Supporting loose forms of collaboration

Using Linked Data to realize an architecture for collective knowledge construction

HANNES EBNER

Doctoral Thesis
Stockholm, Sweden 2014
Akademisk avhandling som med tillstånd av Kungl Tekniska högskolan framlägges till offentlig granskning för avläggande av teknologie doktorsexamen i medieteknik torsdagen den 22 maj 2014 klockan 10.00 i sal F3, Kungliga Tekniska högskolan, Lindstedtsvägen 26, Stockholm.

Hannes Ebner, March 2014

Published under the Creative Commons BY-NC-ND 4.0 license. http://creativecommons.org/licenses/by-nc-nd/4.0/

Printed by Universitetsservice US-AB
Abstract

This thesis is driven by the motivation to explore a way of working collaboratively that closely reflects the World Wide Web (WWW), more specifically the potential of the Web architecture built on Semantic Web technologies and Linked Data. The goal is to describe a generic approach and architecture that satisfies the needs for loose collaboration and collective knowledge construction as exemplified by the applications described in this thesis. This thesis focuses on a contribution-centric architecture which allows for flexible applications that support loose forms of collaboration.

The first research question deals with how Web-based collective knowledge construction can be supported. The second research question explores the characteristics of collective knowledge construction with respect to the Open World Assumption (OWA). The OWA implies that complete knowledge about a subject cannot be assumed at any time, which is one of the most fundamental properties of the WWW. The third research question investigates how Semantic Web technologies be used in order to support such a contribution-centric architecture.

The thesis and its underlying publications are of a technical character and are always grounded in theoretical models and considerations that have led to functional implementations. The research has evolved in iterative development processes and was explicitly directed at building applications that can be used in collaborative settings and that are based on standardized Web technologies. One of the main outcomes, an information model, was developed together with such an application and provides a number of novel approaches in the context in which it was designed.

The validity of the presented research is supported by evaluations from different perspectives: a list of implemented applications and showcases, results from structured interviews that have investigated the suitability for various resource annotation processes, as well as scalability aspects.

The thesis concludes that it is ultimately up to the application how “loose” the collaboration should be and to which extent the OWA is incorporated. The presented architecture provides a toolkit to support the development of loosely collaborative applications. The showcased applications allow the construction of collaborative conceptual models and to collaboratively annotate educational resources. They show the potential of the used technology stack and the introduced contribution-centric architecture that sits on top if it.
Acknowledgements

This research would not have been possible without the support of several people and organizations.

I would like to start thanking my supervisors Marko Turpeinen and Ambjörn Naeve for their complementary support during all these years.

I am grateful for the help of my then colleagues at the Knowledge Management Research group, who introduced me to the Semantic Web and laid the foundation of my work; in particular, thanks to Matthias Palmér, who provided much appreciated feedback on my thesis and with whom I co-founded a company, but also to Fredrik Enoksson, Mikael Nilsson and Erik Isaksson for our inspiring discussions.

I also had the privilege of working closely with Nikos Palavitsinis and Nikos Manouselis; our unwound conversations supported me in getting different perspectives and to stay the course.

I would also like to thank my colleagues at the Centre for Sustainable Communications; especially Jorge Zapico, with whom I hacked on quite a few projects and co-initiated the Green Hackathon event series.

My colleagues at the department of Media Technology and Interaction Design provided me with perspectives on very differing areas of research for which I am grateful; thanks to Cristian Bogdan for his invaluable feedback on the drafts of this thesis, and to Anders Lundström for sometimes diverting my attention from work to fly fishing.

I want to thank my family back home in Austria for always supporting my decisions.

Finally, I would like to thank my wife Maria for all of the support she has given through all of these years, and my son Vide for always making me smile.

Hannes Ebner
Saltsjöbaden, March 2014
Contents

Acknowledgements v

Contents vi

List of Figures viii

List of Tables ix

1 Introduction 1
1.1 Knowledge construction using Web technologies 1
1.2 The Open World Assumption 2
1.3 Research objectives 2
1.4 Main research questions addressed in this thesis 4
1.5 Research methodology 5
1.6 The research context of this thesis 8
1.7 Scope and main target audience 9
1.8 Terminology 10
1.9 Outline 11

2 Theoretical background and technologies 13
2.1 Web Architecture 13
2.2 Concepts 15
2.3 Web applications and Open World Assumption 17
2.4 A contribution-centric approach 20

3 A contribution-centric architecture 21
3.1 Artifacts applied 21
3.2 Management of resources and their metadata 23
3.3 An information model to support a contribution-centric architecture 25
## CONTENTS

3.4 Artifacts based on entries ........................................ 29
3.5 The use of Web technologies and Linked Data ................. 30
3.6 Related technologies ............................................... 31
3.7 Summary .......................................................... 31

4 Use cases for the information architecture ...................... 35
4.1 Collaborative conceptual modeling .............................. 35
4.2 Collaborative annotation of educational resources ............. 39
4.3 Evaluation .......................................................... 42
4.4 Summary .......................................................... 47

5 Conclusions .......................................................... 49
5.1 Research questions revisited ...................................... 49
5.2 Building applications on a contribution-centric architecture .... 53
5.3 Future work ........................................................ 53

Bibliography .......................................................... 55

Paper summaries ...................................................... 63
  Paper 1: Collaborative Construction of Artifacts .................. 63
  Paper 2: Collaborative Tasks using Metadata Conglomerates – The Social Part of the Semantic Desktop ......... 63
  Paper 4: Learning Object Annotation for Agricultural Learning Repositories ............................ 65
  Paper 5: An information model for managing resources and their metadata ............................. 65
Other publications ..................................................... 66
List of Figures

3.1 ReM$^3$ Entry and its linked information .......................... 26
3.2 ReM$^3$ terms described in an RDF schema .......................... 33

4.1 Screenshot of a Context-map with link and contextual-neighborhood ... 37
4.2 Screenshot of a Context-map with a highlighted contribution ........... 38
4.3 Screenshot of a portfolio view in EntryScape .......................... 40
List of Tables

3.1  Comparison of Web applications and ReM$^3$  . . . . . . . . . . . . . . . . . . . . . 32
Chapter 1

Introduction

1.1 Knowledge construction using Web technologies

The World Wide Web (WWW) and the underlying technologies serve as the foundation for a large spectrum of applications. The networked character of the WWW allows many ways to collectively construct knowledge, ranging from loose and ad-hoc approaches, without a well-defined list of collaborators or a clear goal to work towards, all the way to approaches controlled by well-structured collaborative applications with clearly defined tasks and specific workflows.

This thesis explores the loose end of collective knowledge construction (which is also referred to as loose forms of collaboration in this text) and describes a architecture that supports flexible forms of collaboration under the Open World Assumption (OWA – see section 2.1). In contrast to strictly controlled collaborative environments, the presented architecture builds on contributions that are non-invasive modifications and result in knowledge artifacts. Non-invasive modifications provide an addition to the original artifact, which is in contrast to invasive modifications, where the original is changed as soon as it is contributed to. Before the architecture is introduced, it is explored whether the concept of collaboration fits such knowledge construction processes and how it relates to relevant research within Computer-supported Cooperative Work (CSCW) and related fields. However, the main context of this thesis are the areas involved in information management and Web Science, more specifically Web architecture built on Semantic Web technologies and Linked Data.
1.2 The Open World Assumption

According to the high-level design model of the WWW all content is published under the OWA, however, there are closed islands (e.g. behind paywalls) which are not discussed here as they do not belong to the “open” WWW. The OWA implies that anyone can write anything about anything and complete knowledge cannot be assumed. The OWA is of special importance for collective construction of knowledge and therefore a central topic in this thesis. Social contracts and coordinated actions can exist in both open and closed worlds, but the research described here focuses on Web technologies and the OWA. The contribution-centric architecture as discussed in this thesis goes further and allows for contributing to artifacts without having the explicit access right to edit.

1.3 Research objectives

**Collaborative construction of conceptual models**

The contribution-centric architecture that is the central part of this thesis was developed in the context of various research projects (see section 1.6) from where also the requirements for the developed applications emerged. The initial case was a conceptual modelling tool to which collaborative support should be added. One of the core requirements was that it should be possible to add information to existing conceptual models, while keeping contributions separate. The author of the original conceptual model should not have to think about future contributions, i.e., work without involving access control or an explicit group of collaborators. Very much like the interlinked information on the Web, the collectively constructed knowledge should be able to emerge without the need for any controlled process.

**Resource annotation**

The work that followed after the construction of conceptual models as collaborative artifacts was centered around resource and metadata management, more concretely the annotation of educational resources with metadata. The *Organic.Edunet project* had the goal of establishing a federation of learning repositories, searchable through a unified interface, namely the Organic.Edunet portal. The repository tools, appearing in this thesis as SCAM (the backend, now called EntryStore [1]) and Confolio (the frontend, now called EntryScape [2]), were developed during the course of the project – mostly due to the fact that so called “entries” became central to manage information, the name was changed from SCAM to EntryStore [1] in 2013.
1.3. RESEARCH OBJECTIVES

Content providers used (and still use) the platform to either contribute to existing educational material or to publish resources from scratch. The work within Organic.Edunet required taking a technical path that eases problems that typically occur in situations where data integration is necessary.

Data integration and Linked Data

The overall goal was to develop something that can be (re)used also in other situations and projects, without being restricted to Organic.Edunet. One of the objectives became that the implemented architecture and the flexible way of describing resources and enhancing metadata should be formalized as a generic information model. Such a model should then be used to contribute and to contextualize information in a way that is similar to the artifact ideas as implemented for conceptual modelling.

Due to the heterogeneous character of the individual institutional repositories that were to be harmonized, and because of the need of an approach to solve data integration problems, the decision in the research projects was made to make extensive use of Semantic Web [3] technologies and specifically Linked Data [4]. The implementation of the backend already made extensive use of a triple store, but it was to be formalized into an information model based on the Resource Description Framework (RDF). An integration into an emerging set of open conceptual models forming the Web of Data (now better known as the Linked Open Data cloud [5]) was desired.

A generic architecture

The overall goal of this thesis is to describe a generic approach for collectively constructing knowledge that not only can be applied to the applications described in chapter 4, but also rather generically to applications with needs that can be satisfied by the information model. The information model provides a way of expressing and linking information and does not require any specifics other than the reliance on Web technologies.

The contribution-centric architecture as described in this thesis is grounded both in the collaborative construction of artifacts and the information model that was formalized in paper 5.
1.4 Main research questions addressed in this thesis

The research objectives above were comprehensive as they originated partially from the Description of Work (DoW) of the funded research projects, and partially from the research interests of the Knowledge Management Research group [6] at KTH within which this work was carried out. This context determined parts of the technological framework such as that project outcomes should be based on Web technologies, with an emphasis on the Semantic Web [3] stack. The objectives led to a series of research questions which are described below.

How can Web-based collective knowledge construction be supported?

This research question deals with Web technologies that support loose forms of collaboration. It is investigated how already existing Web-based collaborative techniques are used for collective knowledge construction.

What are the characteristics of collective knowledge construction with respect to the Open World Assumption?

The contribution-centric architecture described in this thesis is based on non-invasive modifications (referred to as contributions) to support collective knowledge construction. It has a close match in the Open World Assumption (OWA) (see section 2.1) which is one of the most fundamental properties of the WWW. Open collaboration applications, such as wikis and blogs, work within their context and for their purpose, but they are restricted because of socially induced complications.

It is investigated how the proposed contribution-centric architecture compares to the approaches of other collaborative Web applications. This is done with the help of the characterization of collective knowledge construction with respect to the OWA.

How can Semantic Web technologies support a contribution-centric architecture?

Based on this research question it is investigated how the Semantic Web stack can be used to support the proposed contribution-centric architecture. Semantic Web research so far has not been focused on discussing collaborative Web applications and it is investigated whether such a technology stack provides a good fit for the proposed architecture.
1.5 Research methodology

Most of the research described in this thesis has been carried out in the context of collaborative international projects, where it is common to deliver according to a detailed work plan in which the project is outlined several years in advance. Also, in such projects a strong focus lies on applied research, concrete outcomes and innovation. This means that research starts by introducing new technology into a certain setting as a solution to a problem. The technology is demonstrated in this setting and followed by training. Thereafter, researchers study how people use the new technology and publish on the results. Finally, this leads to evaluation of the technology, in turn leading to further improvements and onto the next project. See further “A design science research methodology for information systems research” [7].

This thesis centers around Web technologies and the architecture of collaborative Web applications. It is appropriate to put the research into the context of Web Science as described by Hendler et al in [8]. There the approach and circumstances for understanding and researching the Web and Web applications are described with (inter alia):

*It may seem that the best way to understand the Web is a set of protocols that can be studied for their properties, with individual applications analyzed for their algorithmic properties. However, the Web wasn’t (and still isn’t) built using the specify, design, build, test development cycle Computer Science has traditionally viewed as software engineering best practice.*

*...*

*A software application is designed based on an appropriate technology (such as algorithm and design) and with an envisioned “social” construct; it is indeed a contradiction in terms to talk about a Web application built for a single user on a single machine. The system is generally tested in a small group or deployed on a limited basis; the system’s "micro" properties are thus tested.*

*...*

*The macro system, that is, the use of the micro system by many users interacting with one another in often-unpredicted ways, is far more in-
interesting in and of itself and generally must be analyzed in ways that are different from the micro system.

These two categories of systems, micro and macro, are used to explain some of the outcomes described in this thesis. What does this mean in the context of this thesis? The micro system is typically found within funded projects (such as Organic.Edunet) where the actual development happens. The group of users is limited and usually well-known. A macro system on the other hand is established when the system is used “in the wild”, possibly in ways that were not expected. Such unpredicted and suddenly emerging applications are made possible by the use of Web architecture (and Linked Data specifically) which facilitates a switch from micro to macro systems. E.g., a resource described with generic metadata in a library system can be referred to, the metadata can be built upon, and the generic resource can for example be contextualized to be used as teaching material in a specific educational setting.

Iterative design and development processes

The presented research has evolved in iterative development processes and was explicitly directed at building applications that can be used in collaborative settings and that are based on standardized Web technologies. One of the main outcomes, an information model, was developed together with such an application and provides a number of novel approaches in the technological context in which it was designed.

The iterative phases of the work consisted of the following key elements:

The context: a work plan grounded in the projects’ DoW. Most of the work described in this thesis has been carried out in the projects PROLEARN and Organic.Edunet, see section 1.6. In PROLEARN it was necessary to support the creation of conceptual models with an open-ended list of collaborators, mostly to avoid restrictions and to open up for contributions by persons who otherwise would have been overlooked and excluded from contributing. This led to the first ideas about artifacts and in the following to papers 1 and 2. The technologically federated character of Organic.Edunet made it necessary to involve a number of different repositories from which metadata should originate to build upon. Such a complex prerequisite together with the demand for a conceptually sound solution led to a new iteration of a framework for resource and metadata management which is
described in papers 3 and 4. This work developed into a generic information model for managing resources and their metadata, which was described in paper 5.

Investigations using workshops and video conferences. This phase was carried out using different forms of workshops and also video conferences where the participants were asked what they need and expect. This led to “lightweight prototypes” which were used for the phase to define use cases. There were well-specified use cases from the beginning, but they originated from the project funding proposal and it is quite natural that such use cases change over the course of time or make adjustments necessary. Brainstorming and feedback sessions showed the approximate path to go, but the use cases were created and evaluated on a more realistic level after the projects had been running for some time. In addition it was necessary to perform some technical analyses to mitigate the risks for interoperability issues.

Definition of use cases. Use cases and basic interaction such as “flexible collaboration around conceptual models”, “management of distributed metadata for resources”, etc, were either existing use cases from the beginning or emerged later in the course of the projects.

Delivery of prototypes and stable software. Prototypes and functional, stable software were developed during the projects. The iterative approach allowed for using these prototypes as input for refinements in previous steps and further development.

Most of the work was carried out between the phases involving use cases and the delivery of prototypes and software. Use cases change, so the relevant phase has to be revisited, and because of this the software changes. This is a process which carries on as long as the software is maintained and used. The core of this thesis, the information model, can be seen as a spin-off from a successful iteration of the last phase.

Published work
The research process is reflected by the following elements in the peer-reviewed publications of which this thesis consists:

- Paper 1 provides a description of the concept of artifacts in the context of conceptual modeling as use case. An implementation of the collaboration server Collaborilla and client-support in the conceptual modeling tool Conzilla is provided.
• Paper 2 presents a collaborative task management application that explicitly integrates metadata into the collaboration process. The paper is based on the approach as presented in paper 1 and provides a use case and lightweight prototypes in the form of mockups.

• Paper 3 describes a framework for resource and metadata management that is adapted to the needs of web applications, in particular mashups. The requirements and use cases emerged from projects in which context the framework was developed and used. The paper describes a functional prototype that is used as input for further iterations in the respective projects.

• Paper 4 presents the design and first implementation of a Web-based tool for learning object annotation in the agricultural context. This paper builds on the framework from paper 3 and gives an account of the first stable application, resulting from a development process closely accompanied by project workshops and video conferences. The feedback was mostly provided by content providers who used the system productively.

• Paper 5 introduces an information model together with a stable reference implementation. The paper mostly builds on the work from papers 3 and 4, but in addition several showcases are presented and the information model is evaluated from several perspectives. The evaluation includes an analysis of the suitability of the information model for resource annotation, a preliminary scalability analysis, and the level of adoption in applications.

1.6 The research context of this thesis

PROLEARN

Within the EC-funded Network of Excellence PROLEARN [9] the conceptual modelling tool Conzilla [10] was extended with collaborative features. This required the development of the collaboration server Collaborilla and the addition of client-side support to Conzilla. This resulted in an artifact-based tool for collaboratively constructing conceptual models as described in paper 1.

Organic.Edunet

In the EC-funded eContentplus project Organic.Edunet a resource annotation tool consisting of the Web-based frontend Confolio and its backend SCAM version 4 were developed. In Organic.Edunet these applications were used to annotate resources
with educational metadata, to manage information originating from different repositories and also to adapt (contextualize) existing metadata for new educational purposes.

**Data-driven sustainability and data bridges**

In the context of the VINNOVA-funded Centre of Excellence Sustainable Communications [11] and EIT ICT Labs [12], the projects “Persuasive services”, “Data-driven sustainability”, and “Data bridges” served, among other things, as test environment for the SCAM backend. Trials were made with some use cases where distributed resource and metadata management was required, e.g. when it was necessary to open up and manage heterogeneous metadata.

**TEL-Map**

In the EC-funded project TEL-Map [13] one of the objectives was to provide extended information about individual researchers’ projects and also collaborative (e.g. EC-funded) research projects. For this purpose metadata profiles were created and the combination of the SCAM backend with the Confolio frontend was used to collect, manage, and expose these metadata.

### 1.7 Scope and main target audience

The scope of this thesis is grounded in the software that was produced during the above mentioned applied research projects. It was necessary to carry out basic research and experimental development and this thesis gives an account of both how this research was carried out and what the research resulted in. The most important outcome is a generic information model that originated from the implementation work in several projects.

The main target audience of this thesis are researchers with a technical focus in the field of Web Science and architects of collaborative Web applications. This thesis argues that the proposed information model is generic enough to design and implement a wide range of Web applications. The application perspective is mostly ignored by the information model, no assertions are made regarding collaborative features or interaction design in general. Such design decisions depend on the application and its use cases which should not be approached by a generic architecture.

The goal of this thesis is to present an architecture that shows how knowledge artifacts are created, how knowledge effectively can be built upon using a proposed
generic information model, and how everything has been successfully applied in practice.

1.8 Terminology

Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>CSCL</td>
<td>Computer-supported Collaborative Learning</td>
</tr>
<tr>
<td>CSCW</td>
<td>Computer-supported Cooperative Work</td>
</tr>
<tr>
<td>DC</td>
<td>Dublin Core</td>
</tr>
<tr>
<td>DCAM</td>
<td>Dublin Core Abstract Model</td>
</tr>
<tr>
<td>DCMI</td>
<td>Dublin Core Metadata Initiative</td>
</tr>
<tr>
<td>DoW</td>
<td>Description of Work</td>
</tr>
<tr>
<td>FOAF</td>
<td>Friend Of A Friend</td>
</tr>
<tr>
<td>HTML</td>
<td>Hypertext Markup Language</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hypertext Transfer Protocol</td>
</tr>
<tr>
<td>ISO</td>
<td>International Organization for Standardization</td>
</tr>
<tr>
<td>JSON</td>
<td>JavaScript Object Notation</td>
</tr>
<tr>
<td>KW</td>
<td>Knowledge Worker</td>
</tr>
<tr>
<td>LD</td>
<td>Linked Data</td>
</tr>
<tr>
<td>LOD</td>
<td>Linked Open Data</td>
</tr>
<tr>
<td>LOM</td>
<td>Learning Object Metadata</td>
</tr>
<tr>
<td>MLR</td>
<td>Metadata for Learning Resources</td>
</tr>
<tr>
<td>OER</td>
<td>Open Educational Resources</td>
</tr>
<tr>
<td>OWA</td>
<td>Open World Assumption</td>
</tr>
<tr>
<td>OWL</td>
<td>Web Ontology Language</td>
</tr>
<tr>
<td>PLE</td>
<td>Personal Learning Environment</td>
</tr>
<tr>
<td>RDF</td>
<td>Resource Description Framework</td>
</tr>
<tr>
<td>RDFS</td>
<td>Resource Description Framework Schema</td>
</tr>
<tr>
<td>REST</td>
<td>Representational state transfer</td>
</tr>
<tr>
<td>SIOC</td>
<td>Semantically-Interlinked Online Communities</td>
</tr>
<tr>
<td>SPARQL</td>
<td>Protocol and RDF Query Language</td>
</tr>
<tr>
<td>SW</td>
<td>Semantic Web</td>
</tr>
<tr>
<td>TEL</td>
<td>Technology Enhanced Learning</td>
</tr>
<tr>
<td>URI</td>
<td>Uniform Resource Identifier</td>
</tr>
<tr>
<td>URL</td>
<td>Uniform Resource Locator</td>
</tr>
<tr>
<td>W3C</td>
<td>World Wide Web Consortium</td>
</tr>
</tbody>
</table>
WWW  World Wide Web
XML  eXtensible Markup Language

Project name changes

As mentioned above, the projects SCAM and Confolio have been renamed to EntryStore and EntryScape in 2012. In this thesis both the old and the new names are used, depending on which work at which point in time is referred to.

1.9 Outline

Chapter 2 provides an overview of the technologies and theoretical background of important concepts to position this thesis.

Chapter 3 describes a contribution-centric architecture by summarizing two applications in which the artifact concept was applied. The ReM$^3$ information model is introduced, followed by a discussion how artifacts can be based on it. The chapter concludes by relating the presented work to recent related developments and initiatives.

Chapter 4 gives an account of how the proposed contribution-centric architecture has been applied. Use cases are presented and it is reconsidered how knowledge is collectively constructed.

Chapter 5 summarizes the conclusions of the thesis, ties together the results of the published research, and provides pointers to possible future directions of research and development.
Chapter 2

Theoretical background and technologies

The work described in this thesis is mostly of technical character and is situated within Web architecture, in particular Semantic Web research and Linked Data. This chapter provides an overview of the most important concepts and related work to position this thesis. It does not critically discuss the theoretical underpinnings of the Semantic Web and related technologies, but focuses on their intersection with collaboration and how the important theoretical concepts in this intersection can be defined. The result is an approach that is rather technical in nature.

2.1 Web Architecture

World Wide Web

The World Wide Web (WWW) is a network of interlinked documents that are formulated in the Hypertext Markup Language (HTML) [14, 15] and can be retrieved using the Hypertext Transfer Protocol (HTTP) [16]. Every document on the WWW can be uniquely identified and fetched using its Uniform Resource Locator (URL) [17]. A URL is a special case of a Uniform Resource Identifier (URI) [18] where the identifier also is a locator which enables the identified entity to be retrieved from the stated location. The WWW is the primary technology that incorporates the OWA as one of its fundamental characteristics with its underlying technologies HTML and HTTP (among others). It provides a means for creating links between everything, but in “standard HTML” (excluding Semantic Web techniques such as RDFa) there is no machine processable way of refining the semantics of the links.
and a machine would have to rely on heuristics. Due to the lack of formalized relationships and definitions of entities there is no way to express machine processable semantics for the links.

Semantic Web and Linked Data

The Semantic Web (SW) builds on the “Web-way” of constructing knowledge and can be seen as an evolution of the “classical” WWW towards a Web that allows machines to aid humans in processing the contained knowledge. Information expressed in the main format of the Semantic Web, the Resource Description Format (RDF), allows for machine-processability and opens up for advanced information processing. RDF, as well as HTTP and HTML, is a recommendation developed by the World Wide Web Consortium (W3C). The data model of RDF is based on statements where each statement consists of a subject, a predicate, and an object. The subject is a resource, the object can be a resource or a literal, and the predicate denotes the relationship between subject and object, very similar to human natural language. Statements are also called “triples” in RDF, and a set of RDF triples is an RDF graph.

The ability to make statements about resources with RDF makes it very suitable for knowledge representations. Data fulfill the requirements for Linked Data [4] if they are identified by a URL, retrievable by HTTP, expressed in RDF and contain links to other documents that are expressed and exposed in the same way. Linked Data are deemed to be the part of the SW that is most relevant for the framework discussed in this thesis. Linked Data shifts the focus away from logics, which were considered necessary in early SW research. Instead, the focus lies on publication and consumption of interlinked structured data and builds on standard Web technologies such as URIs and HTTP. LD builds on the SW-stack, but focuses on pragmatically employing semantic technologies.

Open World Assumption

The Open World Assumption (OWA) is a concept in the field of knowledge representation [19] where it means that it is assumed that no one entity has complete knowledge about something. Under a closed world assumption, any statement can be evaluated to be true or false; this is not the case under an OWA as it cannot be guaranteed that all facts are known or have been discovered. On the Web, where all content is published under an OWA, anyone can write anything about anything and it cannot be assumed that any one entity has knowledge about everything said. Equivalently, in the context of the Semantic Web, an OWA means that any state-
2.2 Concepts

Resources

This thesis and its underlying papers discuss “resources” in several contexts which makes it necessary to provide a proper definition for this rather vague concept. Within Computer Science, the most common conception of resource is a regular file or a link to content on the Web. A more generic perspective includes concepts, books, physical persons, events, etc. Because of the focus on Web technologies in this thesis it makes sense to adhere to the definition used by the W3C Technical Architecture Group (TAG) [24] as stated in the Architecture of the World Wide Web volume one [25]:

By design a URI identifies one resource. We do not limit the scope of what might be a resource. The term “resource” is used in a general sense for whatever might be identified by a URI. It is conventional on the hypertext Web to describe Web pages, images, product catalogs, etc. as “resources”. The distinguishing characteristic of these resources is that all of their essential characteristics can be conveyed in a message. We identify this set as “information resources”.

According to this definition, both information resources as well as other resources (without digital representation) can be identified with URIs and therefore can be referred to in various settings.

Artifacts

The artifact concept as it is used in this thesis was first discussed in paper 1 “Collaborative Construction of Artifacts”. Artifacts are the central point of discussion...
because of the limitations that synchronous or turn-based collaboration implicate. Paper 1 focuses on situations where collaboration is asynchronous with a small risk for inconsistencies. A principle that builds the foundation for this approach is the separation of the artifact from its expression in a container, i.e. artifacts as unions of separate contributions. Contributions have no separate identity, but are assumed to be uniquely identified by the container they are expressed in and the artifact’s identifier. There are no dependencies between contributions, only between artifacts and containers in which contributions are expressed. In the case of a text document, the document is the artifact and the container in which the document is expressed is the file. Breaking the one-to-one connection between document and artifact allows multiple contributions to reside at distributed locations, which was successfully demonstrated with conceptual models being treated as artifacts.

To turn the problem upside down, instead of providing a separate rich application where collaboration can take place, collaboration is tracked wherever it takes place. The only requirement is that the collaboration is centered around something, i.e. the artifact, which has a global identifier, e.g. in the form of a URI. It is up to the application to bring together the contributions to form a view of the artifact and to integrate collaboration facilities if necessary. How the integration is realized will differ depending on what is collaborated around, e.g. collaboration around a text document is different from collaborating around a folder. The main assumption is that collaboration is emerging from individual contributions that can be merged into a whole. Another assumption is that the contributions can be stored separately.

To summarize, an artifact is defined by the following characteristics:

- An artifact is separate from its expression in a container.
- An artifact is a union of a set of separate contributions.
- An artifact’s contributions are independent from each other.
- An artifact is uniquely identified (e.g. by a URI).

This thesis, in particular paper 1, discusses artifacts in the context of conceptual models, where the artifact has an identity which holds a range of contributions together (see section 3.1 for details).

**Collaboration and collective knowledge construction**

In the beginning of the work for this thesis, the concept of collaboration was the one that the author worked with. Conceptual models were being created collaboratively,
even though without having write-access to the same model. It was possible to collaborate around certain topics by just contributing to, or adding to, an existing conceptual model.

Later on, the projects in which context this thesis was carried out included also use cases where contributors eventually have a more distant relationship with the original author, i.e., they don’t know each other and neither are they familiar with each other’s goals of the work. Such use cases required support for situations in which new knowledge is built on top of already existing knowledge, even of seemingly distant origin. E.g., in Organic.Edunet it was about the reuse of resources from any kind of repository or archive to contextualize and create educational resources.

This thesis refers to collaborative situations with different degrees of “looseness”. Examplified this means that if a developer builds a conceptual model of a system architecture for his colleagues to comment on, everybody involved probably has the same view of the problem and the same goal, namely to design the system architecture of their system. They collaborate around the same thing, and the colleagues contribute to the developer’s model. In another example, a user publishes a conceptual model of a political topic, similarly to an article on Wikipedia. No explicit contributors are known or expected, but eventually somebody adds to the original conceptual model. A contribution is published, but there is no coordination and there doesn’t have to be an agreement between the original and the contribution. The term “loose forms of collaboration” is used in this thesis to describe atypical forms of collaboration that are carried out when collaborating using a contribution-centric architecture.

2.3 Web applications and Open World Assumption

The Web architecture and its enhancements provide a fundamental technology stack and build a foundation for Web applications, but require an additional application layer to actually make use of these technologies. The following subsections provide an overview of a set of technologies and applications that are briefly characterized with respect to semantics among contributions. The list of applications below does not aim for being complete, but provides an overview of prime examples within the field.
CHAPTER 2. THEORETICAL BACKGROUND AND TECHNOLOGIES

Wikis

A wiki is a Web application that allows users to create, modify and delete content, i.e., wiki pages. Most wikis follow a collaborative approach where every page is contributed to by multiple users during its lifetime. Wikis enforce consensus as there is only a single synchronized view of every page, i.e. there is no support for multiple views of the same topic. The primary example of a large-scale wiki is Wikipedia and, disregarding the synchronization requirement, it represents the biggest Web applications for collective knowledge construction. The SW counterparts of Wikipedia are DBpedia and Wikidata. DBpedia provides a “semantified” view of the formalized knowledge on Wikipedia by extracting the structured data from infoboxes. Wikidata on the other hand is an own system in which everyone can edit the data, equivalent to the Wikipedia principle. Instead of converting the Wikipedia data, Wikidata maintains a separate database with the goal of creating Wikipedia infoboxes from the data on Wikidata. Wikis serve as an example for a synchronous OWA. Everybody is allowed to contribute, but the collaborative result can never be distributed or inconsistent, it has to be consolidated.

Rahhal et al [26] described multi-synchronous semantic wikis (MS2W) as different from synchronous and asynchronous ones in a way that multi-synchronicity allows “to manage multiple streams of activity instead of giving the illusion of one stream” and “simultaneous work in isolation and later integration of the contributions”. However, the MS2W requires that every user declares all collaborators in advance to ensure convergence and consistency. MS2W follows an approach that is restricted to wikis and requires consistency but not direct synchronicity. MS2W has been implemented as an extension of Semantic MediaWiki [27] with the goal of respecting the simplicity of publishing content on wikis while at the same allowing to construct a rather decentralised network of semantic wikis.

Blogs

Weblogs (blogs) can be seen as a collection of published documents (blog posts) on the Web. Typically a blog post is written by a single author, and multiple authors can publish content on the same blog. The posts have, like wiki pages, a synchronized character, but do not require consensus. Additional functionality such as the possibility to comment on posts, allows to extend every blog post with additional information. Even though it is possible to restrict the comments to user accounts of the local system, it is common to allow anyone to post comments. A special form of comments on blog posts are linkback mechanisms such as pingbacks and trackbacks, i.e., automated ways of informing authors that their blog posts have...
been mentioned in or referred to from other blog posts. Such techniques allow that referenced blog posts show links to posts that comment on or develop them further. Linkbacks provide a good notification mechanism on the Web, but are only used in the context of blogs and lack the possibility of adding semantics (“agrees with”, “enhances”, etc) to the relationships between posts. The possibility of linking to blog articles and taking advantage of linkback mechanisms to add comments which do not reside in the same system as the original blog post shows that blogs are not in disagreement with the OWA.

**Forums**

Web forums are Web pages with an extended ability to comment on forum pages (posts) and their comments. After the original post is published, other users write follow-up comments and the discussion evolves in threads. External users are usually not allowed, which limits the possibilities of interlinking related pages. Semantic information between posts and their comments is usually not maintained or exposed. If the backend holds such information, it could be embedded directly in each Web page using technologies such as RDFa [28]. A discussion thread in a forum corresponds to an artifact built out of separate contributions (e.g. replies to the original post). However, all contributions are managed in the same container, namely the discussion thread (which may be paginated). It is not possible to contribute to a discussion without posting to the thread using the dedicated forum application, which is why an Open World cannot be assumed.

**Annotea**

Annotea [29] is an extensible technology building on the Semantic Web, and developed by the W3C. The primary goal is to support enhanced collaboration through shared metadata and different views. It provides a mechanism for adding comments inside existing Web documents such as HTML and XML. The approach allows to point into existing Web pages to add annotations. Such annotations are typically small chunks of text that are identified and described with metadata expressed in RDF. The biggest direct benefit from the use of SW technologies is for Koivunen that “the user generated metadata can be easily combined and reused in many other applications” [29]. Annotea requires client-side support (ideally in a Web browser) for editing, publishing and fetching annotations from one or more repositories and to include annotations in the right place. Multiple annotations can exist for every document and pointer; contradictory information cannot be prevented as there is no mechanism to enforce synchronization or consensus.
Comments on a web page are basically contributions which create an artifact, which brings Annotea close to a contribution-centric approach. There are a number of applications based on the principles of Annotea [30], e.g. multimedia annotation and collaborative bookmark management. Annotations are made to content within documents (using XPointer [31]), rather than to support linking between documents or between documents and metadata.

2.4 A contribution-centric approach

How can knowledge be constructed under an Open World Assumption? A contribution-centric approach without compulsory collaborative features such as common goals and coordination allows for flexible applications that can be categorized as collaborative. Collective knowledge construction can be explained by construction of artifacts using the contribution model which is at the core of this thesis.

Blogs with linkback mechanisms, multi-synchronous semantic wikis and Annotea follow an OWA-related approach. However, they are only used pragmatically in their primary use case, i.e. blogs are used to publish blog posts and connect them with other blog posts, wikis are used to publish mostly textual web content, and Annotea focuses on annotations. Analyzing the abilities of the collaborative technologies above allows the conclusion that there is a lack of models that allow the implementation of a contribution-centric architecture that is not restricted to a certain use case. Annotea is the technology that is closest to the idea of collaboratively constructed artifacts as described in paper 1. The major distinction is that Annotea focuses on annotations rather than on annotated documents. The artifacts in paper 1 can therefore be seen as complementary to Annotea which follows an approach that does not require the typical characteristics of collaboration, but is restricted to a certain way of annotating information and is not really prepared for generic applications.

Semantic Web research tends to lack a discussion on how collaborative applications can be realized, by taking advantage of the capabilities of SW-based architectures, and going further than just storing its (meta)data in RDF. This thesis describes an architecture that provides a working implementation of a contribution-centric approach. Also, this architecture natively supports Linked Data and provides a means to develop Web applications based on SW-applications. In addition to the artifact concept and an information model, some use cases and an evaluation of these show whether the proposed approach works in practice.
Chapter 3

A contribution-centric architecture

This chapter focuses on the core concepts and properties that build the foundation of the proposed architecture. The first part describes two applications where the artifact concept was applied. After that, a metadata management framework and the resulting Resource and Metadata Management Model (ReM³) are described. The chapter concludes with a discussion on how artifacts can be based on the information model, together with Web technologies with a focus on Linked Data.

The previous chapter concluded that there is a lack of models that allow the implementation of a contribution-centric architecture where artifacts are a result of separate contributions and different views.

This chapter describes a contribution-centric architecture that consists of two cornerstones: artifacts and an information model for managing resources and their metadata. The architecture builds upon the ideas and research around knowledge artifacts that are described in papers 1 and 2, and integrates it with the information model that was conceived in papers 3 and 4 and thoroughly elaborated in paper 5.

3.1 Artifacts applied

Conceptual models as artifacts

In paper 1 the object of demonstration that collaboration based on artifacts works is the conceptual modelling tool and concept browser Conzilla [32, 33, 34, 35] and its collaboration server Collaborilla [36, 37]. Collaborilla makes it possible to construct collaborative views that are perceived as merged instances of data. The approach of viewing an artifact as a set of contributions made is necessary to decide on some requirements which, in the case of Collaborilla, are specific for conceptual models.
As listed in paper 1, these are:

- Contributions have no separate identity but are assumed to be uniquely identified by the artifact’s identifier and the containers they are expressed in.

- There are no dependencies between contributions, only between artifacts and containers in which contributions are expressed.

- The artifact’s contributions - indicated via containers - are ordered.

- Metadata is available both on artifacts and containers.

A more detailed description of the application in practice and how e.g. conceptual models are treated as artifacts follows in chapter 4.

Tasks as artifacts

As a continuation of the collaborative construction of artifacts, paper 2 explores the applicability of artifacts to a different area, namely collaborative tasks. The focus of this paper lies on knowledge workers (KWers) that are embedded in teams with common organizational goals. Collaborative tasks require that participating KWers share information to achieve a common task goal. There is lack of widely-adopted solutions to collaboratively edit structured task information which makes it necessary to think of new approaches.

The proposed solution applies, along the lines of Conzilla and Collaborilla, a wiki principle to structured content, to enable a contribution-based model focusing on artifacts. Paper 2 presents a combination of the task management client Kasimir [38, 39] and the collaboration server Collaborilla. The approach is appealing because artifacts and contributions enable collaborative tasks while the semantics of the task information is preserved. Modifications are plain contributions and can be handled concurrently. From the collaborative perspective there is no difference between conceptual models and tasks. A KWer can contribute to a task without changing the original one. It is up to the client to merge various contributions into one task artifact.

Also here Collaborilla contains two metadata types:

1. A registry for tasks and their contributions, which is a mapping between the task identifier and a contribution and the location from where the contribution can be loaded.
2. Metadata about a published task or contribution – it helps the KWer to make a decision whether to take a task contribution into account or not, e.g., for a task contribution the contributing person and time is relevant metadata.

A task is identified by a URI, but contributions have no separate identity. There are no dependencies between contributions since all of them are expressed independently of each other. A contribution’s only relationship is with the corresponding task, i.e. the artifact.

The application of the artifact concept to collaborative tasks was described in paper 2. Even though both the collaboration server Collaborilla and the task management application Kusimir exist as software, an integration has never been implemented. The described outcomes are purely theoretical and Conzilla together with Collaborilla remain the only applications based on collaborative artifacts as described in papers 1 and 3.

3.2 Management of resources and their metadata

Artifacts as enabled by Collaborilla support contributions for conceptual models, tasks, etc, but there is neither a specific focus on resources nor a discussion around interoperable expressions on the Web using standard Web technologies. Such management of resources and their metadata is the central topic of papers 3 and 5, with paper 4 describing an application within the Organic.Edunet project.

A mashup-friendly framework

Papers 3 and 4 describe a set of requirements that focuses on establishing a generic and mashup-friendly framework for resource and metadata management. Typically, a mashup combines several data sources into one user interface. In the best case this results in a Web application that provides an added value compared to its individual components. Mashups build on Application Programming Interfaces (API), standard protocols such as HTTP, and standardized data formats such as JavaScript Object Notation (JSON) and RDF. To make mashups possible, a generic way of managing resources and metadata is beneficial as solutions can be made less specific for each situation. There is a need to differentiate resources, their metadata, and administrative information, as well as digital from non-digital resources. The targeted applications include tools for resource annotation, e-portfolios as well as Personal Learning Environments (PLE). A generic design enables the use of the same framework across use cases and the actual materialization of a tool becomes a
CHAPTER 3. A CONTRIBUTION-CENTRIC ARCHITECTURE

matter of design decisions at the user interface layer. The resulting SCAM framework is the outcome of this generalization and is described in paper 3. Collaborilla was specifically designed to support artifacts, and was tested and evaluated with collaborative conceptual models. Unlike the ideas around artifacts from paper 1 and 2, SCAM operates on a higher level and enables mashups by linking information across systems. Paper 3 concludes that SCAM can act “as the least common denominator between mashed up applications regarding resource and metadata management” as it provides “a unified mechanism of accessing the managed resources and its descriptive information, which might be (re)used by any number of tools”.

Information management through entries

The concept of an *Entry* is a central contribution of this thesis and was introduced in paper 3 and refined in paper 5. It holds all information that is necessary to link together a resource, its metadata, and administrative information such as access control and provenance. It is the elementary building block on which everything is based inside SCAM, which can be seen as a repository that manages entries and gives different meaning to entries with different purposes. E.g., an entry can hold together a simple file with some descriptive information, or it can describe a whole folder with resources. The use cases are numerous. Entries are formalized in RDF as named graphs and exposed via a REST API [40].

A detailed specification of the information model that is based on entries is the main subject of paper 5 and is summarized in section 3.3. This chapter also explores the relationship of entries to previously introduced concepts of resource, artifact, and collaboration.

Resource annotation with metadata

SCAM version 4 was initially implemented and used within the Organic.Edunet project [41, 42]. The primary use case in Organic.Edunet was the annotation of resources with metadata, the processes are described in detail in paper 4 and summarized in chapter 4. The Web application Confolio was built as a user interface and abstraction of SCAM’s capabilities, and was the tool that was used to annotate educational resources with metadata.

Due to the distributed character of the Organic.Edunet project (it was designed as a federation of repositories) some specific requirements were the capability to manage distributed and heterogeneous metadata, and also support for the contextualization of such metadata for different purposes. These requirements are not
limited to the Organic.Edunet project, in contrary, they fit the generic approach of SCAM and provide a good use case for the information model that emerged from the various iterations of SCAM (and later EntryStore). The information model is the main topic of paper 5 and also described in the next section.

3.3 An information model to support a contribution-centric architecture

This section mostly summarizes the Resource and Metadata Management Model (ReM$^3$) as described in paper 5. ReM$^3$ is a result of the continued development of the SCAM framework and provides a formal specification of the information model. ReM$^3$ emerged in projects with a focus on the publication, enrichment and management of heterogenous metadata. The main purpose was to annotate resources with metadata and to curate them into collections, independently from the content repository they originate from. A multitude of requirements led to several problems that were described along with proposed solutions in paper 5.

The model’s characteristics, originating from the problem statements and conclusions in paper 5, can be summarized as follows and form the corner stones of the information model:

- Everything is centered around “entries”, which are used to keep track of information. A more detailed explanation is given in the conceptual overview.

- All entities in the information model are expressed in named graphs which are identified by dereferencable URIs (i.e., an entity can be fetched from its URI). The relationships between those entities are expressed in the entry information.

- ReM$^3$ allows a distinction between situations where either both resource and metadata, only metadata, or neither metadata nor resource are handled in the system.

- The same resource can be annotated with different metadata, allowing for contextualized use.

- Entries contain statements to manage resources and their metadata and to keep track of all involved pieces of information. Unique identification and linking allows for exposing the combination of resource and metadata according to the Linked Data principles [4].
The RDF data model [43] is used as a common carrier. The ReM³ entry together with RDF allows the integration of heterogeneous metadata as well as enrichment of metadata originating from other sources.

A more detailed conceptual overview is provided in the following sections.

Conceptual overview
The main purpose of the ReM³ information model is to keep track of resources and their metadata. Entries are managed inside of contexts which are containers for sets of entries. An entry consists of a resource, metadata, and meta-metadata with administrative information which is called “entry information” in ReM³. See figure 3.1 for a conceptual overview of a ReM³ entry.

![Figure 3.1: ReM³ Entry and its linked information](image)

The cardinalities between the entities above are 1:1.

Each entry holds three different types which determine how the entry is treated:

1. The Entry Type indicates the location of the entry’s resource and metadata.
2. The Resource Type determines whether there is a digital representation of the resource.

3. The Graph Type is used to manage special cases of entries which should receive a special treatment within the implementation of the model.

The entry type has four possible values, which are described in paper 5 section 3.1 with:

1. **Local**: both metadata and resource are maintained in the entry’s context.

2. **Link**: the metadata, but not the resource, are maintained in the entry’s context.

3. **Reference**: the resource and the metadata are maintained outside the entry’s context.

4. **Link reference**: the resource and the metadata are maintained outside the entry’s context, in addition there are complementary metadata maintained in the entry’s context.

The information model is based on RDF and relies on the use of named graphs [44, 45]. See figure 3.2 on page 33 for an RDF schema of ReM³. ReM³ manages all information in named graphs and provides access to them through dereferencable URIs. Provenance can be kept track of using the named graph URIs and the entry information. ReM³ supports both agent-centered and object-centered provenance, i.e., the model keeps track of information about which users were involved in creating or modifying, as well as the origins of a resource or its metadata. Similarly to provenance, access control is also expressed in the entry information. It is expressed as a set of read and write permissions per entry, with a context being a special case of an entry managing a graph resource. A context’s access control is inherited by all contained entries.

A detailed conceptual description of ReM³, including a thorough discussion of the use of named graphs, provenance and access control can be found in paper 5 chapter 3.

**Contributions and contextualization**

ReM³ allows non-invasive contributions by creating a new entry that refers back to the original entry that it is built upon. Three different entry types (“link”, “reference”, and “link reference”) can also determine that information originates
from elsewhere. The most complex entry type, link reference, allows the use of the original resource in a different usage context by making it possible to add additional or different metadata while preserving the original metadata. E.g., the same video can be used in different lectures, depending on what to focus on, the target group or other aspects.

Contributions to entries can happen in parallel (several contribution entries referring to the same original entry) or in dependency of each other in cases where contributions are built on contributions. Similarly to the approach taken in Collaborilla, contributions can be merged and viewed as a whole, or viewed separately in various combinations. ReM³ manages the original metadata (in the model called “external”) separate from the contextualized metadata. Metadata are maintained inside their own named graphs and are only merged within the presentation layer in the client application if the latter is designed to do so. Managing everything in separate graphs and with separate identifiers has the benefit of making it possible to keep track of the provenance of these information entities.

The non-invasive character of contributions necessitates a different approach for finding contributions. It is not possible to link from the entry to the contribution if the author of the comment has insufficient access rights for modifying the entry he commented on. The link can only exist in the other direction, from the contribution back to the original. For this purpose, the “inverse relational cache” was designed and implemented. This cache provides a powerful way of managing information in distributed systems and can be used in many different use cases. E.g. to be able to comment on an entry, a user does not have to have write-access to the entry she wants to comment on. Instead the comment is created in the user’s own context and refers back to the commented entry. Lists of entries can be built the same way, entries refer back to their parent entry and build a hierarchy. However, the inverse relational cache does not cover incoming links from external sources; for this to work it would be necessary to support the corresponding protocols to perform “linkback”-actions such as Pingbacks [46] or Trackbacks [47].

ReM³ keeps track of such relations in its own named graph, one per entry. This feature of the information model works like the rest of the WWW where write-access cannot be assumed and where it can be an expensive and tedious task to find relationships between documents. ReM³ exposes this information using Web standards, primarily HTTP and RDF.
3.4 Artifacts based on entries

The idea about artifacts as results of separate contributions and different views was first presented with the collaboration server Collaborilla in paper 1. This idea is still the foundation of the proposed architecture, but the ReM³ information model is a step further in relation to the original artifact idea.

Collaborilla is based on contributions around content, e.g., conceptual models, and everything is stored in specific container files. As an extension of this approach, ReM³ allows for contributions around uniquely identifiable concepts; everything that can be identified using a URI can be published, referred to and built upon. Artifacts can be created using ReM³ entries, through linking resources and their metadata, and providing the possibility to build entries upon entries using the link reference entry type. The inverse relational cache allows contributions even to read-only entries and other contributions can be found by looking up inverse links from a specific contribution to the entry.

The primary use cases in the beginning of developing ReM³ and SCAM within Organic.Edunet were resource annotation and metadata harvesting. Both use cases demanded that it should be possible to annotate a resource with metadata, independently from the resource’s location. Another requirement was that the history of a resource, its metadata, and all its contributions should be traceable. Whether or not contributions should be merged into a single view should be up to the user or user interface, but it should not happen inside the information model.

Contributions can be expressed on several contextual levels, using either explicit (strong) or implicit (weak) relationships:

- Contributions to the original identity of a resource, i.e., an entry that builds an explicit relationship between a resource and its metadata. An example is a picture that is annotated with metadata to serve as educational material.

- Contributions to contributions, i.e., an entry that builds an explicit relationship between an entry and additional metadata. An example is an already annotated picture that should be contextualized for the use in a different lecture or course.

- Relations expressed in the resource’s metadata, by using e.g. owl:sameAs, dc:relation, dcterms:hasPart or similar predicates. This way of expressing relationships demands more specific knowledge about the metadata expressions in use, whereas the previous two ways of expressing contributions are built into and natively supported by ReM³. Relations in the metadata can be ex-
pressed explicitly, but may carry implicit relations defined in their underlying schema or ontology.

- Entries that potentially contribute to the same entity, but with weaker implicit relationships (e.g. common tags in their metadata) instead of explicit links. An example is educational material tagged with the same course code.

A practical description of how this is used in practice in real-world projects including an evaluation can be found in chapter 4.

The reference implementation of ReM³ can be used to support artifact constructions, but is more flexible with respect to Web architecture, resources and metadata. ReM³ together with the inverse relational cache can act as a collaboration server supporting artifacts in the style of Collaborilla, but is restricted to contributions within the same server instance.

3.5 The use of Web technologies and Linked Data

ReM³ has been implemented using standard Web technologies in its reference implementation EntryStore. The information structure with its RDF schema based on named graphs is suited for a REST-ful HTTP API [40] and a Linked Data-based implementation.

The basic principles of ReM³ in the reference implementation are:

- The API provides one REST resource per named graph, e.g. entry, metadata, etc.
- The REST resources support all or several of the HTTP verbs, where appropriate.
- RDF is used as a common carrier for metadata.
- SPARQL [48] endpoints, both globally per installation and per context, support querying of public metadata.
- Additional query interfaces such as OAI-PMH [49] and SQI [50] are supported and contribute to an integration into legacy infrastructure.

The recommended way to apply ReM³ in applications is via its REST-ful interface. Lots of modern Web applications make use of JavaScript which can take advantage of the REST-ful approach for accessing and updating data.
3.6. RELATED TECHNOLOGIES

A detailed technical description of the use of Web technologies and APIs can be found in paper 5, paper 3, the EntryStore wiki [1], and the ReM³ specification document [51, 52].

3.6 Related technologies

On a similar subject, the Semantically-Interlinked Online Communities (SIOC) ontology [53, 54] allows to link online community sites using Semantic Web technologies and contains some classes that can be related to ReM³ classes. More specifically, the Container and Item classes fulfill the same role in SIOC that Context and Entry have in ReM³.

The W3C hosts a Linked Data Platform (LDP) Working Group [55] whose declared mission is to “produce a W3C Recommendation for HTTP-based (RESTful) application integration patterns using read/write Linked Data”. The group charter contains a list of technical issues (see section 2.1 in [56]) which should be solved in the context of the work of the LDP WG. Several of the listed issues are solved in ReM³ and EntryStore can be considered as an implementation of the LDP. A detailed description of how ReM³ and EntryStore relate to the LDP will be the topic of a future publication.

3.7 Summary

The proposed architecture builds on the artifact concept from papers 1 and 2, and goes beyond it based on the work described in papers 3 to 5. The described contribution-centric architecture provides the foundation for enabling collective knowledge construction. The main characteristics of ReM³ are the ability to manage resources and metadata together, an inverse relational cache that keeps track of links from the contribution back to the original, and responsibility of the client to merge various contributions into one artifact. The underlying technical infrastructure does not assert much about collaborative features, this is completely up to the application and the use case.

Collaborilla allows the tracking of contributions around content, whereas ReM³ take a more flexible approach by tracking contributions around uniquely identifiable concepts. ReM³ is designed for interlinked information in heterogeneous environments, but also allows for the implementation of collaborative applications in more restricted environments.

Table 3.1 on page 32 compares the Web applications as summarized in section 2.3 with the characteristics of the proposed contribution-centric architecture (as im-
implemented by ReM³). The intention is to highlight the principal differences between ReM³ and established approaches of knowledge construction in Web applications with respect to the OWA. The characteristics as highlighted in the table originate from the discussion of how ReM³ relates to artifacts in a contribution-centric architecture as described in the sections 3.2, 3.3 and 3.4.

<table>
<thead>
<tr>
<th>Data and metadata are separate and treated in a unified way</th>
<th>Blog</th>
<th>Wiki</th>
<th>Forum</th>
<th>Annotea</th>
<th>ReM³</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contributions can be persistently and globally identified</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contributions can be added to uniquely identifiable concepts</td>
<td></td>
<td></td>
<td></td>
<td>(X)¹</td>
<td>X</td>
</tr>
<tr>
<td>Contributions are separate from the original document</td>
<td>(X)²</td>
<td>(X)³</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>The history (provenance) of an artifact and its contributions is traceable</td>
<td>(X)⁴</td>
<td>X</td>
<td>(X)⁴</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>May result in different views consisting of separate contributions</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>

¹ The concept that is commented on (“annotated”) is always contained in a web page.
² Comments made through linkbacks are in fact posts in a blog.
³ Contributions do not modify the original post, but belong to the same discussion thread.
⁴ This is implementation-specific.

Table 3.1: Comparison of Web applications and ReM³.
Figure 3.2: ReM* terms described in an RDF schema (published in paper 5)
Chapter 4

Use cases for the information architecture

This chapter gives an account of how the proposed information architecture has been applied and illustrates the potential for new applications.

The use cases are discussed from three viewpoints:

- Overview, i.e., the type of work being carried out in a typical use case.
- Collective knowledge construction, i.e., how collaborative processes are supported.
- The usefulness of the proposed architecture.

4.1 Collaborative conceptual modeling

The conceptual modelling tool Conzilla is a concept browser to model and navigate complex information structures. In [35] it is described how a concept browser can be used to improve knowledge management by providing a means to structure and filter information. Conzilla is not only a browser, it is also an editor to connect concepts and build relationships between them. This relation-building work results in so called Context-maps, in which information and content are tied to concepts and concept-relations. The same (reused) concept may be described using different metadata in different Context-maps which provide different contexts. The standard modeling language is part of the Unified Modeling Language (UML) [57], but also
the more human language-oriented “dialogue mapping” [58] is supported. Figure 4.1 is a screenshot of a typical Context-map\(^1\) presented in Conzilla.

Conzilla offers two approaches to navigate between Context-maps: contextual neighborhoods [35] and hyperlinks. By reusing the same concept in several different Context-maps a contextual neighborhood is implicitly created. It cannot be modified directly, it is solely defined by the reuse of concepts in different Context-maps. Similarly to the WWW with its HTML, hyperlinks are explicitly created and modified by the user. Hyperlinks function complementary to contextual neighborhoods; a link (the blue-colored concepts in figure 4.1) may originate from a concept and point to a single Context-map, whereas a contextual neighborhood (see the activated menu in figure 4.1) may affect many Context-maps and provides a means of keeping track of reused concepts and Context-maps in which perhaps similar topics are conceptually modelled.

Other publications such as [33, 34, 35, 36, 37] and the Conzilla website [10] provide good entry points into conceptual modeling and the creation of Context-maps with Conzilla.

**Collective construction of knowledge**

The theory and implementation of collaborative creations or add-ons to Context-maps is the core of paper 1 “Collaborative construction of artifacts”. Conceptual models are treated as artifacts consisting of various contributions of different origin, comparable to creating knowledge wiki-style and, under an OWA, practicing how information is expressed on the WWW. The separation of the information (metadata) from the graphical presentation allows the reuse of concepts in different contexts and with different metadata. The unique identification of relevant entities (in Conzilla: concepts, content, and context-maps) and their representation in RDF opens up for collaborative processes across system borders. Contributions are non-invasive and therefore separate from the Context-maps they build upon, i.e., contributions are never physically merged with the original. The addition of a contribution to a Context-map is carried out directly in the RDF model by merging the graph representations into an artifact. Paper 1 provides the background for this approach.

A typical use case is that *User A* creates a Context-map to which *User B* adds some information such as additional concepts, concept-relations and metadata. The workflow for such a use case can be summarized as follows:

\(^1\)Context-map “TEL-Map Roadmapping and Awareness Building” by Ambjörn Naeve.
4.1. COLLABORATIVE CONCEPTUAL MODELING

Figure 4.1: Screenshot of a Context-map with link and contextual-neighborhood

1. User A creates and publishes a Conzilla Context-map using the Collaborilla collaboration server.

2. User B loads this Context-map by entering its URI and connecting to the collaboration server.

3. User B adds additional concepts, concept-relations and metadata in her own Conzilla session.

4. User B publishes the modifications to the collaboration server.

5. User A reloads the map and may then choose to view his Context-map together with User B’s additions.
Figure 4.2 provides a screenshot where the original Context-map can be seen along with an “activated” contribution. The viewer chose to virtually merge the contribution with the Context-map by ticking the according checkbox. The color of the contribution in the screenshot is red.

Context-map “Causes of Terrorism” by Ambjörn Naeve, contribution by Matthias Palmér.

Figure 4.2: Screenshot of a Context-map with a highlighted contribution

Usefulness and other possible applications

As can be seen from the use case description above, there is no access control or any other explicitly created list of collaborators. If the URI of a published Context-map is known, anyone can load it and contribute to it. The original Context-map is never modified by a contributor, instead a contribution builds upon the original and adds to it. It is the viewer who decides about which contributions to activate and display, and therefore what the resulting view of the artifact consists of. This approach is supported by the underlying information model that builds upon a graph representation in RDF. It also follows the principles of how information is published on the WWW, under an OWA, where anyone can link to anything. Translated into collaborative conceptual models this means that anyone can contribute to any conceptual model and publish the result. However, if User A decides
4.2. COLLABORATIVE ANNOTATION

to move the concept that has been added to by User B, then User B has to perform a corresponding update of the addition. This is a restriction that exists in the current version of Conzilla which may change in the future.

Paper 2 contextualizes the same approach to enable collaborative task management where artifacts are collaborative tasks instead of collaborative conceptual models. The paper focuses on knowledge workers and how they may create collaborative tasks by sharing a common set of information such as the task’s goal and status as well as related resources such as files, involved persons and emails. As with collaborative conceptual models the task management is based on handling metadata which requires the task to be serialized into a machine-readable form, e.g. an RDF representation. The solution proposed in paper 2 is to apply a wiki-style information management to structured content such as collaborative tasks. Even this approach is based on the principles of collaboratively constructed artifacts, but unlike the implementation of the collaborative capabilities of Conzilla as described in paper 1, no prototype has been developed for collaborative task management. However, the theoretical description of how collaborative tasks (similar to collaborative conceptual models) may be implemented using artifacts, along with the existence of the collaboration server Collaborilla, are promising to build even other applications based on the same approach.

4.2 Collaborative annotation of educational resources

The Organic.Edunet project was the main driver behind the development of the web application Confolio and its backend SCAM. The project’s goal was to provide a federation of learning repositories, providing a means to upload or link to, and in a later stage even curate, a large number of educational material in the context of organic agriculture and agroecology.

Confolio was built on the idea of portfolios for individuals (and in more recent versions also groups). Portfolios are used to manage resources, which can be uploaded files, web content, physical entities or abstract concepts. Resources are always combined with metadata consisting of different properties depending on the use case. In figure 4.3 a screenshot is shown of a portfolio view in a recent version of Confolio (EntryScape); resources are listed on the left and the metadata of the selected resource is shown to the right along with a preview.

The Web application is metadata-heavy, which means that creation and presentation of resources always include the aspect of annotating the resource with metadata. The choice of the applied metadata profile is based on several factors (as also described in paper 5):
CHAPTER 4. USE CASES FOR THE INFORMATION ARCHITECTURE

Figure 4.3: Screenshot of a portfolio view in EntryScape

- entries may represent different things, for example web pages or physical objects.
- entries may be described for different purposes and different target groups.
- entries may originate from different information sources which use different standards.

EntryScape relies on the RForms library [59] to generate a user interface for presentation and editing of metadata.

The ReM³ types (summarized in section 3.3) are important features of the ReM³ information model and are heavily used in EntryScape. Most important for the management of distributed information is the Entry Type, which indicates the location of the entry’s resource and metadata. The use of Entry Type in EntryScape and how this contributes to a contribution-centric architecture is described in the following sections from an application perspective. See subsection 3.3 for the technical perspective on Entry Type.
4.2. COLLABORATIVE ANNOTATION

Collective construction of knowledge

The Entry Types Local, Link and Reference indicate that only one metadata graph exists and is managed for a resource, whereas an entry of type Link reference manages two metadata graphs that correspond to local and external metadata. The latter type enables applications where management of distributed data is needed, with the primary example being that already existing data is referred to and built upon without duplicating the source information in the local system. The Link reference Entry Type effectively supports the collective construction of knowledge across systems. The only requirement for the involved systems to be interoperable is the use of common standards and protocols as it is the case when the Linked Data path is followed.

Organic.Edunet provides an example of this approach being applied in productive environments. Within the project, Confolio was used to create and manage collections of Open Educational Resources (OER) in the context of organic agriculture and agroecology.

The two most important use cases were:

1. Uploading of or linking to educational resources which were created within the project consortium.
2. Reuse of existing (non-educational) resources in an educational context.

The latter was achieved by harvesting metadata from institutional repositories, e.g. Intute [60] and Bio@gro [61], and contextualizing the described resources by adding additional metadata which made them usable in educational contexts.

Technically, EntryScape uses ReM³’s Location Type Link reference and multiple graphs to keep the metadata from different sources apart. External, harvested metadata is referred to and cached in one graph, and the local, additional metadata (which is used to contextualize) is kept in another. All information is held together by a ReM³ entry, which also keeps track of provenance, access control, and other administrative information.

The same approach, namely using ReM³’s Link Reference to non-invasively contextualize resources, is also followed in other areas such as cultural heritage. EntryScape has gained support for searching cultural heritage hubs such as Europaena [62] and the Swedish K-Samsök [63] which allow users to contextualize and curate resources in their own EntryScape portfolios.
Usefulness and other possible applications

Dedicated metadata management systems, such as aggregators or applications based on curation, usually have a metadata harvesting workflow that causes redundant information by copying the harvested metadata instances into the local system. If the harvested metadata is then modified, a fork of the information is created which is usually not transferred back into the original metadata repository where it was harvested from. ReM³ builds on the principles of Linked Data and therefore focuses on links instead of copies. Information is referred to and built upon, but not forked. ReM³ uses different graphs for metadata from different origins; external (e.g. harvested) metadata is linked to but because of performance reasons also stored, separate from local modifications. A “global view” of the resource’s metadata can be created by simply merging the relevant metadata graphs.

This was also one of the points highlighted in the structured interview results. The interviewed experts did not know of any similar system that is capable of managing heterogeneous (meta)data in a distributed setting, while avoiding data duplication and taking advantage of Linked Data. For more details on this see section 4.3 for a summary of the evaluation carried out and described in paper 5.

4.3 Evaluation

This section summarizes the different angles from which the presented research has been evaluated. Almost all described concepts and applications have been implemented as real and ready-to-use applications, except a client for collaborative tasks which was outlined in paper 2. Most of the applications are used in specific contexts which are presented as showcases. Structured interviews were conducted along with a performance analysis to evaluate the technological underpinnings behind the ReM³ information model and its reference implementation EntryStore.

Implemented applications

The following applications were described and implemented in the context of this thesis. This summary aims to provide an overview of which applications were described in which respective paper:

- Collaborative support for the Conzilla concept browser was implemented along with the collaboration server Collaborilla. This research was described in paper 1.
4.3. EVALUATION

- The collaborative tasks from Paper 2 also rely on the principles of collaborative artifacts from paper 1, but client-support has not been implemented.

- Version 4.0 of SCAM and Confolio (see papers 3 and 4) was a complete rewrite and does not share any code with previous versions of SCAM. Development was started in the context of the Organic.Edunet project, but the software project is still active beyond the end of the funding research projects.

- Within Organic.Edunet the author implemented the mapping of IEEE LOM into the Dublin Core Abstract Model (DCAM [64]) which has been proposed by the joint IEEE/DCMI task force [65]. This implementation is to date the only existing complete implementation of the LOM/DCAM mapping.

- EntryStore also became the reference implementation of the ReM³ information model as published in paper 5.

Showcases

The showcases as listed in section 6 in paper 5 have a strong focus on the management of educational material, and in particular the annotation of resources with learning resource metadata. The showcases are only summarized here, a more detailed account is given in paper 5:

- The Organic.Edunet project [42] had as a goal to facilitate access, usage and exploitation of digital educational content in the fields of Organic Agriculture and Agroecology. The Organic.Edunet “repository tools”, a combination of the SCAM backend and the Confolio frontend, were used to manage and curate a total of 11000 educational resources.

- To show the general applicability of SCAM and its underlying information model ReM³, an attempt was made to harvest the OAI-PMH target of the ARIADNE aggregator [66] for metadata management inside SCAM. A total of 1.2 million metadata graphs were harvested into a SCAM repository and made available for curation. This proof-of-concept showed that even large amounts of resources and metadata can be handled by the backend.

- Within the scope of the first “Hack4Europe!” hack day in Stockholm 2011 [67], Confolio was adapted [68] for searching within and curation of Europeana [62] content. Support for the presentation and editing of the Europeana Data Model [69] was implemented (in addition to the already existing support for the RDF representations of Dublin Core Terms, IEEE LOM, etc).
The common feature of all three showcases is that the end user is given the possibility to collect material from different sources (manual file upload, link to web content, search result from Europeana, ARIADNE, Organic.Edunet) and to adapt (or extend, change, contextualize) the metadata in her own personal space.

Some of the outcomes have been adopted in productive settings:

- SCAM and Confolio as repository tools in Organic.Edunet are used by installations at four different organizations, namely at (1) the Greek Research and Technology Network, (2) the Austrian Federal Ministry of Education, Arts and Culture, (3) the Agricultural University of Athens, and (4) the Royal Institute of Technology in Sweden.

- The VOA3R project [70] supports OER providers with infrastructure to upload, link and annotate learning resources by offering Confolio to do this.

- The TEL-Map project [13] uses Confolio to describe collaborative EC-funded projects and build researcher profiles, in order to create relationships between projects and researchers.

- In the project Hematology Net [71] Confolio is used to build competence profiles and describe learning outcomes in order to get recommendations for educational material matching the users’ learning goals.

A number of additional showcases (and products) have been developed or are under development in connection with commercialization attempts of EntryStore and EntryScape, but they are based on work which goes beyond this thesis.

**Suitability for resource annotation processes**

A combined structured interview and survey focusing on the suitability for resource annotation processes was carried out and described in detail in paper 5. The interview topics were current practices and technologies for resource annotation processes in repositories with heterogeneous metadata. Most of the participating experts had a background in TEL, so the results are primarily valid for learning repositories, but because of the common characteristics of such systems they may in fact be applied to generic metadata repositories. The structured interview and the survey are only briefly summarized here, with an emphasis on metadata enhancement. A full description can be found in paper 5 section 7.2.

The most popular requested features included the capability to maintain a clear separation between metadata of different origins. It should not be necessary to
copy metadata between systems (e.g. when harvesting a repository), causing diverging forks. Most respondents were concerned about redundancy and conflicting changes in the metadata as it happens e.g. when the same record is harvested by several aggregators, modified further down in the “aggregation chain” and never reconciliated with the original. When asked about linking to metadata instead of copying, performance issues and eventually unavailable information came to the respondents’ minds. However, caching techniques to circumvent performance issues were not mentioned. When linking, the original metadata can be extended and built upon, but not modified, which was considered a disadvantage. The importance of interoperability of protocols and formats was stressed by all respondents; abstract models (e.g. DCAM and MLR [72]) are preferred over XML-based formats.

Most of the features and issues brought up by the respondents are covered by the possibilities of a Linked Data-enabled platform. Some of the most relevant points mentioned by experts are the reuse of metadata by referring to the original, the possibility to interlink resources, and the separation of metadata from different sources. These requested capabilities are not only applicable to LD, but also to the ReM³ information model and its derived applications EntryStore and EntryScape. They are covered in particular by the resource and metadata management as described and discussed in papers 3 to 5. Recent research [73, 74] shows that LD is a technology that is increasingly used to interlink educational resources.

Scalability
ReM³ and in particular its reference implementation EntryStore have undergone a scalability analysis based on load tests. This section contains only a summary, see paper 5 section 7.1 for all details. The overall goal was to determine how many concurrent requests that still allow for a responsive experience in interactive applications [75]. The goal was set to reliably receive a response within 50 ms when reading and modifying ReM³ entries (including metadata). The test system was able to sustain an average throughput of 478 reading requests per second with an average response time of 38 ms and 144 writing requests per second with an average response time of 34 ms.

It is difficult to draw conclusions regarding the amount of active users that this corresponds to since this is very use case-specific. However, the test results indicate that such RDF-based systems, especially taking into consideration an additional administrative overhead such as ReM³ entries, are not prone to deliver poor performance. In contrast, it is fully feasible to build interactive Web applications on top of such an architecture.
The limitations of such a system lie rather in the requirement to present a big amount of information (i.e., many resources with metadata) in a meaningful and user-friendly way. The findability and manageability of systems with several thousands of resources and users may have a potentially big impact on the adoption. With EntryScape it is possible to break down amounts of resources and to organize them in folder-like structures as well as in personal- and community-spaces. It also helps to provide a search interface with the possibility to narrow down large result sets. However, how large amounts of resources are finally handled is an implementation detail of the user interface. Neither ReM³ nor EntryStore place a restriction on how the user interface presents entry sets and search functionality; the way it is done in EntryScape serves as an example and is not the only possible way.

**Enabling a Linked Data Platform**

The work of the W3C’s Linked Data Working Group [55] is based on its charter [56] which includes a list of technical issues that guide the discussions of the WG. Some of these technical issues (as listed by 2013-09-16) are solved by ReM³, even though they have not been explicitly addressed by paper 5.

To be able to implement an interoperable and sufficiently generic information management platform, the following issues had to be addressed in ReM³ and its reference implementation EntryStore:

- resource and collection creation (issues 3 and 4)
- resource access with paging (issue 5)
- simultaneous modifications of resources, concurrency (issue 6)
- multiple resource types, rdf:type (issue 7)
- collection management (issues 9 and 10)
- integration of SPARQL (issue 13)

This shows that 8 out of the 16 technical problems from the W3C LDWG charter can be solved using ReM³ and EntryStore. In addition, the latter fulfills a number of the functional and non-functional requirements as stated in the document “Linked Data Platform Use Cases and Requirements” [76], which was initiated to foster the development of the LDP specification [77].

It is possible to resolve even more issues by either slightly modifying the information model or by changing some implementation details. It will be the focus of
future work to advance EntryStore towards a Linked Data Platform that meets all of W3C’s requirements.

How to realize a write-enabled Web of Data is the center of discussion of articles [78, 79] and the W3C’s Read Write Web community group [80]. A write-enabled Web architecture based on Linked Data is still under discussion. Berners-Lee et al revisit the topic in [81], arguing for Linked Open Data as an important component of the Linked Data Web. This paper also describes how the read-write Linked Data Web presumably could be achieved.

The discussions and developments around the Linked Data Platform and a read-write Linked Data Web show that information models such as ReM$^3$ and Linked Data-based platforms such as EntryStore are indeed focusing on relevant technologies and contribute to their advancement.

4.4 Summary

The research presented in this thesis has been evaluated from different perspectives. The list of implemented applications (section 4.3) and the presented showcases (section 4.3), along with the scalability analysis (section 4.3), demonstrate the applicability of the theoretical concepts in real-world applications.

The artifact and entry concepts in ReM$^3$ together form the foundation for an architecture that provides some real benefit for the presented use cases. One of the most significant features is the possibility to extend and contextualize while keeping a reference to the original and avoiding the creation of duplicate information. Especially for metadata annotation processes this is acknowledged by the results of the survey and structured interviews (see 4.3) with experts of the field; the interviews show a demand for an information architecture such as the one described in chapter 3.
Chapter 5

Conclusions

This thesis has explored ways of collectively constructing knowledge with a strong tie to modern Web architecture and the OWA. As mentioned in the introduction, the main context of this thesis is heavily grounded within Web architecture built on Semantic Web technologies and in particular Linked Data, which also led to the focus on Web-based collective knowledge construction, both in the research questions and their answers.

The projects and work on which the described research is based have led to a contribution-centric architecture which has resulted in an information model and a reference implementation. This chapter discusses this thesis’ research questions and ties together the results of the published research.

5.1 Research questions revisited

How can Web-based collective knowledge construction be supported?

To answer this question it was necessary to discuss prime examples of related techniques for collaborative Web applications. This was done in chapter 2 with a focus on contributions. This thesis describes activities which may be unplanned and ad-hoc, and where “contribution-centric” seems to be a sufficiently neutral term for loose forms of collaboration and collective knowledge construction. The main focus lies on use cases where contributors may have a distant relationship (if one at all) with the original author, or no relationship at all.

The applications that have been summarized in section 2.3 support Web-based collective knowledge construction:
• In public wikis such as Wikipedia everybody is allowed to contribute, but the collaborative result has to be consolidated.

• Blogs using linkback mechanisms can be used to link blog posts which is suitable for building knowledge networks, but this is restricted to blog posts.

• Forums with their discussion threads allow to let knowledge evolve, but this has to happen inside the forum and in a chronological order.

• Annotea is the only technique that builds on Semantic Web technologies. It is restricted to annotations on content inside Web pages.

The applications above (except Annotea) can be counted to the most popular categories of Web applications. They support Web-based collective knowledge construction but are limited either in their generic applicability or with respect to the OWA, which is discussed in the answer to the next research question below.

In addition to the application categories above, the characteristics and requirements in the sections 3.3 and 3.4 have been identified as supporting for Web-based collective knowledge construction. They are also fundamental for answering the next research question, focusing on the OWA.

What are the characteristics of collective knowledge construction with respect to the Open World Assumption?

Chapter 3 introduces the proposed contribution-centric architecture which builds on artifacts as introduced in paper 1 and an information model for managing resources and their metadata as described in paper 5. The chapter concludes with a comparison in figure 3.1, that summarizes the most important characteristics of this contribution-centric architecture and compares the presented information model ReM$^3$ with the Web applications of collaborative character such as wikis, blogs, etc from section 2.3.

The following list enumerates the characteristics of collective knowledge construction with respect to the OWA, see also the comparison in section 3.7 and the preceding descriptions in the sections 3.2, 3.3 and 3.4:

• Data and metadata are separate and treated in a unified way.

• Contributions can be persistently and globally identified.

• Contributions can be added to uniquely identifiable concepts.

• Contributions are separate from the original document.
5.1. **RESEARCH QUESTIONS REVISITED**

- Traceable history (provenance) of an artifact and its contributions.
- May result in different views consisting of separate contributions.

Chapter 2 analyzes the abilities of some relevant technologies and concludes that there is a lack of models that match the characteristics of a contribution-centric architecture. Annotea is the technology that comes closest to the idea of collaboratively constructed artifacts from paper 1, but it focused on annotations only. Artifacts as described in papers 1 and 2 are restricted to a certain way of adding to existing information and do not directly allow to build generic applications. It is possible to implement certain collaborative properties in an application, but it is up to the application whether this is done by managing sets of contributions as it is done e.g. with conceptual models in Conzilla.

ReM³ as specified in paper 5 (and with a preliminary design in papers 3 and 4) is an information model that is based on interlinking information. Entries are used to manage resources together with their metadata, and it is ultimately up to the application how strictly controlled the working environment should be. An important characteristic of ReM³ is that it facilitates knowledge construction across system borders since it solely relies on HTTP-based URIs, no matter whether they are local or external.

In chapter 3 an architecture was described that is constructed on two cornerstones; it builds on artifacts from papers 1 and 2, and it takes advantage of the ReM³ information model described in papers 3 to 5. The resulting contribution-centric architecture provides the foundation for a Web-based knowledge construction that supports an OWA.

**How can Semantic Web technologies support a contribution-centric architecture?**

The proposed contribution-centric architecture is based on the concept of artifacts as well as the ReM³ information model and its reference implementation EntryStore. ReM³ represents an improvement of the original artifact idea. Collaborilla (implementing the artifacts) allows to track contributions around content, while EntryStore allows to interlink information in heterogeneous environments.

In order to give an account of how Web technologies can used to support this architecture it makes sense to summarize the underlying characteristics of ReM³ as described in section 3.3:

- All entities in ReM³ are identified by dereferencable URIs, supporting contributions in different contexts.
• ReM$^3$ expresses its information in RDF which is structured using named graphs.

• Linking in ReM$^3$ is done according to the Linked Data principles.

• The use of RDF as a common carrier allows the integration of heterogeneous information by taking a metadata harmonization approach [82].

As shown in chapter 4 (and papers 3-5) Web technologies, and in particular the ones that have been standardized with the Semantic Web in mind, have been successfully used to realize a contribution-centric architecture. The Semantic Web stack is now considered to be a mature technology stack [83], which is also reinforced by the latest developments at the W3C where the new Data Activity [84] succeeds the Semantic Web activity [85] (among others).

Web architecture serves as a flexible foundation to build applications on. It does not come with a ready-to-use application layer, so it would be wrong to talk directly about collaboration or collective knowledge construction using Web architecture. An intermediate layer between is needed, which actually takes advantage of some of the technical characteristics of the Web. ReM$^3$ is such an intermediate layer, i.e. a model for Web-based knowledge construction that supports the Open World Assumption. ReM$^3$ supports an OWA-oriented approach, but can be used to implement collaborative applications reflecting a closed world; it is ultimately up to the application built on top of ReM$^3$ how open or closed the system will be. The projects using ReM$^3$ (through SCAM/EntryStore) all have some collaborative component.

The micro and macro categories as mentioned in chapter 1 can be used to explain the larger picture. Taking Organic.Edunet as an example of a micro system, it automatically becomes part of a bigger macro system as users start to reuse and contextualize content. Some of the content used within Organic.Edunet already originates from other micro systems (e.g. Intute and Bio@gro) which probably did not foresee that their content would be used “in the wild” on such a large scale in the future. The content is translated, contextualized for other scenarios, and described using specialized metadata profiles. However, it is difficult to draw the line between micro and macro system. Linked Data enables such transitions without duplicating data or losing information about the origin.
5.2 Building applications on a contribution-centric architecture

Research within the field of Semantic Web has not been focused on discussing collaborative applications that really build on and exploit the possibilities of SW-based architectures. The contribution-centric architecture presented in this thesis may act as an enabler for developing Web applications based on SW-technologies. The applications described in chapter 4 show how collective construction of knowledge can be done in practice and how collaborative features can be implemented with the help of an architecture that is largely collaboration-agnostic. A generic way of managing information – as provided by ReM³ – makes it possible to implement solutions that avoid information models that are too specialized for the application at hand. The evaluation summarized in section 4.3 details several aspects which endorse the validity of the presented research: a list of implemented applications and showcases, results from structured interviews that have investigated the suitability for resource annotation processes, and scalability aspects.

Nevertheless, it is ultimately up to the application how “loose” the collaboration should be and to which extent the OWA is incorporated. The SW-stack (and implicitly the OWA) acts on a conceptually higher level and does not provide any facilities that are known from, or that could be used to implement, collaborative systems (see section 2.2). The presented architecture provides a toolkit to support the development of loosely collaborative applications. The applications to collaboratively build conceptual models (section 4.1) and to collaboratively annotate educational resources (section 4.2) are good use cases to illustrate the potential of the semantic technologies stack and the contribution-centric architecture that builds on top if it.

5.3 Future work

Some possible improvements and ideas have not been explored so far. Context-maps evolve, concepts may be moved (or removed) and contributions to it may point into the void. How can be made sure that contributions remain consistent with the “snapshot” of the artifact when the contribution was made. Perhaps lessons can be learned from techniques that specialize on quick iterations such as the distributed version control system Git [86].

Another area that remains to be explored is how the mechanism of the inverse relational cache (as described in section 3.3) can be employed across system borders in distributed settings.
Future work based on this research will most likely focus even more on the question “What can the Semantic Web and Linked Data do for me?”. These technologies have great potential and it is necessary to build, test and mature more applications that take advantage of the full capabilities of Linked Data – not only storing data in RDF. ReM³ and EntryStore represent a step in this direction, making it possible to build applications on top of the SW-stack without having to deal with the full complexity of it. A spin-off and ongoing commercialization efforts [87] show that there is real business value in driving the transformation from research prototypes to mature products.

The emerging Linked Data Platform [55] and the development towards a read-write Linked Data Web show that ReM³ and EntryStore are well positioned in this stack of highly relevant Semantic Web technologies.
Bibliography


http://www.w3.org/2001/tag/.


Multi-synchronous collaborative semantic wikis. In Web Information Systems


URL http://www.w3.org/TR/rdfa-primer/.

In Invited talk at Workshop on User Aspects of the Semantic Web

http://www.w3.org/2001/Annotea/Projects.html.

[31] Paul Grosso, Eve Maler, Jonathan Marsh, and Norman Walsh. XPointer
http://www.w3.org/TR/xptr-framework/.

[32] Ambjörn Naeve. Conceptual navigation and multiple scale narration in a
TRITA-NA-D9910, Department of Numerical Analysis and Computing
Science, KTH, Stockholm, 1999. URL

Master’s thesis, Department of Numerical Analysis and Computing Science,

[34] Ambjörn Naeve. The Concept Browser – a new form of knowledge
management tool. Proceedings of the 2nd European Web-based Learning


[56] Linked Data Platform (LDP) working group charter, 2013. URL http://www.w3.org/2012/ldp/charter.


[73] Stefan Dietze, Salvador Sanchez-Alonso, Hannes Elner, Hong Qing Yu, Daniela Giordano, Ivana Marenzi, and Bernardo Pereira Nunes. Interlinking educational resources and the web of data: A survey of challenges and


[76] Linked Data Platform use cases and requirements, 2013. URL http://www.w3.org/TR/ldp-ucr/.

[77] Linked Data Platform 1.0, 2013. URL http://www.w3.org/TR/ldp/.

[78] Realizing a write-enabled Web of Data. URL http://www.w3.org/wiki/WriteWebOfData.


[80] W3C Read Write Web Community Group. URL http://www.w3.org/community/rww/.


BIBLIOGRAPHY


Paper summaries

Paper 1: Collaborative Construction of Artifacts

Authors: Hannes Ebner, Matthias Palmér, and Ambjörn Naeve
Published: Proceedings of the 4th Conference on Professional Knowledge Management, Workshop Collaborative Knowledge Management (CoKM2007)
Year: 2007

Summary
The paper describes an approach for collaborative construction of artifacts and an implementation to facilitate collaborative conceptual models. The underlying research questions focus on how to discover, contribute to and publish artifacts. The described approach assumes an “open world” in which contributions are to be made without specific requirements on specifically coordinated collaboration processes among the involved authors and contributors. The approach is applicable to any artifacts that may be separated into distinct contributions and gives each viewer the control of including or excluding contributions.

Contribution
The main author wrote the paper with minor contributions and comments from the co-authors.

Paper 2: Collaborative Tasks using Metadata Conglomerates – The Social Part of the Semantic Desktop

Authors: Olaf Grebner and Hannes Ebner
Published: Proceedings of International Conference on Semantic Systems (I-
Summary
The paper is based on the approach of collaborative construction of artifacts as described in paper 1. It presents an application that enables ad-hoc task management by explicitly integrating task metadata into the collaboration process. The described application is based on the Kasimir task management client and the Collaborilla collaboration server.

Contribution
The paper was written in close collaboration between the authors who spent an equal amount of work on the publication. The author of this thesis was mainly responsible for the parts on collaboration and artifacts.


Authors: Hannes Ebner and Matthias Palmér
Published: Proceedings of the workshop Mash-Up Personal Learning Environments (MUPPLE), in conjunction with the 3rd European Conference on Technology-Enhanced Learning (ECTEL)
Year: 2008

Summary
The paper proposes a generic framework for mashup-friendly resource and metadata management, presents a use case and presents an implementation that builds on the framework. The paper introduces the Entry as a way of managing resources and their metadata together. The framework is described on a conceptual level along with an implementation that exposes an HTTP API. The presented application building on top of the framework is a Web application that allows the mashup of information from different sources.

Contribution
The main author wrote the article in close collaboration with the co-author and a special focus on the architecture of the backend.
Paper 4: Learning Object Annotation for Agricultural Learning Repositories

Authors: Hannes Ebner, Nikos Manouselis, Matthías Palmér, Fredrik Enoksson, Nikos Palavitsinis, Kostas Kastrantas, and Ambjörn Naeve
Published: Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT)
Year: 2009

Summary
The paper describes a Web-based tool that has been developed to annotate learning objects (resources) with educational metadata. The application allows the organization of resources in a portfolio-like structure of learning repositories. It also provides the foundation for the Organic.Edunet federation of learning repositories. The approach is based on the resource and metadata management framework as described in paper 3.

Contribution
The main author wrote the paper with minor contributions and comments from the co-authors.

Paper 5: An information model for managing resources and their metadata

Authors: Hannes Ebner and Matthías Palmér
Published: Semantic Web Journal, Volume 5, Number 3/2014
Year: 2014

Summary
The paper presents the Resource and Metadata Management Model (ReM3) which is a refinement and formalization of the framework described in paper 3. The information model builds upon Linked Data principles and allows the expression of relations between resources and metadata, and the management of provenance as well as access control. The model’s reference implementation is presented along with several show cases. The results of an evaluation from several perspectives is
described, e.g., the suitability for resource annotation, a scalability analysis, and its level of adoption.

Contribution

The main author wrote the article with minor contributions and comments from the co-author.

Other publications

The following papers have been published in the context of the author’s research projects, but are not included in this thesis.


