:1. <http://doi.acm.org/10.1145/2652524.2652598>,
- esri, 1998. ESRI Shapefile Technical Description. (<https://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>, accessed: 2016-03-16). Environmental Systems Research Institute, Inc.
- esri, 2016. ArcGIS for Server. (<http://www.esri.com/software/arcgis/arcgisserver>, accessed: 2016-03-28). Environmental Systems Research Institute. <http://www.esri.com/software/arcgis/arcgisserver>,
- European Commission, 2006. *Towards a European e-Infrastructure for e-Science Digital Repositories*. Tech. rep. European Commission. <http://cordis.europa.eu/fp7/ict/e-infrastructure/docs/e-scidr.pdf>,
- European Commission, 2009. *COMMISSION REGULATION (EC) No 976/2009 implementing Directive 2007/2/EC as regards the Network Services*. European Union.
- European Commission, 2013. *Digital science in Horizon 2020*. (http://ec.europa.eu/information_society/newsroom/cf/dae/document.cfm?doc_id=2124, accessed: 2016-03-20). EU Digital Economy & Society.

- European Commission, 2015. Validation of the results of the public consultation on Science 2.0: Science in Transition. Tech. rep. European Union - Research and Innovation. https://ec.europa.eu/research/consultations/science-2.0/science_2_0_final_report.pdf,
- Evans, E., 2004. Domain-driven Design: Tackling Complexity in the Heart of Software. Addison-Wesley.
- Eynden, V. V. den, Corti, L., Woollard, M., Bishop, L., 2009. MANAGING AND SHARING DATA a best practice guide for researchers. Tech. rep. (a best practice guide for researchers). UK Data Archive - University of Essex, Essex.
- Faundeen, J. L., Burley, T. E., Carlino, J. A., Govoni, D. L., Henkel, H. S., Holl, S. L., Hutchison, V. B., Martín, E., Montgomery, E. T., Ladino, C., Tessler, S., Zolly, L. S., 2013. The United States Geological Survey Science Data Lifecycle Model. U.S. Geological Survey (USGS). <http://pubs.er.usgs.gov/publication/ofr20131265>, 10.3133/ofr20131265.
- Fensel, D., Lausen, holger, Bruijn, J. de, Roman, D., 2007. Enabling Semantic Web Services - The Web Service Modeling Ontology. Springer-Verlag, Heidelberg, Berlin.
- FGDC, 1998. *Content Standard for Digital Geospatial Metadata*. Federal Geographic Data Committee. <http://www.fgdc.gov/metadata/csdlm/>,
- Fielding, R., Irvine, U. C., Gettys, J., Mogul, J., Frystyk, H., Leach, L. M. M., Berners-Lee, T., 1999. Hypertext Transfer Protocol – HTTP/1.1. IETF. <https://www.ietf.org/rfc/rfc2616.txt>,
- Foertsch, R., Datenhaltung und Storage. (accessed: 2015-08-06). Institute of Archeology, University of Cologne. <http://archaeologie.uni-koeln.de/node/141>,
- Fowler, M., 2006. Continuous Integration. ThoughtWorks. <http://www.martinfowler.com/articles/continuousIntegration.html>,
- Fox, P., Hendler, J., 2009. “The Fourth Paradigm: Data-Intensive Scientific Discovery”. In: ed. by T. Hey, S. tansley, K. Tolle. Microsoft Research, Redmond, WA. Chap. Semantic eScience: encoding meaning in nextgeneration digitally enhanced science, 147–152.
- Fraser, M., 2005. Virtual research environments: overview and activity. *Ariadne* (44).
- Frické, M., 2009. The knowledge pyramid: a critique of the DIKW hierarchy. *Journal of Information Science* 35 (2), 131–142. eprint: <http://jis.sagepub.com/content/35/2/131.full.pdf+html>.
- Friesike, S., 2014. Creative Commons Licences. In: *Opening Science – The Evolving Guide on How the Internet is Changing Research, Collaboration and Scholarly Publishing*. Springer Open, Heidelberg, 287–289.
- FSF, 2016. GNU Wget. (<https://www.gnu.org/software/wget/>, accessed: 2016-03-28). Free Software Foundation.
- Galiegue, F., Zyp, K., Court, G., 2013. JSON Schema: core definitions and terminology. (accessed: 2015-08-17). Internet Engineering Task Force. <http://tools.ietf.org/html/draft-zyp-json-schema-04>,
- Galinski, C., 2006. Wozu Normen? Wozu semantische Interoperabilität? In: Pellegrini, T., Blumauer, A. (Eds.), *Semantic Web - Wege zur vernetzten Wissensgesellschaft*. Springer-Verlag, 47–72.
- Gebhardt, H., Glaser, R., Radtke, U., Reuber, P., 2011. *Geographie - Physische und Humangeographie*. Spektrum, Akademischer Verlag, Heidelberg.
- General Bathymetric Chart of the Oceans, 2014. GEBCO 2014 Grid - Gridded bathymetry data. (accessed: 2015.10.22, http://www.gebco.net/data_and_products/gridded_bathymetry_data/). Hosted by the British Oceanographic Data Centre (BODC). http://www.gebco.net/data_and_products/gridded_bathymetry_data/,
- GeoDjango Contributors, 2014. GeoDjango – A world-class geographic web framework. (2014-12-18). Django Software Foundation. <http://geodjango.org/>,
- GeoExt Contributors, 2014. GeoExt – JavaScript Toolkit for Rich Web Mapping Applications. (2014-12-18). GeoExt Community. <http://geoext.org>,

- GeoNetwork Contributors, 2015. GeoNetwork opensource. (<http://geonetwork-opensource.org/>, accessed: 2016-01-31). OpenSource Geospatial Foundation (OSGeo).
- GeoNode Contributors, 2014. GeoNode–Open Source Geospatial Content Management System. (2014-12-18). <http://geonode.org>,
- GeoServer Contributors, 2014. GeoServer – open source server for sharing geospatial data. (2014-12-18). Open Source Geospatial Foundation. <http://geoserver.org>,
- Giorgetta, M. A., Jungclaus, J., Reick, C. H., Legutke, S., Bader, J., Böttinger, M., Brovkin, V., Crueger, T., Esch, M., Fieg, K., Glushak, K., Gayler, V., Haak, H., Hollweg, H.-D., Ilyina, T., Kinne, S., Kornblueh, L., Matei, D., Mauritsen, T., Mikolajewicz, U., Mueller, W., Notz, D., Pithan, F., Raddatz, T., Rast, S., Redler, R., Roeckner, E., Schmidt, H., Schnur, R., Segschneider, J., Six, K. D., Stockhause, M., Timmreck, C., Wegner, J., Widmann, H., Wieners, K.-H., Claussen, M., Marotzke, J., Stevens, B., 2013. Climate and carbon cycle changes from 1850 to 2100 in MPI-ESM simulations for the Coupled Model Intercomparison Project phase 5. *Journal of Advances in Modeling Earth Systems* 5 (3), 572–597. issn: 1942-2466.
- GitLab Community, 2015. GitLab - Create, review and deploy code together. (accessed: 2015-08-07). GitLab B.V. <https://about.gitlab.com/>,
- Goldfarb, B., Henrekson, M., 2003. Bottom-up versus top-down policies towards the commercialization of university intellectual property. *Research Policy* 32 (4).
- Goodchild, M., 1992. Geographical information science. *International Journal of Geographical Information Systems* 6 (1), 31–45.
- Goodchild, M. F., 2013. Prospects for a Space-Time GIS. *Annals of the Association of American Geographers* 103 (5), 1072–1077. eprint: <http://www.tandfonline.com/doi/pdf/10.1080/00045608.2013.792175>.
- Google Inc., 2014. AngularJS – Superheroic JavaScript MVW Framework. (2014-11-06). Google Inc. <https://angularjs.org/>,
- Gray, J., Szalay, A. S., Thakar, A. R., Stoughton, C., vandenBerg, J., 2002. *Online scientific data curation, publication, and archiving*. <http://dx.doi.org/10.1117/12.461524>,
- Grobe, H., Schumacher, S., Sieger, R., Schäfer-Neth, C., Schindler, U., Diepenbroek, M., 2015. PANGAEA - Data Publisher for Earth & Environmental Science. (<http://www.pangaea.de/> - accessed: 2016-01-20).
- Gruber, T., 1993. A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition* 5, 199–220.
- GSDI, 2009. Spatial Data Infrastructure Cookbook. Global Spatial Data Infrastructure Association (GSDI).
- Gutteridge, C., Dutton, A., Smith, A., 2013. Open Organisations Vocabulary. (<http://purl.org/openorg/>, accessed: 2016-01-28.).
- Haidle, M., Richter, J., 2012. Frühe menschliche Expansionen – mehr als räumliche Ausbreitungen. *Archäologie in Deutschland* 2012 (4), 20–21.
- Hare, J., Sinclair, P., Lewis, P., Martinez, K., Enser, P., Sandom, C., 2006. Bridging the semantic gap in multimedia information retrieval: top-down and bottom-up approaches. *Mastering the Gap: From Information Extraction to Semantic Representation / 3rd European Semantic Web Conference*, Budva, Montenegro.
- Heimler, S., 2014. Semantic MediaWiki Model Development through Object-oriented JSON Schema. SMW CON FALL 2014, VIENNA, <http://fannon.de/p/mobo-paper.pdf>,
- Heimler, S., 2015a. Mobo Documentation. (accessed: 2015-08-19). <http://fannon.gitbooks.io/mobo-documentation/>,
- Heimler, S., 2015b. Schema-Driven development of Semantic MediaWikis. MA thesis, University of Applied Sciences Augsburg.
- Hey, A. J., Trefethen, A. E., 2003. The data deluge: An e-science perspective.

- Hey, T., Payne, M. C., 2015. Open science decoded. *Nat Phys* 11 (5), 367–369. issn: 1745-2473.
- Hey, T., Tansley, S., Tolle, K., 2009. *The Fourth Paradigm: Data-Intensive Scientific Discovery*. Microsoft Research.
- Hey, T., Trefethen, A. E., 2005. Cyberinfrastructure for e-Science. *Science* 308 (5723), 817–821. eprint: <http://www.sciencemag.org/content/308/5723/817.full.pdf>.
- Hider, P., 2012. *Information Resource Description - Creating and managing Metadata*. Facet Publishing, London, UK.
- Hitzler, P., Krötsch, M., Rudolph, S., 2010. *Foundations of Sematic Web Technologies*. CRC Press, Taylor & Francis Group, Boca Raton.
- Hoffmeister, D., Ntageretzi, K., Aasen, H., Curdt, C., Hadler, H., Willershäuser, T., Bareth, G., Brückner, H., Vött, A., 2014. 3D model-based estimations of volume and mass of high-energy dislocated boulders in coastal areas of Greece by terrestrial laser scanning. *Zeitschrift für Geomorphologie, Supplementary Issues* 58 (3), 115–135.
- Horne, D. J., Holmes, J. A., Rodriguez-Lazaro, J., Viehberg, F. A., 2012. Chapter 18 - Ostracoda as Proxies for Quaternary Climate Change: Overview and Future Prospects. In: David J. Horne Jonathan A. Holmes, J. R.-L., Viehberg, F. A. (Eds.), *Ostracoda as Proxies for Quaternary Climate Change*. Vol. 17. *Developments in Quaternary Sciences*. Elsevier, 305–315.
- Huvila, I., 2012. Being Formal and Flexible: Semantic Wiki as an Archaeological e-Science Infrastructure. *Revive the Past. Computer Applications and Quantitative Methods in Archaeology (CAA)*, ed. by M. Zhou, I. Romanowska, Z. Wu, P. Xu, P. Verhagen. *Proceedings of the 39th International Conference, Beijing*. Pallas Publications, Amsterdam, 186–197.
- IDF, 2015. The DOI® System. (accessed: 2015-08-04). International DOI Foundation. <http://www.doi.org/>,
- ISO, 2012. *ISO 26324 — Digital object identifier system*. International Organization for Standardization. <https://www.iso.org/obp/ui/#iso:std:43506:en>,
- ISO, 2015. Standards. (accessed: 2015-08-11). International Standards Organization. <http://www.iso.org/iso/home/standards.htm>,
- ISO19115-1, 2014. *Geographic information – Metadata – Part 1: Fundamentals*. International Organization for Standardization. http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=53798,
- Jashapara, A., 2005. *Knowledge Management: an Integrated Approach*. Prentice Hall, Harlow.
- Jenkins Contributors, 2016. *Jenkins - An extendable open source automation server*. (<https://jenkins-ci.org/>, accessed: 2016-02-24).
- Jennex, M., 2009. Re-Visiting the Knowledge Pyramid. *System Sciences, 2009. HICSS '09. 42nd Hawaii International Conference on*, 1–7.
- Jessup, L. M., Valacich, J. S., 2005. *Information Systems Today*. Prentice Hall, Upper Saddle River, NJ.
- Jong, S., Slavova, K., 2014. When publications lead to products: The open science conundrum in new product development. *Research Policy* 43 (4), 645–654. issn: 0048-7333.
- JRC, 2016. *JRC Data Catalogue – making open science a reality*. (<https://ec.europa.eu/jrc/en/news/jrc-data-catalogue-making-open-science-reality>, accessed: 2016-03-24).
- Kääriäinen, J., Välimäki, A., 2009. “Software Process Improvement: 16th European Conference, EuroSPI 2009, Alcalá (Madrid), Spain, September 2-4, 2009. Proceedings”. In: ed. by R. V. O’Connor, N. Baddoo, J. Cuadrado Gallego, R. Rejas Muslera, K. Smolander, R. Messnarz. Springer Berlin Heidelberg, Berlin, Heidelberg. Chap. *Applying Application Lifecycle Management for the Development of Complex Systems: Experiences from the Automation Industry*, 149–160. http://dx.doi.org/10.1007/978-3-642-04133-4_13,
- Kasrtens, K., 2012. *Is the Fourth Paradigm Really New?* (accessed 13.07.2015). <http://serc.carleton.edu/earthandmind/posts/4thparadigm.html>,

- Kauppinen, T., Baglatzi, A., Keßler, C., 2012. Linked science: interconnecting scientific assets. Data Intensive Science. CRC Press, USA (forthcoming 2012).
- Kauppinen, T., Espindola, G. M. de, 2011. Linked Open Science-Communicating, Sharing and Evaluating Data, Methods and Results for Executable Papers. *Procedia Computer Science* 4. (Proceedings of the International Conference on Computational Science, {ICCS} 2011), 726–731. issn: 1877-0509.
- Kienreich, W., Strohmaier, M., 2006. Wissensmodellierung - Basis für die Anwendung semantischer Technologien. In: Pellegrini, T., Blumauer, A. (Eds.), *Semantic Web - Wege zur vernetzten Wissensgesellschaft*. Springer Berlin & Heidelberg.
- Klump, J., Bertelmann, R., 2013. Forschungsdaten. In: Kuhlen, R., Semar, W., Strauch, D. (Eds.), *Grundlagen der praktischen Information und Dokumentation*. De Gruyter Saur, Munich, Germany, 575–583.
- Klyne, G., Carroll, J. J., 2004. Resource Description Framework (RDF): Concepts and Abstract Syntax. Tech. rep. (Online: <http://www.w3.org/TR/rdf-concepts/>). W3C. <http://www.w3.org/TR/rdf-concepts/>,
- Koren, Y., 2012. Working with Mediawiki. WikiWorks Press, San Bernardino, CA.
- Koren, Y., 2015. DataTransfer MediaWiki Extension. (accessed: 2015-08-21). https://www.mediawiki.org/wiki/Extension:Data_Transfer,
- Koren, Y., Gambke, S., 2015a. Semantic Forms MediaWiki Extension. (accessed: 2015-08-21). https://www.mediawiki.org/wiki/Extension:Semantic_Forms,
- Koren, Y., Kong, J. H., Gambke, S., Dauw, J. D., 2015b. Semantic Result Formats Semantic MediaWiki Extension. (accessed: 2015-08-21). https://semantic-mediawiki.org/wiki/Semantic_Result_Formats,
- Korzybski, A., 1933. *Science and Sanity: An Introduction to Non-Aristotelian Systems and General Semantics*. Inst of General Semantics.
- Kralidis, T., Tzotsos, A., 2014. pyCSW-Metadata Publishing Just Got Easier. <http://pysw.org>,
- Kratz, J., Strasser, C., 2014. Data publication consensus and controversies [v3; ref status: indexed, <http://f1000r.es/4ja>]. *F1000Research* 3.
- Krötzsch, M., Vrandečić, D., Völkel, M., 2006. Semantic MediaWiki. English. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L. (Eds.), *The Semantic Web - ISWC 2006*. Vol. 4273. *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 935–942.
- Krötzsch, M., Vrandečić, D., Völkel, M., Haller, H., Studer, R., 2007. Semantic Wikipedia. *Journal of Web Semantics* 5 (4), 251–261.
- Kudryashov, K., 2016. Behat BDD framework for PHP. (<http://behat.org/>, accessed: 2016-02-24).
- Kurose, J., Ross, K., 2010. *Computer Networking: A Top-Down Approach*. Pearson Education, Limited.
- Lenzerini, M., 2002. Data Integration: A Theoretical Perspective. *Proceedings of the Twenty-first ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS '02*. ACM, Madison, Wisconsin, 233–246. <http://doi.acm.org/10.1145/543613.543644>,
- Lindberg, H., Rydin, P., 2002. Model view controller. (US Patent App. 09/768,389). <http://www.google.com/patents/US20020143800>,
- Lobacher, P., 2014. *Typo3 Extbase - Moderne Extension-Entwicklung für Typo3 CMS mit Extbase und Fluid*. Open Source Press.
- Longley, P. A., Goodchild, M. F., Maguire, D. J., Rhind, D. W., 2005. *Geographical Information Systems and Science*. John Wiley & Sons, Ltd., Chichester, UK.
- Lubas, R. L., Jackson, A. S., Schneider, I., 2013. *The Metadata Manual*. Chandos Information Professional Series. Chandos Publishing, Oxford, UK.

- Ludwig, J., Enke, H., 2013. Leitfaden zum Forschungsdaten-Management: Handreichungen aus dem WissGrid-Projekt. Verlag Werner Hülsbusch.
- Ma, X., Fox, P., Narock, T., Wilson, B., 2015. Semantic e-Science. *Earth Science Informatics* 8 (1), 1–3. issn: 1865-0473.
- Maali, F., Erickson, J., Archer, P., 2014. Data Catalog Vocabulary (DCAT). (accessed: 2015-08-19). World Wide Web Consortium. <http://www.w3.org/TR/vocab-dcat/>,
- MapServer Contributors, 2014. MapServer – Open source web mapping. (2014-12-18). Open Source Geospatial Foundation. <http://mapserver.org>,
- Märker, M., Hochschild, Z. K. K., 2009. Multidisciplinary Integrative Georelational Database for Spatio-Temporal Analysis of Expansion Dynamics of Early Humans. Proceedings of CAA, Williamsburg, Virginia, USA. 18.-22.03, ed. by B. Frischer, G. Guidi.
- Märker, M., Kanaeva, Z., Quenéhervé, G., Hochschild, V., Bolus, M., Bruch, A. A., Conard, N. J., Haidle, M. N., Hertler, C., Kandel, A. W., Mosbrugger V. & Schrenk, F., 2013. ROAD: A georelational database with WebGIS functionalities for the assessment of the Role of Culture in early expansion of humans. *Journal of Archaeological Science*.
- Märker, M., Willmes, C., Hochschild, V., Bareth, G., 2014. How to exchange data between DB Systems on Early Humans. A case study based on the SFB 806 DB and the ROCEEH ROAD system. 2nd Workshop on Data Management, CRC806-Database, 2nd Data Management Workshop, Cologne. <http://dx.doi.org/10.5880/SFB806.11>,
- Max-Planck-Gesellschaft, 2003. Berlin Declaration on Open Access to Knowledge in the Sciences and Humanities. (accessed: 2015-09-13). Max-Planck-Gesellschaft. <http://openaccess.mpg.de/Berliner-Erklaerung>,
- Mellars, P., Stringer, C., 1989. *The Human Revolution: Behavioural and Biological Perspectives on the Origins of Modern Humans*. Edinburgh University Press.
- Merriam Webster, 2015. Full Definition of LIFE CYCLE. (accessed: 2015-09-14). Merriam Webster an Encyclopedia Britannica Company. <http://www.merriam-webster.com/dictionary/life%20cycle>,
- Miller S., J., 2011. *Metadata for digital collections : a how-to-do-it manual*. Neal-Schuman Publishers, Inc., New York, USA.
- Mockapetris, P., 1987. DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION. IETF. <https://www.ietf.org/rfc/rfc1035.txt>,
- Naumann, J. D., Jenkins, A. M., 1982. Prototyping: The New Paradigm for Systems Development. *English. MIS Quarterly* 6 (3), issn: 02767783.
- Navigli, R., Velardi, P., 2008. From Glossaries to Ontologies: Extracting Semantic Structure from Textual Definitions. In: Buitelaar, P., Cimiano, P. (Eds.), *Ontology Learning and Population: Bridging the Gap between Text and Knowledge*. Vol. 167. *Frontiers in Artificial Intelligence and Applications*. IOS Press, Amsterdam, 71–90.
- Nebert, D., Whiteside, A., Vretanos, P., 2007. *OpenGIS® Catalogue Services Specification*. Open Geospatial Consortium Inc. <http://www.opengeospatial.org/standards/cat>,
- Nelson, B., 2009. Empty archives. *Nature* 461 (7261), 160–163.
- Nelson, T., 1990. On the Xanadu project. *BYTE Magazine* 15 (9), 298–299.
- Nielsen, J., 1999. *Designing Web Usability: The Practice of Simplicity*. New Riders Publishing, Thousand Oaks, CA, USA.
- NISO, 2004. Registry for the OpenURL Framework - ANSI/NISO Z39.88-2004. (accessed: 2015-08-04). National Information Standard Organization. http://www.openurl.info/registry/docs/implementation_guidelines/,
- NISO, 2010. *Z39.84-2005 (R2010) Syntax for the Digital Object Identifier*. Baltimore, Maryland: National Information Standards Organization. www.niso.org/standards/z39-84-2005/,

- Nogueras-Iso, J., Muro-Medrano, P., Zarazaga-Soria, F., 2005. Geographic Information Metadata for Spatial Data Infrastructures: Resources, Interoperability and Information Retrieval. Springer London, Limited.
- Norris, J., 2015. Future trends in geospatial information management: the five to ten year vision - Second edition. Tech. rep. UN Committee of Experts on Global Geospatial Information Management (UN-GGIM). http://ggim.un.org/docs/UN-GGIM-Future-trends_Second%20edition.pdf,
- North, K., Kumta, G., 2014. Knowledge Management : Value Creation Through Organizational Learning. Springer International Publishing.
- NSF, 2011. Data Management Plan Requirements. National Science Foundation. <http://www.nsf.gov/eng/general/dmp.jsp>,
- OGC, 2015a. The OGC's Role in Government & Spatial Data Infrastructure. (accessed: 2015-08-25). Open Geospatial Consortium Inc. http://www.opengeospatial.org/domain/gov_and_sdi,
- OGC, 2015b. The Open Geospatial Consortium. (accessed: 2015-08-24). <http://www.opengeospatial.org/>,
- OKFN, 2016. Open Definition 2.1. (<http://opendefinition.org/od/2.1/en/>, accessed: 2016-03-15). The Open Knowledge Foundation.
- Olanoff, D., 2011. Ship it faster and cheaper – GitLab is GitHub for your own servers. The Next Web. (accessed: 2015-08-18).
- Open Knowledge Foundation, 2014. CKAN – The open source data portal software. (2014-10-30). CKAN Association. <http://ckan.org>,
- Open Knowledge Foundation, 2016. CKAN API documentation. (<http://docs.ckan.org/en/ckan-2.3.2/api/index.html>, accessed: 2016-01-27). CKAN Association.
- Open Refine Contributors, 2016. Open Refine - A free, open source, powerful tool for working with messy data. (<http://openrefine.org/>, accessed: 2016-01-07). <http://openrefine.org/>,
- Open Source Matters, Inc., 2015. Joomla - Open Source CMS. (accessed: 2015-09-01). Open Source Matters, Inc. <http://www.joomla.org/>,
- OpenLayers Contributors, 2015. OpenLayers - A high-performance, feature-packed library for all your mapping needs. Open Source Geospatial Foundation (OSGeo). <http://openlayers.org/>,
- OpenLink Software, 2015. OpenLink Virtuoso. (accessed: 2015-08-18). OpenLink Software. <http://virtuoso.openlinksw.com/>,
- Oracle Inc., 2015. MySQL - The world's most popular open source database. (accessed: 2015-08-31). Oracle Corporation and/or its affiliates. <http://www.mysql.com/>,
- Oracle Inc., 2016. Java Software. (<https://www.oracle.com/java/index.html>, accessed: 2016-03-07). Oracle Inc.
- Oregon State University Libraries, 2015. Data Management Druing the Reserach Lifecycle. (accessed: 2015-07-30). Oregon Stae University. <http://guides.library.oregonstate.edu/lifecycle>,
- OSI, 2007. The Open Source Definition. (<http://opensource.org/docs/osd>, accessed: 2016-03-06). The Open Source Initiative.
- Oßwald, A., Scheffel, R., Neuroth, H., 2012. Langzeitarchivierung von Forschungsdaten Einführende Überlegungen. In: Neuroth, H., Strathmann, S., Oßwald, A., Scheffel, R., Klump, J., Ludwig, J. (Eds.), Langzeitarchivierung von Forschungsdaten: Eine Bestandsaufnahme. Verlag Werner Hülsbusch, Boizenburg, Germany, 13–23.
- ownCloud Contributors, 2016. ownCloud - open source priovate cloud. (<https://owncloud.org/>, accesde: 2016-03-07). ownCloud.

- Patashnik, O., 1988. Designing BibTeX styles. (<http://www.pctex.com/files/managed/a/a3/btxhak.pdf> - accessed: 2015-05-11).
- Peel, M. C., Finlayson, B. L., McMahon, T. A., 2007. Updated world map of the Köppen-Geiger climate classification. *Hydrology and Earth System Sciences* 11 (5), 1633–1644.
- Pitman, W. C., Heirtzler, J. R., 1966. Magnetic Anomalies over the Pacific-Antarctic Ridge. *Science* 154 (3753), 1164–1171. eprint: <http://www.sciencemag.org/content/154/3753/1164.full.pdf>.
- PIWIK Contributors, 2014. PIWIK - Open Analytics Platform. (2014-12-18). <http://piwik.org/>,
- Pontika, N., Knoth, P., Cancellieri, M., Pearce, S., 2015. Fostering open science to research using a taxonomy and an eLearning portal. *Proceedings of the 15th International Conference on Knowledge Technologies and Data-driven Business*, ACM, 11.
- Poole, D., Mackworth, A., 2010. *Artificial Intelligence - Foundation of computational agents*. Cambridge University Press.
- PostgreSQL Contributors, 2015. PostgreSQL - Worlds most advanced open source database. The PostgreSQL Global Development Group. <http://www.postgresql.org/>,
- Pratas, A., 2014. *Creating Flat Design Websites*. Packt Publishing.
- Pryor, G., 2012. *Managing Research Data*. Facet Publishing.
- Python Software Foundation, 2016. Python programming language. (<https://www.python.org/>, accessed: 2016-01-25). Python Software Foundation.
- QGIS Development Team, 2015. QGIS Geographic Information System. (www.qgis.org, accessed: 2015-12-2). Open Source Geospatial Foundation (OSGeo).
- Ramsey, P., Santilli, S., Obe, R., Cave-Ayland, M., Park, B., 2014. PostGIS – Spatial and Geographic objects for PostgreSQL. (2014-12-18). Open Source Geospatial Foundation (OSGeo). <http://postgis.net/>,
- Rau, J., Kurfürst, S., Helmich, M., 2013. *Zukunftssichere TYPO3-Extensions mit Extbase und Fluid*. O'Reilly Verlag.
- Raymond, E., 2001. *The cathedral and the bazaar: musings on Linux and Open Source by an accidental revolutionary*. O'Reilly Series. O'Reilly.
- Red Hat, Inc., 2015. Red Hat Enterprise Linux. (accessed: 2015-08-07). Red Hat, Inc. <https://www.redhat.com/de/technologies/linux-platforms/enterprise-linux>,
- Reenskaug, T., 2003. *The Model-View-Controller (MVC) - Its Past and Present*. University of Oslo. https://heim.ifi.uio.no/~trygver/2003/javazone-jao0/MVC_pattern.pdf,
- Richter, J., 1996. Out of Africa II - Die Theorie über die Einwanderung des modernen Menschen nach Europa auf dem archäologischen Prüfstand. *Archäologische Informationen* 19, 67–73.
- Richter, J., Melles, M., Schäbitz, F., 2012a. Temporal and spatial corridors of Homo sapiens sapiens population dynamics during the Late Pleistocene and early Holocene. *Quaternary International* 274, 1–4. issn: 1040-6182.
- Richter, J., Schyle, D., 2012b. Von Afrika nach Eurasien: Der Weg des Modernen Menschen. *Archäologie in Deutschland* 2012 (4), 24–27.
- Riley, J., 2009. *Glossary of Metadata Standards*. (accessed: 2015-08-02). Indiana University Libraries Digital Projects & Services. http://www.dlib.indiana.edu/~jenlrile/metadatamap/seeingstandards_glossary_pamphlet.pdf,
- RIN, 2008. To share or not to share: publication and quality assurance of research data outputs. Tech. rep. (accessed: 2015-09-14). Research Information Network. <http://www.rin.ac.uk/our-work/data-management-and-curation/share-or-not-share-research-data-outputs>,
- Rothenberg, J., Widman, L. E., Loparo, K. A., Nielsen, N. R., 1989. *The Nature of Modeling*. in *Artificial Intelligence, Simulation and Modeling*, John Wiley & Sons, 75–92.

- Rowley, J., 2007. The wisdom hierarchy: representations of the DIKW hierarchy. *Journal of Information Science* 33 (2), 163–180. eprint: <http://jis.sagepub.com/content/33/2/163.full.pdf+html>.
- RRZK, 2015a. AFS - Andrew File System. (accessed: 2015-08-05). University of Cologne. <http://rrzk.uni-koeln.de/afs.html>,
- RRZK, 2015b. Webprojekte. (accessed: 2015-08-06). University of Cologne. <http://rrzk.uni-koeln.de/webprojekt.html>,
- RRZK, 2016a. BSCW - Basic Support for Cooperative Work. (<http://rrzk.uni-koeln.de/bscw.html>, accessed: 2016-03-07). University of Cologne.
- RRZK, 2016b. SoFS – kostenloser Online-Speicherplatz. (<http://rrzk.uni-koeln.de/sofs.html>, accessed: 2016-03-01). University of Cologne.
- Sabatier, P. A., 1986. Top-Down and Bottom-Up Approaches to Implementation Research: a Critical Analysis and Suggested Synthesis. *Journal of Public Policy* 6, 21–48. issn: 1469-7815.
- Schaffert, S., Bry, F., Baumeister, J., Kiesel, M., 2008. Semantic Wikis. *Software, IEEE* 25 (4), 8–11. issn: 0740-7459.
- Schuck, W., Richter, J., Schäbitz, F., Melles, M., 2009. 1st Phase CRC 806 Proposal 2009-2013. Funding Proposal.
- Schuck, W., Richter, J., Schäbitz, F., Melles, M., 2013. 2nd Phase CRC 806 Proposal 2013-1017. Funding Proposal.
- Seemann, M., 2014. *Das Neue Spiel - Strategien für die Welt nach dem digitalen Kontrollverlust*. Orange Press.
- Selenium Contributors, 2016. SeleniumHQ Browser Automation. (<http://www.seleniumhq.org/>, accessed: 2016-02-24).
- SemanticMediawiki Contributors, 2015. Semantic MediaWiki – free and open-source extension to MediaWiki. (accessed: 2015-08-17). Open Semantic Data Association e. V. <https://semantic-mediawiki.org/>,
- Sesame community, 2015. The Sesame framework. (accessed: 2015-08-18). RDF4J. <http://rdf4j.org/>,
- Shalizi, C., 2012. *Introduction to statistical computing*. Carnegie Mellon University.
- Sitek, D., Bertelmann, R., 2014. Open Access: A State of the Art. In: *Opening Science – The Evolving Guide on How the Internet is Changing Research, Collaboration and Scholarly Publishing*. Springer Open, Heidelberg.
- Solr Contributors, 2015. Solr - open source enterprise search platform. Apache Foundation. <http://lucene.apache.org/solr/>,
- Staab, S., Studer, R., Schnurr, H.-P., Sure, Y., 2001. Knowledge processes and ontologies. *IEEE Intelligent systems* 1, 26–34.
- Stringer, C., Gamble, C., 1994. *In Search of the Neanderthals: Solving the Puzzle of Human Origins*. Thames & Hudson, Limited.
- Sveiby, K.-E., 1994. What is Information? (accessed: 05-12-2015). <http://www.sveiby.com/articles/Information.html>,
- Swartz, A. H., 2008. Guerilla Open Access Manifesto. (https://archive.org/stream/Guerilla-OpenAccessManifesto/Goamjuly2008_djvu.txt, accessed: 2016-03-10). The Internet Archive.
- Swartz, A., Hendler, J., 2001. The Semantic Web: A Network of Content for the Digital City. *Proceedings Second Annual Digital Cities Workshop*, Kyoto, Japan, October, 2001,
- Sync & Share NRW, 2016. sciebo - die Campuscloud. (<http://www.sciebo.de/>, accessed: 2016-03-01).
- Taylor, I. J., Deelman, E., Gannon, D. B., Shields, M., 2014. *Workflows for e-Science: Scientific Workflows for Grids*. Springer Publishing Company, Incorporated.

- Taylor, J., 1999. Defining e-Science. (accessed 01-17-2013). Director General of Research Councils Office of Science and Technology. <http://www.nesc.ac.uk/nesc/define.html>,
- Taylor, K. E., Stouffer, R. J., Meehl, G. A., 2012. An Overview of CMIP5 and the Experiment Design. *Bulletin of the American Meteorological Society* 93 (4), 485–498. issn: 00030007.
- Templeton, A., 2002. Out of Africa again and again. *Nature* 416 (6876), 45–51. issn: 0028-0836.
- Tilly, N., Hoffmeister, D., Cao, Q., Huang, S., Lenz-Wiedemann, V., Miao, Y., Bareth, G., 2014. Multitemporal crop surface models: accurate plant height measurement and biomass estimation with terrestrial laser scanning in paddy rice. *Journal of Applied Remote Sensing* 8 (1), 083671.
- Tolle, K., Tansley, D., Hey, A., 2011. The Fourth Paradigm: Data-Intensive Scientific Discovery [Point of View]. *Proceedings of the IEEE* 99 (8), 1334–1337. issn: 0018-9219.
- Tomlinson, R. F., Calkins, H. W., Marble, D. F., 1976. *Computer Handling of Geographical Data*. (Paris: UNESCO).
- Tonnhofer, O., Helle, D., 2014. MapProxy – Open source proxy for geospatial data. (2014-12-18). Omniscale GmbH & Co. KG. <http://mapproxy.org/>,
- Trognitz, M., 2015. Der Lebenszyklus von Forschungsdaten. IANUS - Forschungszentrum Archäologie & Altertumswissenschaften. <http://www.ianus-fdz.de/it-empfehlungen/lebenszyklus>,
- Trudeau, R., 2013. *Introduction to Graph Theory*. Dover Books on Mathematics. Dover Publications.
- Typo3 Association, 2015. The TYPO3 Association. (accessed: 2015-08-10). <http://typo3.org/association/>,
- Typo3 Contributors, 2014. Typo3 – Open Source Enterprise CMS. (2014-10-30). TYPO3 Association. <http://typo3.org>,
- Ubuntu Community, 2014. Ubuntu Linux distribution. (accessed: 2014-09-14). Canonical Ltd. <http://www.ubuntu.com/>,
- Ulbricht, D., Elger, K., Bertelmann, R., Klump, J., 2016. panMetaDocs, eSciDoc, and DOIDB—An Infrastructure for the Curation and Publication of File-Based Datasets for GFZ Data Services. *ISPRS International Journal of Geo-Information* 5 (3), 25. issn: 2220-9964.
- UO Libraries, 2015. Defining Research Data. (accessed: 2015-07-27). University of Oregon. <https://library.uoregon.edu/datamanagement/datadefined.html>,
- Verheul, J., Zickel, M., Becker, D., Willmes, C., 2015. LGM major inland waters of Europe - GIS dataset. CRC806-Database. 10.5880/SFB806.14.
- Vermeersch, P. M., 2011. Radiocarbon Palaeolithic Europe Database v14. Dept. of Earth and Environmental Sciences, Katholieke Universiteit Leuven, Belgium. <http://ees.kuleuven.be/geography/projects/14c-palaeolithic/index.html>,
- Viehberg, F. A., Willmes, C., Esteban, S., Vogelsang, R., 2015. A semantic wiki as repository to review published palaeo-data in East Africa. Collaborative Research Centre 806. 10.5880/SFB806.10.
- Voß, J., 2013. Was sind eigentlich Daten? LIBREAS. Library Ideas 23.
- Vrandečić, D., Krötzsch, M., 2006. Reusing Ontological Background Knowledge in Semantic Wikis. *SemWiki 2006*, ed. by M. Völkel, S. Schaffert. ESWC.
- Vretanos, P. A., 2010. *OpenGIS Web Feature Service 2.0 Interface Standard*. Open Geospatial Consortium Inc. <http://www.opengeospatial.org/standards/wfs>,
- Vretanos, P. A., 2005. *GML simple features profile*. Open Geospatial Consortium Inc. <http://www.ogcnetwork.net/gml-sf>,
- W3C, 2002. *Really Simple Syndication (RSS) 2.0*. W3C. <http://validator.w3.org/feed/docs/rss2.html>,
- W3C, 2015a. HTML, The Web’s Core Language. (accessed: 2015-08-11). World Wide Web Consortium. <http://www.w3.org/html/>,
- W3C, 2015b. SEMANTIC WEB. (accessed: 2015-07-28). World Wide Web Consortium. <http://www.w3.org/standards/semanticweb/>,

- W3C, 2015c. Vocabularies. (accessed: 2015-09-09). World Wide Web Consortium. <http://www.w3.org/standards/semanticweb/ontology>,
- Wallace, D. P., 2007. Knowledge Management: Historical and Cross-Disciplinary themes. Westport, CT: Libraries Unlimited.
- Warnke, M., 2011. Theorien des Internet. Junius Verlag.
- Warnke, M., 2013. Databases as Citadels in the Web 2.0. In: Lovink, G., Rasch, M. (Eds.), *Unlike Us Reader*. Institute of Network Cultures, PressBooks.com: Simple Book Production.
- Welty, C. A., 1995. An Integrated Representation for Software Development and Discovery. PhD thesis, Rensselaer Polytechnic Institute.
- Weninger, B., Edinborough, K., Bradtmöller, M., Collard, M., Crombe, P., Danzeglocke, U., Holst, D., Jöris, O., Niekus, M., Shennan, S., Schulting, R., 2010. A Radiocarbon Database for the Mesolithic and Early Neolithic in Northwest Europe. In: Crombe, P., Strydonck, M. V., Sergeant, J., Boudin, M., Bats, M. (Eds.), *Chronology and evolution within the Mesolithic of North-West Europe*. Brussels, 143–176.
- Whyte, A., Pryor, G., 2011. Open Science in Practice: Researcher Perspectives and Participation. *The International Journal of Digital Curation* 1 (6), 199–213.
- Wikimedia Fdn., 2015. MediaWiki. (accessed: 2015-08-18). Wikimedia Foundation, Inc. <https://www.mediawiki.org/>,
- Willmes, C., Baaser, U., Volland, K., Bareth, G., 2010. Internet based Distribution and Visualization of a 3D Model of the University of Cologne Campus. *Proceedings of Digital Earth Summit 2010*, International Digital Earth Society. Nessebar, Bulgaria. http://www.cartography-gis.com/pdf/36_Willmes_Germany_paper.pdf,
- Willmes, C., Bareth, G., 2012a. A data integration concept for an interdisciplinary research database. *Proceedings of the Young Researchers forum on Geographic Information Science - GI Zeitgeist*, ed. by A. Degbelo, J. Brink, C. Stasch, M. Chipofya, T. Gerkenmeyer, M. I. Humayun, J. Wang, K. Brolemann, D. Wnag, M. Eppe, J. H. Lee. ifgiPrints 44. Akademische Verlagsgesellschaft AKA, Heidelberg, 67–72.
- Willmes, C., Becker, D., Brocks, S., Hütt, C., Bareth, G., 2014a. Köppen-Geiger classification of MPI-ESM-P LGM simulation. CRC806-Database. 10.5880/SFB806.2.
- Willmes, C., Becker, D., Brocks, S., Hütt, C., Bareth, G., 2014b. Köppen-Geiger classification of MPI-ESM-P Mid-Holocene simulation. CRC806-Database. 10.5880/SFB806.3.
- Willmes, C., Becker, D., Brocks, S., Hütt, C., Bareth, G., 2014c. Köppen-Geiger classification of MPI-ESM-P Pre-Industrial simulation. CRC806-Database. 10.5880/SFB806.4.
- Willmes, C., Becker, D., Brocks, S., Hütt, C., Bareth, G., 2014d. pyGRASS implementation of Köppen-Geiger classifications from CMIP5 simulations. CRC806-Database. 10.5880/SFB806.1.
- Willmes, C., Weskamm, J., 2007. SeismoGIS - Ein GISTool zur Analyse von Erdbebendaten für die Erdbebenstation der Universität zu Köln. In: Strobl, J., Blaschke, T., Griesebner, G. (Eds.), *Angewandte Geoinformatik 2007*. Wichman Verlag, Heidelberg, 858–866.
- Willmes, C., Weskamm, J., K.-G. Hinzen, U. B. and., Bareth, G., 2008. SeismoGIS: A tool for the visualization of earthquake data. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Science*, vol. XXXVII. Part B8. International Society of Photogrammetry and Remote Sensing. Beijing China, 1239–1244. http://www.isprs.org/proceedings/XXXVII/congress/8_pdf/12_WG-VIII-12/11.pdf,
- Willmes, C., Becker, D., Brocks, S., Hütt, C., Bareth, G., 2016. High-resolution Köppen-Geiger paleoclimate classifications. *Transactions in GIS*.
- Willmes, C., Brocks, S., Hoffmeister, D., Hütt, C., Kürner, D., Volland, K., Bareth, G., 2012b. Facilitating integrated spatio-temporal visualization and analysis of heterogeneous archaeological and palaeoenvironmental research data. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* I-2, 223–228.

- Willmes, C., Kürner, D., Bareth, G., 2014e. Building Research Data Management Infrastructure using Open Source Software. *Transactions in GIS* 18, 496–509. issn: 1467-9671.
- Willmes, C., Yener, Y., Gilgenberg, A., Bareth, G., 2015. CRC806-Database: Integrating Typo3 with GeoNode and CKAN. *Proceedings of the 2nd Data Management Workshop*, ed. by C. Curdt, C. Willmes, G. Bareth. Cologne, Germany.
- Winkler-Nees, S., 2013. Status of Discussion and Current Activities: National Developments. In: Neuroth, H., Strathmann, S., Oßwald, A., Ludwig, J. (Eds.), *Digital Curation of Research Data*. vhw Verlag, Glücksstadt. 18–37.
- Winn, J., 2013. Open Data and the Academy: An Evaluation of CKAN for Research Data Management. *IASSIST 2013*, 28-31 May 2013, Cologne. <http://eprints.lincoln.ac.uk/9778/>,
- Wordpress Contributors, 2014. Wordpress – Open Source Blog Engine. (2014-11-09). <https://wordpress.org>,
- Wynne, M., Hellesoy, A., 2012. *The Cucumber Book: Behaviour-Driven Development for Testers and Developers*. Pragmatic Bookshelf.
- Yu, L., 2007. *Introduction to the Semantic Web and Semantic Web Services*. Chapman & Hall, CRC Press, Taylor & Francis Group, Boca Raton.

List of Tables

2.1	A ranking of the support infrastructure for decision making (Longley et al. 2005).	33
2.2	Document vs. knowledge approach, after (Staab et al. 2001).	39
2.3	Example knowledge item, through linking of context information.	47
2.4	Overview of some State-of-the-Art Research Data Lifecycle Models.	60
3.1	Specific demands for RDM infrastructure implementations.	85
3.2	Table overview of applied OGC OWS.	100
3.3	Table overview of applied metadata standards.	101
3.4	Creative Commons License options model. Official descriptions from (Creative Commons 2015).	104
5.1	Server infrastructure of the CRC806-Database, as of July 2015.	121
5.2	Overview of main software components and applications of the CRC806-Database.	123
5.3	MediaWiki and most relevant extensions setup	130
6.1	Overview of Typo3 Extbase & Fluid Extensions.	141
6.2	CKAN metadata schema. Source: Open Knowledge Foundation (2014).	148
6.3	Metadata Serialization formats available from CKAN.	149
6.4	Five star open data logo badges.	154
6.5	Vocabularies applied for RDFa in the CRC806-Database system.	154
6.6	The BibTeX Schema.	159
9.1	The sections of the CRC806-Database web application.	203
11.1	Integrated published archeological databases.	240

12.1 Discussion of RDM demands.	254
12.2 Addressing the funders demands.	257
12.3 Application vulnerabilities and their post-project handling.	261
12.4 CRC806-Database major update.	262
12.5 Volume of data stored in the AFS backend by cluster, as of March 2016.	271

List of Figures

1.1 Our Way. Source: (Richter et al. 2012a).	22
1.2 Geographical focus regions of the second CRC 806 funding phase 2013 - 2017 (Schuck et al. 2013: 12).	23
2.1 The wisdom hierarchy (Rowley 2007), or the DIKW pyramid (Jennex 2009; Frické 2009) based on the Knowledge Pyramid (Ackoff 1989).	28
2.2 The different degrees of formalization in KR methods: from unstructured textual content to ontology and logical rules. Source: Navigli et al. (2008: 72)	38
2.3 Five-star deployment scheme for Open Data. Source: http://5stardata.info/ , CC-0.	43
2.4 Example of a semantic network. Source: Own work.	44
2.5 Example architecture of an KMS strictly based on SWT. Source: Allemang et al. (2011: 57)	49
2.6 Major network topologies by Baran (1964).	52
2.7 A graph with labeled nodes by degree and its degree distribution.	52
2.8 Connectedness and Diameter of a network or graph. Source: (Barabási et al. 2003: 51).	53
2.9 Random vs scale-free network. Source: (Barabási et al. 2003).	54
2.10 The Research Lifecycle divided into 6 research stages and the data management part is subdivided into 5 stages. Source: (Oregon State University Libraries 2015)	61
2.11 Overview of the Open Access process. Source (Bartling et al. 2014; Sitek et al. 2014).	71
2.12 Six parts of a GIS. Changed after: (Longley et al. 2005: 24)	75
3.1 Characterization of Information Types in repositories. Source: (European Commission 2006: 16)	90
3.2 CRC806-Database Research Data Lifecycle. Source: own work.	91
3.3 Project Repository Lifecycle Model. Source: Own work.	93
3.4 System architecture of the CRC 806 Research Data Management infrastructure. Source: Willmes et al. (2014e).	96
3.5 Architecture of the CRC806-KB infrastructure. Source: Own work.	98
4.1 Domain Driven Design layered view. Source: Own work, after Lobacher (2014: 74).	108
4.2 Model View Controller (MVC) concept. Source: Own work.	109
4.3 The components of a Continuous Integration System. Source: (Duvall et al. 2007: 5).	110
4.4 The prototyping approach: a four step iterative process. Source: Own work, after Naumann et al. (1982).	115

5.1	RRZK data storage infrastructure. Source: Curdt (2014b: 71).	120
5.2	GeoNode Component Architecture. Source: GeoNode Contributors (2014)	126
5.3	MapProxy overview. Source: http://mapproxy.org/ , CC-BY: Omniscale GmbH & Co. KG.	127
5.4	SMW Architecture and integration into MW. Source: Krötzsch et al. (2007).	128
5.5	SMW interfaces. Source: Own work.	129
5.6	Example SemanticMap.	133
5.7	The Mobo CLI interface.	134
6.1	Layered architecture diagram of the CRC806-RDM infrastructure. Source: Own work	140
6.2	MVC based architecture of Typo3 Extabase & Fluid Data extension.	147
6.3	Layouts build the outer frame for a template, whereas recurring elements can be implemented in a template with partials. Source: (Rau et al. 2013).	151
6.4	Screenshot of the CRC806-Database data catalog datamap.	152
6.5	UI element for license editing (left) and display (right) on the dataset pages.	153
6.6	Screenshot of data catalog example bibliography metadata information.	155
6.7	Screenshot of data catalogue example related resources of a dataset.	158
6.8	Interfaces of the Bibliographic Database.	161
6.9	The CRC806-Database CI-System pipeline. Source: Own work.	167
7.1	Architecture of the CRC806-Database SDI. Source: Own work.	170
7.2	Screenshots of the GeoNode Web GUI based upload process.	173
7.3	GUI screenshots of the OpenGeo Explorer QGIS plug-in.	173
7.4	Integration of spatial KB data into the SDI. Source: Own work.	174
7.5	GeoNode Web UI for registering external (remote) services. Source: Screenshot.	175
8.1	Data model development feedback process during external dataset import. Source: Own work.	182
8.2	Structure of the Mobo-Schema. Source: Heimler (2015a).	183
8.3	Form for editing/creating a dataset record in the CRC806-Knowledgebase. Source: Screenshot.	188
8.4	A dataset record page of the CRC806-Knowledgebase. Source: Screenshot.	189
8.5	A Bibliography record page of the CRC806-Knowledgebase. Source: Screenshot.	190
8.6	Spreadsheet import via Data Transfer extension. Source: Own work.	190
8.7	DataImport process tool-chain. Source: Own work.	191
8.8	Semantic Search user interface. Source: Screenshot.	194
8.9	Outputs of the two queries, map result and broadtable result. Source: Screenshot.	196
9.1	Navigation bar of the CRC806-Database web application. Source: Screenshot.	203
9.2	The frontpage of the CRC806-Database web application. Source: Screenshot.	205
9.3	The data catalog interface. Search, filter and browse interface on the left. Detail dataset page on the right. Source: Screenshot.	206
9.4	Form for uploading a new dataset (left). Form for editing an existing dataset (right). Source: Screenshot.	209
9.5	Screenshot of access rights UI. Source: Screenshot.	210
9.6	Screenshot of the Request Access form. Source: Screenshot.	210
9.7	The publications DB interface. List, search, filter and browse interface on the left. Detail metadata information page on the right. Source: Screenshot.	211
9.8	The publication lists rendered on the main CRC 806 website. Source: Screenshot.	213
9.9	The publication lists rendered on the IRTG website. Source: Screenshot.	214

9.10	The members list (left). A members profile page (right). Source: Screenshot.	215
9.11	The interface of the integrated search. Integrated searches are triggered through the search box in the navigation bar (left). The results are shown in a list (right). Source: Screenshot.	216
9.12	The data metrics interface of the CRC806-Database. Source: Screenshot.	217
9.13	The PIWIK visitor statistics interface. Source: Screenshot.	218
9.14	Screenshots of the CRC806-Database admin console. A: Admin console drop-down menu. B: Member admin user list view. C: Member profile edit view. Source: Screenshot.	220
9.15	The UI for administering DOI requests. The left side shows the list of DOI requests and its status. The right side shows the editing UI for DOI datasets. Source: Screenshot.	222
10.1	The maps SDI catalog style interface (left). WebGIS based geodata layer detail view (right). Source: Screenshot.	224
10.2	List of related resources in the information panel of the maps detail view. Source: Screenshot.	225
10.3	Detail views of the maps datasets detail page information panel. Source: Screenshot	226
10.4	The OpenLayers based WebGIS interactive map. Source: Screenshot.	227
10.5	The MapProxy cached GEBCO WMS displayed in QGIS Desktop. Source: Screenshot	228
10.6	MapProxy web interface. Source: Screenshot.	230
10.7	GeoNode web application. Dataset list on the left, detail dataset view on the right. Source: Screenshot.	231
11.1	Types of geodata sources. Source: Own work.	235
11.2	Wiki page of a Bibliographic record with annotated resources. Source: Screenshot. .	237
11.3	UI of the Afriki KB. Source: Screenshots.	239
11.4	UI of the Contextual Areas KB. Source: Screenshots.	241
12.1	Köppen-Geiger Paleoclimate classification publication, annotated with the open access GIS datasets in the CRC806-Database web application. Source: Screenshot. .	267
12.2	Time Slice Map of the LGM paleoenvironment of Europe. Source: (Becker et al. 2015)	268
12.3	Example CRC806-Database publication metadata document of Becker et al. (2015). .	269
12.4	Datasets and Publications by Cluster and Project, as of March 2016. Source: Own work.	270
12.5	Data format allotment of resources stored in the CRC806-RDM repository, as of March 2016. Source: Own work.	271
12.6	Downloads per Resource, as of March 2016. Source: Own work.	272

Listings

4.1	Vocabulary definition in the HTML document header.	116
4.2	RDFa annotated bibliographic record.	116
5.1	Example Semantic MediaWiki template definition.	131
5.2	Example Semantic MediaWiki template call, for storing structured data.	131
5.3	Example inline query, yielding a map as result format.	132
6.1	Folder structure of the AFS based long term storage.	144
6.2	File tree structure of the Extbase & Fluid Data extension.	145
6.3	Example XML Metadata for issuing a DOI, DataCite Schema 3.0.	150
6.4	Example view helper, for displaying a list of blog posts.	151
6.5	RDFa vocabularies included in the data catalogue pages.	154
6.6	Example RDFa on data catalogue sites.	155
6.7	Example RDF export.	156
6.8	Example BibTeX entry.	160
6.9	PHP snippet for rendering the publications list.	161
6.10	Example RDFa annotation on the user profile pages.	164
6.11	RSS Feed shortened example.	165
6.12	Folders and files of the Integrated Search extension.	166
7.1	Setup of the extra table space for the growing PostgreSQL tables.	171
7.2	Mapserver configuration file for the CRC 806 SRTM WCS.	175
7.3	Example MapProxy configuration for the GEBCO Bathymetrie WMS.	177
7.4	File tree structure of the Extabase & Fluid Maps extension.	178
8.1	CRC806-Knowledgebase Mobo project model folder and file structure.	184
8.2	An example Mobo Field definition: /field/Dataset.yaml.	185
8.3	An example Mobo Model definition: /model/Dataset.yaml.	185
8.4	An example Mobo Form definition: /form/Dataset.yaml.	186
8.5	An example DataTransfer CSV import file.	191
8.6	Example DataTransfer XML.	191
8.7	An example script header, of a python import script.	192
8.8	Example main part of an Python import script.	192
8.9	Serialization of the XML tree into an XML document.	193
8.10	Temporal Mobo model, '/model/Temporal.yaml'.	193
8.11	Temporal model ASK queries, '/smw_query/temporal-query.ask'.	194
8.12	Example query for artefacts in a given timePeriod.	195
8.13	An example inline query yielding a map as output format, showing artefacts dated in a given TimePeriod.	196
8.14	An example ASK API request.	196
8.15	JSON result.	197

Danke

Vieles ist mir sicherlich entgangen, was wichtig gewesen wäre. Es wird sich nach den Prinzipien der Selbstorganisation zusammenfinden, sobald ein neuer Fokus entstanden ist. Das ist eine der großen Stärken der Wissenschaft. – Josef H. Reichholf (2008): Warum die Menschen sesshaft wurden. Fischer Verlag, Frankfurt am Main.

Zuerst möchte ich mich ganz herzlich bei Herrn Prof. Dr. Georg Bareth bedanken, in dessen Arbeitsgruppe am Geographischen Institut der Universität zu Köln ich seit 2006, zunächst als Studentische Hilfskraft (SHK), dann als Wissenschaftlicher Mitarbeiter bzw. als Doktorand, Mitglied bin. In diesen 10 Jahren habe ich sehr viel gelernt, ich durfte an vielen sehr interessanten Projekten und noch mehr Konferenzreisen teilnehmen. Herr Bareth hat maßgeblichen Anteil an meiner wissenschaftlichen Entwicklung, was ich sehr zu Schätzen weiß und für den ich ihm stets zu Dank verpflichtet bin.

Herrn Prof. Dr. Ulrich Lang möchte ich herzlich für die Übernahme des Zweitgutachtens der vorliegenden Arbeit danken. Er kennt mich bereits aus meinem Diplomstudium in dem ich an Vorlesungen und Seminaren in Informatik von ihm teilnahm. Ich war sogar für 6 Monate mal SHK im RRZK (Abt. Systeme), als Herr Lang erst seit kurzem die Leitung des RRZK übernommen hatte. Das Zweitgutachten meiner Diplomarbeit hatte er damals auch schon übernommen und meine Diplomprüfung im Nebenfach Informatik legte ich ebenfalls bei ihm ab.

Zu größter Dankbarkeit bin ich auch meinen SHK's die seit 2009 im Rahmen des SFB 806 mit mir zusammengearbeitet haben verpflichtet. In chronologischer Reihenfolge danke ich Johannes Weskamm, Oliver Biesmann, Elke Dornauf, Christoph Hütt, Daniel Kürner, Kai Volland, Daria Grafova, Yasa Yener, Anton Gilgenberg, Jan Verheul und Mirijam Zickel. Hierbei sind Daniel Kürner und Yasa Yener etwas hervorzuheben, beide haben überdurchschnittlich viel im Rahmen Ihrer HiWi-Tätigkeit geleistet und haben erheblichen Anteil an der Programmierung der CRC806-Database, für den ich überaus dankbar bin. Für das kollegiale und freundschaftliche Miteinander in der Arbeitsgruppe möchte ich mich bei allen AG Mitgliedern bedanken, besonders aber bei meinen "Büro-Genossen" Daniel Becker und Sebastian Brocks, sowie Christoph Hütt, Tim Schlüter und Andreas Bolten für Feedback, Diskussionen, Ideen und Freundschaft.

Abschließend möchte ich mich bei meiner Familie und Freunden für Rückhalt, Bestärkung, Trost, Ermunterung und Motivation bedanken. Ich bin nicht immer einfach, umso mehr weiß ich diejenigen zu Schätzen die es mit mir aushalten. Herzlichen Dank!

Erklärung

Ich versichere, dass ich die von mir vorgelegte Dissertation selbständig angefertigt, die benutzten Quellen und Hilfsmittel vollständig angegeben und die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken im Wortlaut oder dem Sinn nach entnommen sind, in jedem Einzelfall als Entlehnung kenntlich gemacht habe; dass diese Dissertation noch keiner anderen Fakultät oder Universität zur Prüfung vorgelegen hat; dass sie – abgesehen von unten angegebenen Teilpublikationen – noch nicht veröffentlicht worden ist, sowie, dass ich eine solche Veröffentlichung vor Abschluss des Promotionsverfahrens nicht vornehmen werde. Die Bestimmungen der Promotionsordnung sind mir bekannt. Die von mir vorgelegte Dissertation ist von Prof. Dr. Georg Bareth betreut worden.

Köln, den 04.04.2016

Christian Willmes

Teilpublikationen

Willmes, C., Becker, D., Brocks, S., Hütt, C. and Bareth, G. (2016): High Resolution Köppen-Geiger Classifications of Paleoclimate Simulations. Trans. in GIS. doi:10.1111/tgis.12187

Willmes, C., Kürner, D. and Bareth, G. (2014): Building Research Data Management Infrastructure using Open Source Software. Transactions in GIS. doi: 10.1111/tgis.12060

Willmes, C., Brocks, S., Hoffmeister, D., Hütt, C., Kürner, D., Volland, K., Bareth, G. (2012): Facilitating integrated spatio-temporal visualization and analysis of heterogeneous archaeological and palaeoenvironmental research data. ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. I-2, 223-228. doi: 10.5194/isprsannals-I-2-223-2012.

Willmes, C. and Bareth, G. (2012): A data integration concept for an interdisciplinary research database. In: Proceedings of the Young Researchers forum on Geographic Information Science - GI Zeitgeist, ifgiPrints 44, Münster, Germany, March 2012, ISBN: 978-3-89838-663-0, Akademische Verlagsgesellschaft AKA, Heidelberg, pp. 67 - 72.