2019

Decentralised Hosting and Preservation of Digital Collections

Goebert, Samuel

http://hdl.handle.net/10026.1/13650

http://dx.doi.org/10.24382/494
University of Plymouth

Copyright Statement

UNIVERSITY OF
PLYMOUTH

Decentralised Hosting and Preservation of Digital Collections

by

Samuel Goebert

A thesis submitted to the University of Plymouth

in partial fulfilment for the degree of

DOCTOR OF PHILOSOPHY

School of Computing, Electronics and Mathematics

July 2018

Acknowledgement

This thesis has been, without a doubt, the single largest test of my own commitment and patience. It would not have been a success without the support of the following people:

First and foremost, I wish to thank my Director of Studies, Prof Dr Bettina Harriehausen-Mühlbauer for encouraging me to start the PhD program. Her faith, endless patience, motivation, and immense knowledge helped me during all stages of research and the writing of this thesis. Her encouragement allowed me to grow as a researcher and as a person. She has been a tremendous mentor for me and I could not have imagined having a better Director of Studies for my PhD study.

Besides my Director of Studies, I would like to thank the other members of my thesis committee: Prof Dr Christoph Wentzel and Prof Dr Steven Furnell, for their encouragement, guidance and insightful comments, but also for the hard question which inspired me to widen my research from various perspectives.

In addition, I would like to express my gratitude to the staff of Plymouth University, especially Carole Watson, for the superb guidance throughout this process, timely feedback, and last-minute favours.

I thank my fellow doctoral students for their feedback, cooperation, stimulating discussions, for the sleepless nights when we were working together to meet deadlines, and for all the fun we have had in the last few years.

Finally, I must express my very profound gratitude to my wife Mandy, and my daughter Mira for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching, writing, and finishing this thesis.

This accomplishment would not have been possible without all of you. Thank you.

Authors Declaration

At no time during the registration for the degree of Doctor of Philosophy has the author been registered for any other University award without prior agreement of the Doctoral College Quality Sub-Committee

Work submitted for this research degree at the University of Plymouth has not formed part of any other degree either at the University of Plymouth or at another establishment.

The following external institutions were visited for consultation purposes:

Hessisches Hauptstaatsarchiv, Wiesbaden, Germany

German National Library, Frankfurt am Main, Germany

Publications (or public presentation of creative research outputs):

Goebert, Samuel, et al. 'A non-proprietary RAID replacement for long term preservation systems' Conferences, iPRES 2011 - Proceedings of the 8th International Conference on Preservation of Digital Objects

Goebert, Samuel, Bettina Harriehausen-Mühlbauer, and Steven Furnell "Decentralized Hosting And Preservation Of Open Data." Archiving Conference. Vol. 2013. No. 1. Society for Imaging Science and Technology, 2013.

Goebert, Samuel, Bettina Harriehausen-Mühlbauer, and Steven Furnell. "Towards A Unified OAI-PMH Registry." Archiving Conference. Vol. 2014. No. 1. Society for Imaging Science and Technology, 2014.

Presentations at conferences:

Archiving Conference. Vol. 2013. No. 1. Society for Imaging Science and Technology, 2013, Washington D.C., United States

INC 2014, July 19-21, 2014, Darmstadt, Germany

Archiving Conference. Vol. 2014. No. 1. Society for Imaging Science and Technology, 2014, Berlin, Germany.

Seminar, "Collective Discovery and Storage Of Digital Cultural Heritage", 2014, Mikkeli, Finland

Word count of main body of thesis: 28857

Signed..........................

Date............................

Decentralised Hosting and Preservation of Digital Collections

Samuel Goebert

The Internet is changing from a web of documents to a web of data. Open-data collections like Wikipedia, Internet Archive, Stack Exchange and OpenStreetMap have become important sources of global knowledge. The data are freely available and everybody is invited to contribute.

Preserving digital collections is also performed by entities not affiliated with the original initiatives and involves copying content and meta-data about a collection to new storage locations. A copied collection retrieved from an untrusted location requires the task of revalidating the authenticity of the data. Since budgets are limited, novel ways of finding storage space have to be acquired. Safely storing and validating data without the need to own and control the storage location enables this.

This thesis develops a protocol for decentralised hosting of digital collections. The result is a formalised, decentral mode of discovery, curation and hosting for datasets, retaining authenticity even at untrusted storage locations. Donating storage space and bandwidth becomes possible for entities not affiliated with the original initiative and ensures long-term access to the authentic collection for the public at the same time. The protocol is leveraging the bittorrent protocol, a variation of the block chain protocol and is backwards compatible with existing web application architecture.

This novel approach is validated through a proof-of-concept prototype. A series of test scenarios is used to illustrate how a decentralised collection would behave given multiple participants. The results support the use of a decentralised hosting approach for digital collections leveraging storage locations that are under the supervision of entities not affiliated with the original initiative.

The thesis concludes with a detailed summary of the contributions to the field and suggests further areas of study in the context of distributed-preservation.

Contents

List of Figures

List of Abbreviations

AIC	-	Archival Information Collection
AIP	-	Archival Information Package
AIU	-	Archival Information Unit
API	-	Application Programming Interface
ANP	-	Archive Network Protocol
CD-ROM	-	Compact Disk - Read Only Memory
CPU	-	Central Processing Unit
DiMag	-	Digital Magazine
DNS	-	Domain Name System
DVD	-	Digital Video Disk
FTP	-	File Transfer Protocol
GB	-	Great Britian
GCP	-	Good Clinical Practices
HGB	-	Handelsgesetzbuch
HTML	-	HyperText Markup Language
HTTP	-	HyperText Transfer Protocol
ISBN	-	International Standard Book Number
ICQ	-	Instant-Messaging-Service
IP	-	Internet Protocol

ISO	-	International Organization for Standardization
JSON	-	JavaScript Object Notation
LIDAR	-	Light Detection and Ranging
LOIRP	-	Lunar Orbiter Image Recovery Project
MOT	-	Ministry of Transport
NAS	-	Network Attached Storage
NASA	-	National Aeronautics and Space Administration
NPM	-	Package manager for JavaScript
OAIS	-	Open Archival Information System
OAI	-	Open Archives Initiative
OAI-PMH	-	Open Archives Initiative Protocol for Metadata Harvesting
OCR	-	Optical Character Recognition
OS	-	Operating System
OSI	-	Open Systems Interconnection Model
OWASP	-	Open Web Application Security Project
P2P	-	Peer-to-Peer
PDF	-	Portable Document Format
PDI	-	Preservation Description Information
PGP	-	Pretty Good Privacy
RAID	-	Redundant Array of Independent Disks
REST	-	Representational State Transfer

RSS	-	Really Simple Syndication
SAN	-	Storage Area Network
SIP	-	Submission Information Package
TOS	-	Terms of Service
UUID	-	Universally Unique Identifier
UK	-	United Kingdom
UNESCO	-	United Nations Educational, Scientific and Cultural Organization
URL	-	Unified Resource Locator
WebDAV	-	Web-based Distributed Authoring and Versioning
WWW	-	World-Wide Web
XAM	-	extensible Access Method
XML	-	Extensible Markup Language

1 Introduction and Overview

This chapter introduces the context of this research, namely open-data and digital-preservation. It provides a summary of the main issues surrounding digital-preservation of open-data. It argues why extending current preservation approaches with distributed-system architecture will benefit the collaboration and archiving of openly shared data collections. Additionally, the aims and objectives of this research are presented. The chapter concludes with a summary of the thesis structure.

1.1 Introduction

> “Data is the bedrock on which the scientific edifice is built. More efficient data-sharing and more open access to information and resources will make it easier for observations to be confirmed, experiments to be replicated, hypotheses to be supported, rejected or refined, and, ultimately, for answers to societal challenges to be provided.“ (The European Federation of National Academies of Sciences and Humanities, 2012)

The Internet is changing from a web of documents to a web of data (Bizer et al., 2008). Open-data initiatives like Wikipedia, Internet Archive, Stack Exchange, and OpenStreetMap have become important sources of global knowledge. The data is freely accessible and everybody is invited to contribute.

Preserving these digital collections beyond the lifetime of the original initiative has become a worldwide effort with cultural and social significance. This effort is performed by entities not affiliated with founding initiatives and involves copying the original collections to new storage locations.

The process of copying the data must be repeated regularly to incorporate the latest changes from the original collections. Editing of a copied collection is possible however it creates a new and independent set of data. Interfaces that support contributions have to be provided by the original collection otherwise change-sets that happen outside the collection are not synchronized back. Changes that are not in an official release by the original initiative are generally regarded as untrustworthy. The trust in a genuine, fact-based dataset is tied to download locations, digital verification methods like hash-values or initiatives that hosts the data. An example of the conflict of interest with data is the mapping data of the Crimea region (Bandeira, 2019). Depending on the download location of the dataset, the region might belong to the Ukraine or Russia (Panek and Brychtova, 2015).

With many collections that store different versions and content than the original dataset, it becomes difficult to determine the leading dataset that should be preserved and contributed to. Smaller collections that have a similar goal fail to gather contributors because they lack visibility, as contributors seek the largest exposure for their contributions. After Wikimedia announced Wikivoyage, a wiki dedicated to travel, many of the authors left existing sites with a similar focus like Wikitravel and World66 for it. Due to the open license of the content at Wikitravel, a copy of the data set served as a starting point for the Wikimedia site. Today Wikivoyage is the dominant data set based on the numbers of the recent edits (Cohen, 2012).

1.1.1 Long-Term Preservation

> "Long-term is long enough to be concerned with the impacts of changing technologies, including support for new media and data formats, or with a changing user community. Long-term may extend indefinitely." (International Standards Organisation, 2012)

Professional archiving of physical objects is a very time-consuming method used to preserve the cultural goods of a society (Mauthner, Parry and Backett-Milburn, 1998). Physical objects have been archived for several hundred years using special environments and controlled conditions to prevent degradation (Culham, 1989). A craft that has been passed on for generations is facing the incorporation of a new type of artefact. Many objects that form our cultural heritage today are solely produced in digital format with no real-world equivalent. Erway (2010) describes the digital camera as the fastest growing form of born-digital content. The new responsibility for memory-organisations is to preserve these digital-born objects as well.

> "Digital-preservation (whether of e-journals, eBooks, or anything else) is the series of management policies and activities necessary to ensure the enduring usability, authenticity, discoverability, and accessibility of content over the long-term." (Kirchhoff, 2011)

Traditional memory-organisations such as archives, museums, and churches are challenged by the explosion of digital-born data. The size and numbers of datasets, the technological knowledge required, the domain expertise needed for the selection process, and limited budgets are hindering preservation, access, and interpretation of the data for future generations. Many organisations have never thought of themselves as performing an archival function at all.

The domain of digital preservation is not limited to preserving access to the data on a playback medium; the file in binary format itself also possesses its own challenges. Identifying and interpreting a file in binary format presumes the existence of metadata and knowledge about the data to interpret. Once a format has been successfully identified, tools have to exist that are able to render a representation of the interpreted data with the available hardware.

1.1.2 Open-Data

Open-data is the idea that certain data should be freely accessible to everyone to use and republish as they wish, without restrictions from copyright, patents or other mechanisms of control (Auer et al., 2007). Data published in the public domain or under a permissive license such as the GNU General Public License can become the source of innovation and knowledge gains. As the data is reused in ways the author did not intend at the time of creation, it allows for experimentation and discovery in areas beyond the initial scope and purpose.

Open-data is produced by many different entities ranging from data produced by a single entity like governmental agencies or public and private corporations, to loosely coupled communities of volunteers. A community forms around a single problem domain and attracts crowds that are largely self-selected professionals and experts who opt-in to crowdsourcing arrangements as opportunities for creative expression, entertainment or to pass the time when bored (Brabham, 2012).

Open-data initiatives have become important sources of global knowledge (Giles, 2005) even though no commercial offer is the main driving force behind the project. While mainly built by volunteers, the quality of the data has reached and, in some cases, exceeded proprietary offerings (Soden and Palen, 2014).

As relying on open-data becomes more widespread, preserving this information in a trusted way becomes an equally important topic for ensuring continuation. For example, the Humanitarian OpenStreetMap Team (HOT) applies the principles of open source and open-data sharing to humanitarian response and economic development (Crowley, 2013). After the Earthquakes in Haitian 2010 (Norheim-Hagtun and Meier, 2010) or Indonesia 2018 (Akrimullah, Sima and Akhadi, 2018) the projects provided tools and people to create maps of the areas which have been non-existent before.

1.2 Problem Statement

The preservation challenge is to provide enough computing capacity, bandwidth, technological knowledge, electricity, and budget to keep datasets accessible to interested parties. This is not a challenge for a single publicly or privately-funded entity, a certain domain like a memory organisation, or a country. This challenge has become a global issue only solvable by building partnerships and networks of interested organisations, not only within certain domains, such as a library or archival communities but also across domains (Hofman, 2002).

Applying the definition of populations from the mathematical field of statistics to describe a data set, digital collections can be divided into two categories: :

1. Finite Collections: Finite, self-contained and requires no further contributions to complete the collection. Examples of these types of datasets are: public sector information like statistical data about a time period that is in the past (Böhm et al., 2010).

2. Infinite Collections: Continuous datasets that require constant contributions are endless and require enormous resources to build, for example, the "The Macaulay Library at the Cornell Lab of Ornithology" dataset collecting audio and video

material of birds.

Open-data projects are often build by a single person or core group setting the expected quality standards and contribution rules for the dataset (Leskovec, Huttenlocher and Kleinberg, 2010). Contributions from other entities to infinite datasets are reviewed and measured by the core group according to self-assigned standards to keep control over quality and content of the dataset. As the core group is accountable with their reputation for the content of the dataset, open-data collections use a single storage location, which is under the control of the core group, to facilitate all changes to a dataset (Ung and Dalle, 2010).

While the structure of a contribution can be validated for the required syntax that has been specified for a dataset, there are only limited ways of validating the logical aspect of the data. The trust that only validated facts are presented in the dataset, comes with the trust in the core group to establish rules and mechanisms, to prevent intentional vandalism of a dataset. Even subtle changes to a dataset have an enormous impact on the interpretation of the data (Ballatore , 2014).

It is possible for any interesting entity to download open-data and republish it under a domain name that is not affiliated with the original source. However, as the trust in authenticity in the dataset is bound to the location of the download, republishing the information requires the entity to have an equal trust level established as the original source. Failing to demonstrate the level of trust will result in the source simply being regarded as untrustworthy and the data cannot be reused without caution (Javanmardi et al., 2010).

Depending on the size of the dataset, the validation process can require huge resources. While individual releases of a dataset are often accompanied by the digest of a cryptographic hash function that allows for detection of unintentional changes during transport, this method of validation does not apply to the individual data-items. It is possible to detect a change but it remains unclear which exact item was changed; the whole dataset still requires a logical revalidation.

Open-data is often provided as a download of the complete dataset or via an Application Programming Interface that provides access in a more programmatic way. The access itself happens through a website or service that is hosted by the initiative. For example the OpenStreetMap Archive is accessible via the link `https://planet.openstreetmap.org/` as single file or via the API hosted at `https://api.openstreetmap.org/` to retrieve individual items.

As the method of access is different for every project, it is time and resource-consuming to build an automated synchronisation method for every dataset. Manual labour has to occur for every dataset that an archivist wishes to download. Starting from different discovery processes to different validation models forces these archivists to acquire significant resources in the domain of programming. The prevention of common platforms is shifting valuable resources from preserving a dataset to the repetitive task of solving common problems for every collection.

As many organisations have never thought of themselves as performing an archival function, the function of long-term preservation of the dataset and its obligations become an afterthought.

1.3 Aims and Objectives of the Research

This study is concerned with access and hosting of digital collections. More specifically, it identifies the problems of projects that use a central storage location and focuses on the architectural weaknesses that have brought about these problems. The aim is to solve the problem with the underlying philosophy of open-data principles in mind.

The research question studied in the present work is: **How can digital collections be hosted and preserved decentral and still retain its authenticity?**

To answer the question, the research programme can be divided into five key phases:

Phase 1 provides a comprehensive analysis of long-term preservation, storage system architecture, and open-data in its current state.

Phase 2 focuses on three areas of weakness that provide the greatest cause for concern: accessibility, collaboration and authenticity of a digital collection. Based on this understanding, a comprehensive argument is made for the need for a protocol that allows to decentralise the hosting of digital collections. A set of requirements is established, which makes the use of the protocol suitable for long-term preservation.

Having established what preserving entities need and what makes current solutions inappropriate has allowed the research to proceed to the next stage. Phase 3 was to develop a model for decentralised hosting and preservation of digital collections, which represents the focus of the research.

The model comprises a new mechanism for formalised discovery of a storage locations for a decentralised collection; a new validation scheme that allows all participants to validate the data integrity of a collections and a synchronisation method orchestrating contributions to a decentralised collection. The model is an extension to augment existing web applications, which enables these new decentralised collections to be integrated into existing infrastructure.

The objective of Phase 4 was to develop the model such that it could be deployed on the web to provide a real solution to the identified concerns. As such, this phase involved completely specifying the design of the protocol from an engineering perspective and developing, testing, and measuring a prototype.

Finally, the objective of the last phase was the evaluation of the prototype. Several possible areas of improvement are shown by discussing limitations, and describing new areas of research that can be performed to enhance and refine the work.

These objectives can be more formally specified as follows:

1. to assess the current state of approaches to preservation and architecture of storage systems architecture.

2. to identify the weaknesses of open-data project architectures specific to collaboration, accessibility, and authenticity.

3. to develop a model for managing collaboration, accessibility and authenticity in a decentralised environment, and ultimately provide a solution to the identified

concerns.

4. to design and specify a system that implements the model and build and test a prototype to demonstrate its effectiveness.

5. to evaluate the prototype, discuss limitations, and describe further areas of research

These objectives correspond to the general sequence of the material presented in the subsequent chapters of the thesis, as they will be discussed in the next section.

1.4 Research Methods

The underlying research in this thesis is based on design science introduced by March and Smith (1995). It emphasizes the connection between knowledge and practice by showing that we can produce scientific knowledge by designing useful things (Wieringa, 2009). However, one has to carefully distinguish between solving practical problems and knowledge problems. The former changes the research subject to better suit specific stakeholder's needs, while the latter observe the research subject to better understand it (Wieringa, 2009).

IT research can be seen as the study of artefacts being adapted to their environment. It reflects both, practical problems in terms of design science and knowledge or theoretical problems in terms of natural science to explain how and why a designed solution works within its environment (March and Smith, 1995). As IT artefacts always are produced and altered based on particular stakeholders needs, there cannot be fundamental and generalizable theories as in natural sciences, let alone persistent ones. As soon as the stakeholders needs change, the theory becomes invalid. (March and Smith, 1995) therefore suggest a research framework consisting of two dimensions: The first is based on the design science outputs constructs, models, methods, and instantiations. The second is based on generic design science and natural science research activities and includes building, evaluating, theorizing, and justifying one or more of the previous artefacts. They furthermore postulate that these steps should form an iterative cycle until a satisfactory answer to the practical problem is found and scientifically justified.

The IT Design Science Research Cycle according to March and Smith (1995) starts with the evaluation of the problem (Literature Review), theorizing a solution (Build Model), justifying this solution (Paper), and finally building the solution (Prototype). To select the correct methodology for these research activities, Wieringa (2009) suggests to decompose the research question into practical problems and knowledge questions to avoid applying unsound research designs.

Applied to this thesis, the aim of the underlying research presented in section 1.3 can be decomposed into several nested research cycles.

- Authenticity: Evaluate, Theorize, Justify, Build
- Collaboration: Evaluate, Theorize, Justify, Build
- Accessibility: Evaluate, Theorize, Justify, Build

After identifying the overall problem it is further evaluated in chapters 2 for the three basic components long-term preservation, storage system and open-data. As this evaluation represents solving a knowledge problem, literature research and logical derivation of results was chosen as research methodology. The research furthermore depends on the author's experience in the field.

As the evaluation of the problem resulted in three sub problems, a nested design science research cycle was applied to each of them in which the first stage was already completed by the previous evaluation. For Authenticity, Collaboration, and Accessibility, chapter 4 solves a practical problem by modelling a solution for particular stakeholders needs. The subsequent knowledge question justifying the design was answered with literature research and a logical analysis of the design based on the author's own expertise from working in the field, as well as on expertise from subject matter experts through peer review.

After modelling a solution it is justified by ensuring that the proposed solution is valid in the environment it is intended for. In this research this was done in two steps. The first step is described in chapter 5 and consists of an exemplary implementation of the model. The implementation is based on the author's experience in the field and personal discussions with subject matter experts. The second step consist of peer review through a published and presented paper to the long-term preservation community.

Solving the final practical problem of the IT Design Science Research Cycle, building the overall solution, this is left to the long-term preservation community. As this step involves integration into proprietary systems, it is seen as an individual research project. Although this limitation may seem as an incomplete research cycle it should be noted that a complete cycle was executed for the prototype implementation and that the overall design itself has been validated through peer review. The latter is also in line with the overall research aim of proposing means to how digital collections can be hosted and preserved decentral and still retain its authenticity?.

1.5 Thesis Structure

This thesis describes the research leading to the formulation of a protocol for decentralised hosting and preservation of digital collections. The foundations for the thesis are provided in chapter 2, which begins by examining the current state of long-term preservation, storage-system architecture, and digital collections in detail order to gain an understanding of the most important aspects relevant to this research.

Chapter 3 provides substantive evidence of the weaknesses in current open-data preservation approaches that are a direct result of its centralised access philosophy and which are eroding its effectiveness of providing long-term access to a digital collection. The chapter reveals the dichotomy that centralised preservation introduces to an open-data community with a centralised philosophy leading to a trusted project but ineffective preservation system and a distributed design philosophy leading to an untrusted project but effective preservation system. This forms the core focus for the research, which aims to significantly improve the preservation of digital collections without destroying the openness and the popularity of a particular collection.

An exhaustive literature search is presented that reveals the extent of the preservation problems and the existing solutions that have been proposed. The chapter effectively underpins the research by identifying the scope of the centralised architecture problem and concludes that a new, decentralised solution is required to solve this.

Chapter 4 details the approach taken to derive the new protocol taking existing solutions into consideration. It provides the model that is used as a blueprint to show appropriate points for decisions that alter the behaviour of the solution. For each major decision point, a rationalization is given for the choice taken for this thesis, taking into account the requirements learned through the dialog with experts in the field.

Chapter 5 presents such a decentralised solution. The Archive Network protocol is a new way of hosting digital collections that is designed to work according to the decentralised nature of interested communities while retaining the authenticity of the collection.

The chapter provides a conceptual overview and detailed specification of the protocol, which includes the design of an extension for a typical three-tier web application architecture, the introduction of a global timeline as a fundamental entity for synchronisation, and a discovery system for long-term access to a collection built on existing infrastructure to help increase the accessibility of a collection without falling foul of the centralised architecture dichotomy defined in chapter 3.

The chapter essentially provides a blueprint for deploying the protocol, including a functional design specification of its core concepts. The chapter describes the definition of a new infrastructure and the interfaces that can be deployed in a way that is compatible with existing infrastructure.

Chapter 6 describe a prototype of the Archive-Network protocol, which has been developed according to the specification to validate its design. The performance of the prototype has been measured and extensive results are provided in full to demonstrate the effectiveness of decentralised architectures for preservation purposes.

Finally, chapter 7 presents the main conclusions arising from the research, highlighting the key achievements and limitations. The chapter contains a discussion on areas for future research and development.

The thesis also provides a number of appendices in support of the main discussion. These appendices also include several published papers arising from the research program.

2 Literature Review

This chapter introduces the context of this research. The author provides an introduction to the main topics long-term preservation, storage-systems architecture, and open-data.

2.1 Long-term Preservation

This section gives an introduction to the topic of long-term preservation. It moves from physical objects archived by organisations with different ambitions, to explaining what kind of objects are preserved and why. Digital objects are the new frontier for archives; different models of a true representation of a born digital item exist. This will lead us to different interpretations about the consistency and nature of a digital object. The chapter closes by explaining the two most widespread strategies for professionally preserving digital items.

2.1.1 Long-term Timespan

Due to language ambiguities, every domain has a different understanding of the definitive timespan declared by the word long-term. Under German law so-called tax relevant data, for example incoming and outgoing invoices, receipts, inventory, balance sheets, and annual reports, have to be stored according to the German Commercial Code (Handelsgesetzbuch) § 257 S. 4 HGB and § 14b HGB of the Value Added Tax Act for a period of up to 10 years.

The obligation to preserve information is not restricted to the context of tax records. Sebastian Claudius Semler lists the following obligations in the context of the German health care domain.

> “The health history of a deceased adult who has died in a hospital has to be stored for the next 10 years. When the patient dies in a hospital and

> is under the age of 18 years old, then all records have to be stored for 20 years and, in all other cases, the data must be stored for the next 30 years. Englev and Petersen describe relevant documents for clinical trials described in Paragraph 8 of the (Englev and Petersen, 2003) have to be stored for additional 15 years after the study finished, which in case of long-term studies can be multiple times longer.“ (Semler and Ripkens-Reinhard, 2011)

In the context of the economy, the term long-term preservation already applies to data that must be stored for 30 to 50 years. In the context of state archives in Germany, preservation means to store for an indefinite period of time and irrespective of technological changes (Schrimpf et al., 2018). No object that is placed inside an archive will ever leave.

Especially for digital data, this requirement becomes a challenge, since the half-life of storage media for digital items is very small compared to that of their physical ancestors (Nano, 2008).

Not all digital items are visible with the naked eye. Some storage media types, like the microfilm (reels), microfiche (flat sheets), and aperture cards, are purely analog. Still in use today, the format requires external devices to bring the content back into a human visible form. These devices have a long but also finite lifespan.

The lifecycle of a device starts with the design and construction of the device itself and will operate for years, hopefully for decades. After this, the device has to be exchanged in order to bring the medium into an interpretable form again. Long-term preservation does not only include the lifetime of the medium on which the information is stored itself, but all applies to the necessary devices to bring the information into a form that can be interpreted.

2.1.2 Memory Organisations

An archive is a facility to collect, order, store, administer and use mostly written or other remains of an area of governmental administration or other public or private institution (Nestor, 2008).

The focus of the collection that is preserved is dependent on the operator. State archives run by the government also tend to not only store folders produced by the government itself, but also incorporate donations and conduct specific acquisitions of material for ingestion.

Archives run by the Federal Republic of Germany are governed by the Federal Archive Law (Bundesarchivgesetz) and include:

- State Archives
- Country Archives
- City Archives
- Schools

- University Archives

In Germany, the so-called State Archives have the special task of keeping the political actions of the state transparent for future generations. Every piece of information that is generated alongside political actions is judged by a group of archivists and, if determined to be of value, are moved into the archive.

Lukas Zitzer describes the main challenge of archives and memory institutions:

> "The main problem for so called memory organisations is that because of limitation of space a selection has to take place, deciding what is kept and what will be lost. In hindsight it might be obvious which information should have been kept, which is why everything should be archived." (Zitzer, 2008)

Storing all data would be the best way to not lose any valuable data but in the real world it is of course not suitable due to budget and legal constraints. Appraisal is the process of determine if an artefact is entering an archive or not. Part of it is assessing the format, condition and preservation needs (Schrock, 1996). Same as it is done for open-data collections. Open-data projects that rely on crowdsourcing and publish an infinite dataset have of course different needs than an open-data project which publishes a finite dataset that has been generated by the entity them self. In both cases rules and policies have to be created in order to assess what is allowed to enter the dataset.

In the process of finding additional sources of revenue, archives have become creative in advertising and offering their services also to the public. The main state archive of Hessen, Germany for example rents out their personal to conduct investigations in their archives. Similar to museums important artefacts are presented in an appropriate way to foster additional research around a topic or for general consumption. Archives have transitioned from record keeping areas to centres around historical research, who publish their own magazines or papers around the history of the objects in their collections.

Content

Records are items that possess a value and have multiple formats. For example: documents, folders, maps, plans, seals, images, movies, audio material and other materials which might be stored electronically. Archives are not, however, only limited to storing artefacts that are of record types. Depending on the mission of the archive, items from pre-historic civilisations like vases, combs, tools, or other everyday items can also be of interest.

Since modern pop culture items are mostly produced digitally today (Palfrey and Gasser, 2013), archives do not need a physical location to offer archived objects; they can be purely virtual in the form of websites for example. Facebook today is already becoming the largest data storage for personal images or videos for individuals, documenting the life of an individual, sometimes from birth until death (Beaver et al., 2010). While it might be convenient to use and feels similar to an online storage service, Facebook does not offer guarantees for future availability for the content and might delete it at any time with the change of their Terms of Operations.

Curation

Governmental archives gain their possessions mostly from documents that are produced by the government itself. In Germany, every item that is produced in a governmental process will be sent to an archive eventually, after it is not actively used anymore and a certain waiting period has passed. The task of selecting what to archive then begins. Selection or curation of what will be preserved is part of the obligations of archives. Before starting as a government officer in an archive an oath has to be sworn to judge incoming objects according to the law. Storage and resources are limited and not everything will be deemed useful in the future by a particular institution.

Depending on the objectives of the organisation, different viewpoints are examined before an artefact is ultimately archived or destroyed. Only a small part of all the records that are sent to a state archive meet these objectives and are kept in the archive for preservation. This selection process happens also in museums or other memory organisations. The location that can be used for storage of either machines or physical items is finite. Different organisations have different reasons why an item might fit into their collection.

2.1.3 Archiving of Objects

Different strategies for preserving physical objects have been developed over the last several hundred years. Processes to improve the lifetime of paper created a hundred years ago are now in place, as are methods to repair corrosion damage to artefacts like paintings. Optimal storage environments have been created.

When it comes to physical objects that have to be preserved, the environment is tightly controlled in order for it not to move beyond certain parameters. For digital objects, it is especially the environment that constantly changes.

Some of the changing parameters for the environment of digital objects are:

- Operating System
- Physical Interfaces
- Radio Transmission Interfaces
- Central Processing Unit Design
- Storage Hardware

Digital Preservation Strategies

Compared to physical archiving, digital archiving has a unique set of challenges. The interpretation of artefacts often requires a special device, like a monitor or mouse, etc. which suffer from device obsolescence. While some types of devices stay the same in their function over time, like a monitor, hardware like joysticks only have a limited life- and shelf-time.

Hardware interfaces and reading devices especially have an expiration date. While 20 years ago it was normal to store data on a 3.5"floppy disk, today it is very hard to find devices that are able to hold a floppy disk, or even to find compatible software drivers to run the disk drive under a current operating system version. The task of retrieving the data that is stored on a device is becoming more of a challenge in the near future.

Figure 2.1: Picture of a Floppy Disk

It is not, however, only the hardware that has an expiration date; data formats themselves also age over time and are subject to format obsolescence. It might not be possible to read the format with the current version of the originator program. This is true for word processing software as well as for images, videos, and audio data.

Software is only compatible with its environment for a limited period of time. If this compatibility is not actively pursued, an interpretation of the format becomes almost impossible. Encryption is another problem that might make data impossible to access for future generations. The key to encrypting a file is often not part of the preservation artefact.

Sensitive data that has been secured through encryption before entering the archive pose another challenge for digital archives. The data should be stored unencrypted to mitigate the problem of key management in the future but should also be save from malicious hackers or internal personal with access to the files. If the data demands the additional level of security, which archives have to determine by policy if this is data it wants, a possible solution is to utilize a similar process as the migration of file formats. In the authors opinion encrypted data should not enter an archive as it is an addition layer that has to be managed, maintained and cuts into the budget.

Even identifying what the format is that has been used for a file becomes a challenge. When the file format is either not in use anymore or has been declared obsolete by the creating software specialised tools might help identify the right format. There are open-source or commercial offerings that assist in the process of File Format Identification like PRONOM, JHOVE, GDFR, UDFR, Droid to name a few.

The Internet Archive is a web archiving organization, making the past web available via the Wayback Machine since 2001. In its basic form the Internet Archive is collecting documents, images, video, code, and other files from the live web resulting in collections, and web archives (Graham, 2019). Beginning of 2019 Arunkumar and Devendran (2019) counted ca. 361 billion web pages; 1,406,181 movies;121,293 music concerts; 1,737,905 recordings; 5,279,432 texts in the archive. To preserve these files and enable access to viewing and interacting to some degree with the captured websites the Internet Archive is keen on developing and sharing practices that have become more prevalent within academic libraries and archive (Jaffe and Kirkpatrick, 2009).

On October 17, 2003, at their 32th General Conference, UNESCO released the Charter on the Preservation of Digital Heritage. In this Charter, digital material was declared as a unique source of human knowledge that should have the same level of protection as physical heritage. Long-term preservation therefore becomes a topic of international importance that has to be solved transnationally (UNESCO, 2003).

Once the charter had been adopted, several workshops and conferences on digitization and digital heritage took place. The conferences produced guidelines and policies around the topic (De Lusenet, 2007) like the ”The UNESCO/PERSIST Guidelines for the selection of digital heritage for long-term preservation” (Choy et al., 2016). It launched a three-country preservation project on the state of digital material preservation in Africa involving Botswana, Ethiopia and South Africa to support regional implementation of the charter and the guidelines (Kalusopa and Zulu, 2009).

Two preservation strategies have emerged for the conceptual object of digital data (Nestor, 2008):

- The Migration deals with a constant change or evolution of the file format.

- The Emulation deals with preserving through simulation of the software environment that is necessary to interpret a format.

Emulation

The emulation approach has become popular with video games. Games that are only playable on older consoles are still playable today thanks to software emulations for current operating system platforms. Lots of older game consoles are either not sold anymore or are incompatible with the current interfaces of televisions or computers.

The same problem exists with data formats. Software is necessary that runs on current operating systems and is able to interpret the formats used. Old software is bound to certain operating system versions. If no drivers are available, it is only a matter of time until it stops functioning on current computer generations.

Building emulators is a very time consuming and resource intensive task. An emulator has to be built per operating system type and version. It has to be actively maintained to run on future versions of the operating system. The emulator itself is also software and is bound to the same rules as the object of preservation interest.

Unfortunately, knowledge about platforms is getting lost over the years, and it is also only a matter of time until the maintenance for a specific platform is stopped. This is the reason for the concept of emulating the environment for the emulation. Emulators for old operating systems are built to run older emulators, which are only supported by this platform.

The National Library of Germany is actively doing research in the area of emulation and has integrated a software solution in the reading room of the library.

> “The German project EMiL (Emulation of Multimedia objects in Libraries) focused on providing access to multimedia objects of the 90s that usually are stored on CD-ROMs. Objects such as digital encyclopaedias or interactive art pieces are difficult to present using current access systems in a reading room or exhibition space, since their original, now out-dated run time environments typically are not supported by the existing access system. This calls for an emulation of the vintage run time systems in order to provide access.“ (Steinke et al., 2016)

Migration

The second strategy is called migration. When archivists decide that a format is not widely in use anymore, or when there are only a few programs that are able to interpret the format, it is labelled obsolete. Tools like the PRONOM Database help archivists to find more information about a certain file format and determine their usage status. The search for new formats that have similar characteristics to the old format starts. If a successor has been found, automated ways have to be invented to convert the old format into the new one. Conversion is not a lossless process. Certain representations might not be reproducible with the new format.

The loss of information is accepted in this approach because a part of the content is preserved. Whether or not a book is just an accumulation of text or if a certain appearance is part of the book experience is a philosophical question. Emulation provides a closer experience to the original software or hardware but emulators have to be maintained, which is a time and resource intensive task. Knowledge about how to use the original software has to be preserved as well and, in certain situations special hardware has to be maintained.

2.1.4 Interpretation Model

The process of digital preservation includes more than just preserving the binary stream of a file. The format and its relevance are both of the utmost importance for the successful reconstruction of a file.

Stefan Funk defines the three levels for defining a digital object (Nestor , 2008):

- Physical Object
- Logical Object
- Conceptual Object

The physical object is the pure bitstream that can be detected on a data medium. An example is found in the hills and valleys that are burned onto a compact disc (CD-ROM).

The logical object is a single file in the bitstream. For example, a sequence of bits in a bitstream might be identified as a music title. On this level, individual bitstreams can be correlated with file formats.

The conceptual object is the interpretation of data to use the content to accomplish something. To stay consistent with the example, in the case of the CD-ROM, this means playing the audio file or detecting a movie burned onto a DVD (Digital Versatile Disc). The person who interprets the format correctly is able to use the content.

An example that demonstrates the importance of interpreting data at all three levels is found in the Lunar Orbiter Image Recovery Project (LOIRP) conducted by NASA (National Aeronautics and Space Administration of the United States) (Oldenburg, 2010).

Five satellites flew around the moon between August 1966 and August 1967 in a low orbit to fully cartograph it and to find a landing site for the moon mission.

The images of two cameras were exposed to film, developed, fixated, and scanned electronically on-board the satellites. The images were sent to Earth, where they were stored on magnetic tapes. 3062 images of the moon were received and stored on 1500 magnetic tapes. The magnetic tapes were archived but the readers for those tapes are no longer available. Between 2004 and 2010, a team of scientist was tasked with restoring the images from the tapes.

As a precaution to potential data loss, an archivist from NASA bought four magnetic tape readers and stored them in her garage at home during the time the images were produced. Years later, the magnetic tapes readers were refurbished by the new team and the data could be read. However, the file formats used to store the images on the tapes were not documented at all. By accident, a hint in the form of an audio comment on one of the tapes was discovered, which caused a breakthrough in deciphering the file format. All the images were restored; hidden in those images was the first picture of Earth taken from space.

This example demonstrates that it is not enough to preserve just the bitstream; one must also preserve the documentation of the file format including the meta-data, of the storage medium and of the format for the file itself.

2.2 Storage System Architecture

This chapter details the main architectural styles of storage systems. As understanding the client server and peer-to-peer architecture is fundamental for this report, these architectures will be detailed in the following sections.

2.2.1 Storage

A file system is an abstraction mapped onto one or more drives to provide a computer with a consistent access pattern to store and retrieve data permanently.

Christian Bandulet distinguishes between three different main categories for file systems (Bandulet, 2012):

- Local Storage
- Shared Storage
- Distributed Storage

Local Storage

A local file system is a system that can be triggered through a internal data bus on the motherboard. The data is only available for the machine to which the file system is physically attached . If more storage capacity is needed, this can be achieved by adding more disks to the computer's bus, or by interconnecting several drives to form a larger disk.

Figure 2.2: Local disc attached to computer

Shared-Storage

A shared-storage system is connected to a computer through a special network connection. For the computer, requesting a file on the shared-storage system looks like a local disc. For the computer requesting a file, there is no difference between a local drive and a virtual drive that is attached through a network.

Storage systems that fall into this category include, for example, SAN-Filesystems (Storage Area Network) or cluster filesystems like Apple Xsan, Lustre or the Red Hat Global File System. This category of storage system often needs a special kind of network (e.g. fibre channel).

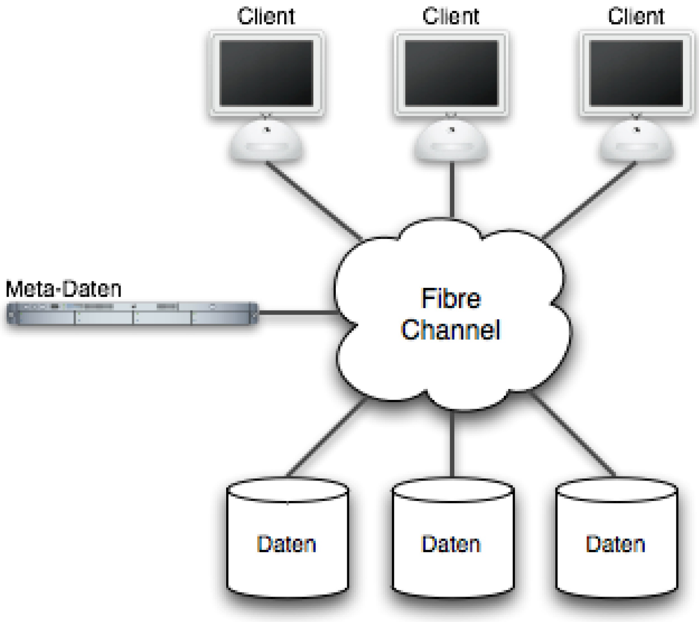

Figure 2.3: Fibre Channel Setup

To provide more storage capacity than a single drive can deliver, a separation in the meta-data server and in one or more storage-server is conducted. For performance reasons, these are separate servers connected with the network. The meta-data server contains the index information were files are available on the storage-server. A file is not requested in the usual file-centric way, but rather in individual blocks; this is similar to how a disk-based system requests data.

A request for a file is directed to a meta-data server. The meta-data server reroutes or forwards the request to the storage server containing the file or at least part of the file that is needed to assemble the requested file. This enables the shared-storage system to store blocks on different drives and raise the total capacity of the system to the combined capacity of all drives. A shared-storage system is able to handle multiple clients at the same time (Ghemawat, Gobioff and Leung, 2003).

Distributed Storage

Distributed systems differ from shared-storage systems in the way that files are stored between the servers and the representation to the computer accessing the system. A file is provided to a computer through a network protocol on the OSI model layer 7, like HTTP, FTP or WebDAV (Zimmermann, 1980). The protocol itself is implemented in the user-space rather than the kernel-space of the operating system. The protocols are object-based, which enables a computer to request a file, and blocks of data are streamed to the client.

An example for a user-space filesystem abstraction layer is the FUSE (Filesystem in Userspace) interface for the Linux kernel. A FUSE file system is typically implemented as a standalone application. A library provides functions to mount the file system, unmount it, read requests from the kernel, and send responses back. In both cases, incoming requests from the kernel are passed to the main program using callbacks. Kernel-space filesystems are directly implemented in the kernel and offer higher throughput then user-space filesystems which have to travel through the kernel multiple times (Vangoor, Tarasov and Zadok, 2017).

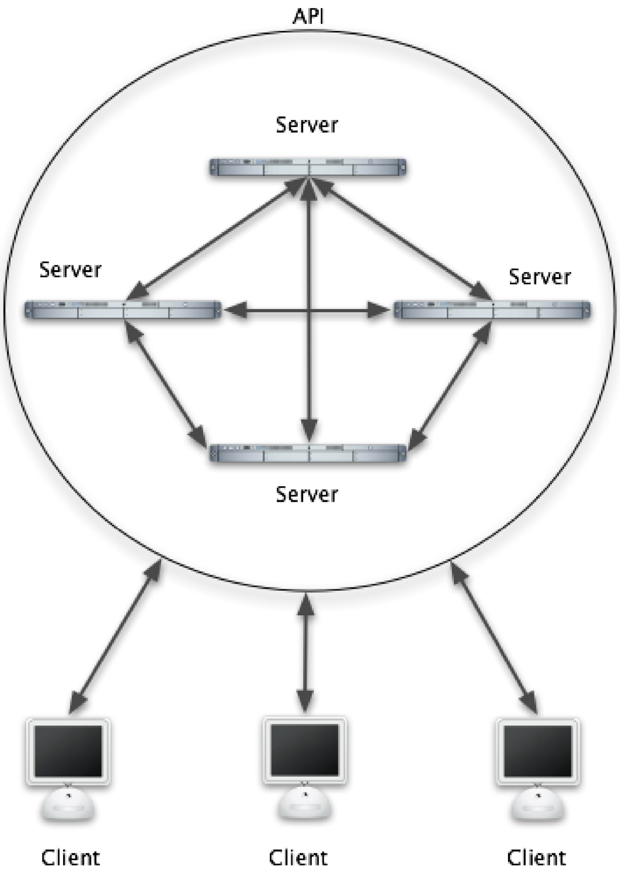

Figure 2.4: Peer-to-Peer System

2.2.2 Client-Server Architecture

To understand the design decisions influencing the protocol details in chapters 4 and 5, it is important to understand the design pattern of the client-server architecture, clarifying the technical side and practical applications of the pattern.

The client-server model is the standard concept for the distribution of tasks in a network. A server is a program that provides a service. Another program allows the client to use the service. The communication between the client and the server depends on the service, which means that the service determines which data are exchanged between both parties. The rules for a service's communication (format, request by the server, and the meaning of the data) are defined by a protocol. This protocol itself is specific to every service.

With the client-server pattern, it is possible to distribute tasks or services in a network to other computers. The pattern requires at least two participants: a client-machine and a server-machine. Both have to be connected through a network. The client requests tasks to be solved on a server; for example, asking for a resource like a file. The server, which runs on a different machine in the network, answers for example by providing the requested file.

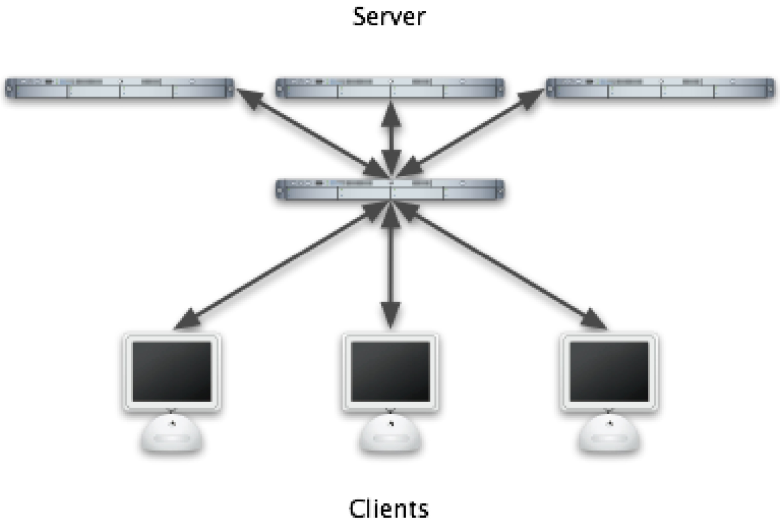

Figure 2.5: Client-Server Architecture

Clients actively query for a service that is in contrast to the passive server that waits for queries; they do not initiate a connection by themselves. The server always has to be ready to accept new connections from clients, since it is not possible to know when a client will connect.

Sacha Berger describes the following reasons to move a service onto a server (Berger, 2014):

Performance/Computational Capacity: Clients used to be machines with weak performance that provide access to resource intensive applications. The computation is done on a performant server. Compared to the origins of computer computations, the capacity has become quite cheap. The majority of clients are no longer low powered machines. This approach is used, for example, in the film industry to render animations of computer-generated images.

Centralisation of Data: Centralisation of data concerns the aspect of the client server model that has the most relevance today. Many different clients are able to read or manipulate the data that are hosted on a server. The effort to synchronise parallel incoming requests is minimised by centralising the hosted data to provide consistent data. The minimised effort is due to the fact that the redundancy of data is minimised. This fact also introduces a disadvantage since the system hosting the data now becomes a single point of failure. If the system goes down, all the clients are affected.

Examples of client server systems

- Fileserver: A server provides clients with a file system. It is the responsibility of the server to provide access management and transactional safety (only one client is allowed to modify data at a time).

- Web server: The web server (processed on a single machine or distributed over several machines) provides clients (web browsers on different machines) with information. The information might be of a static nature in the form of a file (the behaviour is similar to a file server) or dynamically generated by another program.

- Print server: A server (machine or process) provides clients with access to one or more printers. Typical tasks include the ordering of print requests in queues, access control, calculation of printing costs, or the selection of the most suitable printer for the order.

2.2.3 Peer-to-Peer Architecture

Peer-to-Peer (P2P) connections and machine-to-machine connections are synonyms for communication among equals (Ilie et al., 2004). Every computer is at the same time a client and a server. In this thesis, they refer to a network of computers.

In this chapter, three definitions are shown to capture the essence of P2P architecture, sharing own resources, directly accessible and able to request and provide services.

A definition from Clay Shirky (2000) for P2P-Applications:

> "An application which satisfies one of the two points is a P2P-Application:
>
> 1. Participants are able to connect and disconnect dynamically and receive (if necessary) temporary network addresses. 2. Participants have a high degree of autonomy and are therefore equals." (Shirky, 2000)

Rudiger Schollmeier has a slightly different definition:

> “A distributed network architecture may be called a Peer-to-Peer (P-to-P, P2P) network, if the participants share a part of their own hardware resources (processing power, storage capacity, network link capacity, printers). These shared resources are necessary to provide the Service and content offered by the network (e.g. file sharing or shared workspaces for collaboration). They are accessible by other peers directly, without passing intermediary entities. The participants of such a network are thus resource (Service and content) providers as well as resource (Service and content requestors).“ (Schollmeier, 2001)

Andy Oram concludes that the core of these definitions is the following:

> “Peer-to-Peer is a class of application who exploit resources (storage capacity, CPU-time, content, human labor), which are with the participants (peers) of a network. Exploiting these decentralised resources means to adapt to an environment that has non-persistent network connections and no predictable network addresses. This means that P2P applications have to operate independent from the DNS systems and have to be totally independent (autonomous) from servers. Autonomous means that the application has to be a client and a server at the same time or expressed in different terms, every participant in a P2P network is equal. Although only a few P2P systems obey the highlighted autonomy. In most systems there are so call super nodes, who possess more rights than the rest of the network.

> In a P2P network, every machine is equal and is able to request or provide services. Machines can be used as workstations but also fulfil tasks on the network. Core component in a P2P system is the overlay network which provides the peers with function of lookup and search. As soon as the clients have been identified in the network who possess the wanted file or service, a direct connection between the two is initiated.“ (Oram, 2003)

In general terms, P2P can be defined as an application which satisfies the following points:

- All nodes have equal rights. There is no central node in the system.
- A node is requestor and provider at the same time
- Nodes follow simple instructions which result in complex behaviour

Characterisation of Peer-to-Peer Systems

P2P architectures have special characteristics that differentiate them from other architectures. For example:

- Peers are very heterogenic regarding Bandwidth, Computational power, time online, and other resource features
- The availability or the connection quality of the peers is not predictable (Churn

Rate).

- Peers provide services and resources and demand the services of other peers.
- Services and resources are shared between all participants of a network
- Peers construct an overlay network and therefor provide additional search or lookup functionality
- Peers have a high degree of autonomy
- A peer-to-peer system is self-organised.

Classes of Peer-to-Peer systems

Kargl, Schlott and Nagler-Ihlein (2003) distinguishes between different classes of P2P systems regarding technique and area of operation (Kargl et al., 2003). Kargl, Schlott and Nagler-Ihlein (2003) classified the P2P systems into four classes:

1. Central coordinated or semi Peer-to-Peer The coordination and switching between peers is supervised by a central server. A peer has to request information about the network or resources from the central server. Only the data or information exchange is done between the peers. Examples for this architecture are Napster or the chat system ICQ.

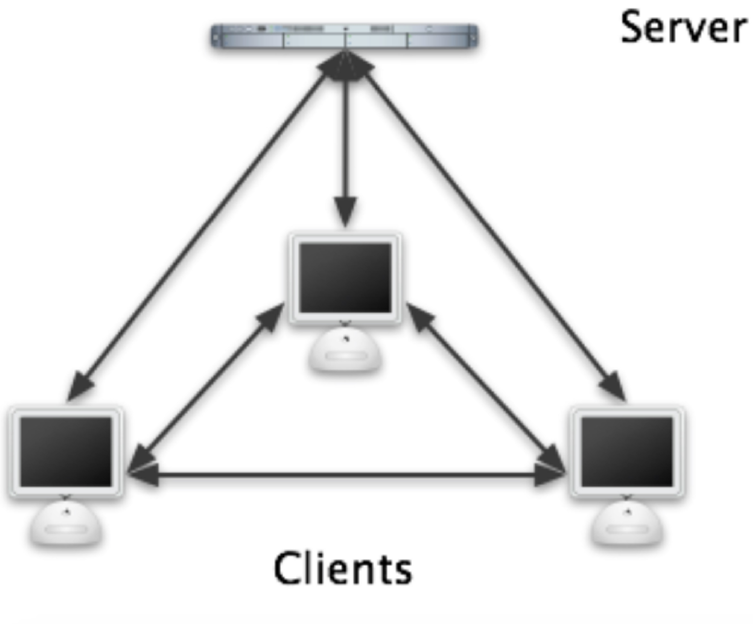

Figure 2.6: Central coordinated or semi Peer-to-Peer

2. Hierarchical Peer-to-Peer Peers are ordered in a group hierarchy and each group has a local coordination node (super node). Interactions between different groups are routed between the coordination node or a higher hierarchy. Data and information exchange is conducted directly between the peers. Examples for this architecture are DNS and the Squid-Proxy.

3. Decentral or pure Peer-to-Peer There are no coordinating nodes. The coordination, meaning discovery, name resolution, and resource finding, is all done in a decentral manner. Every peer is equal. These systems often use a forwarding algorithm, which will ask neighbours first, who will ask their neighbours, etc., until a peer is able to answer the request. Examples of this architecture include Gnutella 5 and Freenet.

4. Meta Peer-to-Peer Clients Meta P2P clients are not independent P2P architectures;

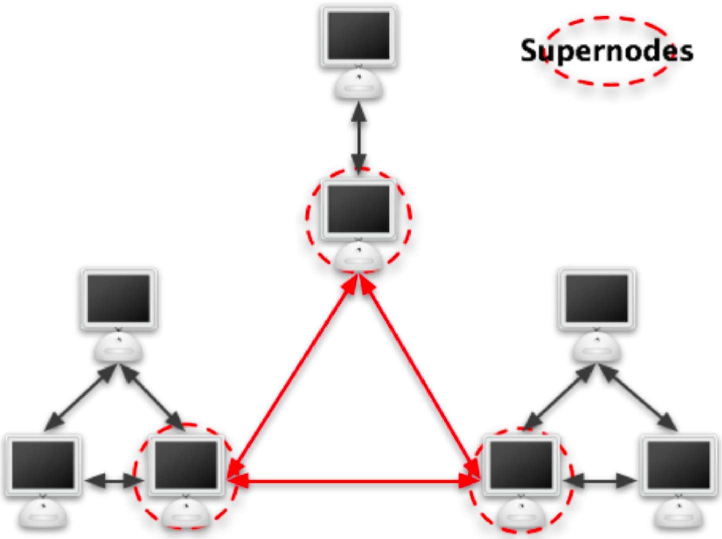

Figure 2.7: Hierarchical Peer-to-Peer

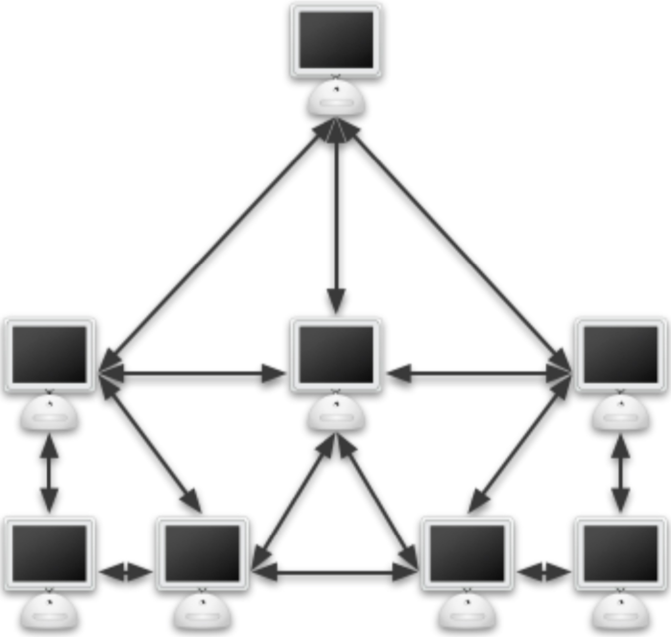

Figure 2.8: Decentral or pure Peer-to-Peer

rather, they connect at a different layer and enable the user to access more than one network of type 1-3 at the same time. The user is able to use a consistent interface to query a network of resources. The protocol is determined by the network that is used. The software MLDonkey is an example of such a client.

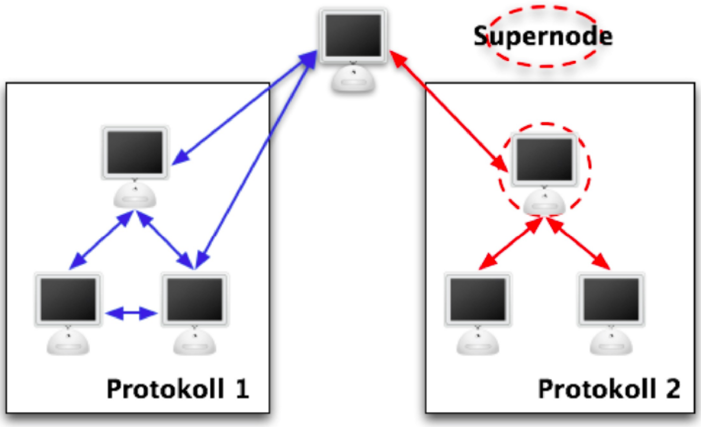

Figure 2.9: Meta Peer-to-Peer Clients

2.3 Open-Data

This section provides an introduction to the topic of open data. After examining the movement surrounding open data, the focus is on the open data movement by governments. The final part of this section showcases a comparison of participation in open data from different governments. This will give us an insight into the types of data that are available to the public.

2.3.1 Open Data Movement

Open data is the idea that some data should be freely available to everyone to use and republish as they wish, without the restrictions of copyright, patents, or other mechanisms of control (Bizer et al., 2009).

To establish a legal framework for openly sharing data between untrusted or unknown participants, the data that are shared have to be governed under a permissive license. Data sets that are released under such a permissive license are called open data. Anyone can copy, distribute, or embed such data as they see fit, without having to pay royalties or even negotiate a license agreement (Engelfriet, 2010). It is legally allowed to store, copy, and display the data; this is not true for closed data, where in most cases a license has to be bought in order to utilise the data in predefined ways.

Open-data can appear in many forms. The Open Knowledge Foundation categorises the following sources (Molloy, 2011):

- Cultural: Data about cultural works and artefacts — for example titles and authors — and generally collected and held by galleries, libraries, archives and museums.

- Science: Data that is produced as part of scientific research from astronomy to zoology.

- Finance: Data such as government accounts (expenditure and revenue) and information on financial markets (stocks, shares, bonds etc).

- Statistics: Data produced by statistical offices such as the census and key socioeconomic indicators.

- Weather: The many types of information used to understand and predict the weather and climate.

- Environment: Information related to the natural environment such presence and level of pollutants, the quality and rivers and seas.

- Transport: Data such as timetables, routes, on-time statistics.

Bizer, Heath and Berners-Lee (2011) proposed a rating system, called the 5 Star Data Schema which assesses the openness of format data is provided. The 5 Star Data Schema had an indirect impact raising the awareness of websites publishing open data. The Open Data Institute in London which issues certificates based on the openness of published data incorporated the 5 Star Model into their audit scheme. By July 2018, approximately 156.000 websites that publish open data have received a badge based on 4 levels of achievement. In their Open Data Barometer Report Davies stated that fewer than 1 in 5 datasets produced by governments worldwide are open arguing that little progress has been made in 10 years of open data (Davies, 2018).

2.3.2 Open Data By The Government

With the revision of the Open Data and Public Sector Information Directive from the EU, open data is a topic for European governments.

While there are no general rules about the types of data and in what state the data should be released, the Sunlight Foundation sponsored a conference for drafting principles around open data provided that the government would empower, if implemented, the public's use of government-held data (Krabina et al., 2012).

The Open Government Implementation model defines the following general guidelines for governments (Krabina et al., 2012):

- Completeness: Datasets released by the government should be as complete as possible, reflecting the entirety of what is recorded about a particular subject. All raw information from a dataset should be released to the public, except to the extent necessary to comply with federal law regarding the release of personally identifiable information. Metadata that defines and explains the raw data should be included as well, along with formulas and explanations for how derived data was calculated. Doing so will permit users to understand the scope of information available and examine each data item at the greatest possible level of detail.

- Primacy: Datasets released by the government should be primary source data. This includes the original information collected by the government, details on how the data was collected and the original source documents recording the collection of the data. Public dissemination will allow users to verify that information was collected properly and recorded accurately.

- Timeliness: Datasets released by the government should be available to the public in a timely fashion. Whenever feasible, information collected by the government should be released as quickly as it is gathered and collected. Priority should be given to data whose utility is time sensitive. Real-time information updates would maximize the utility the public can obtain from this information.

- Ease of Physical and Electronic Access: Datasets released by the government should be as accessible as possible, with accessibility defined as the ease with which information can be obtained, whether through physical or electronic means. Barriers to physical access include requirements to visit a particular office in person or requirements to comply with particular procedures (such as completing forms or submitting FOI requests). Barriers to automated electronic access include making data accessible only via submitted forms or systems that require browser-oriented technologies (e.g., Flash, JavaScript, cookies or Java applets). By contrast, providing an interface for users to download all of the information stored in a database at once (known as ”bulk” access) and the means to make specific calls for data through an Application Programming Interface (API) make data much more readily accessible. (An aspect of this is ”discoverability,” which is the ability to easily locate and download content.)

- Machine readability: Machines can handle certain kinds of inputs much better than others. For example, handwritten notes on paper are very difficult for machines to process. Scanning text via Optical Character Recognition (OCR) results in many matching and formatting errors. Information shared in the widely-used PDF format, for example, is very difficult for machines to parse. Thus, information should be stored in widely-used file formats that easily lend themselves to machine processing. (When other factors necessitate the use of difficult-to-parse formats, data should also be available in machine-friendly formats.) These files should be accompanied by documentation related to the format and how to use it in relation to the data.

- Non-discrimination: ”Non-discrimination” refers to who can access data and how they must do so. Barriers to use of data can include registration or membership requirements. Another barrier is the uses of ”walled garden,” which is when only some applications are allowed access to data. At its broadest, non-discriminatory access to data means that any person can access the data at any time without having to identify him/herself or provide any justification for doing so.

- Use of Commonly Owned Standards: Commonly owned (or ”open”) standards refers to who owns the format in which data is stored. For example, if only one company manufactures the program that can read a file where data is stored, access to that information is dependent upon use of the company’s processing program. Sometimes that program is unavailable to the public at any cost, or is available, but for a fee. For example, Microsoft Excel is a fairly commonly-used spreadsheet program that costs money to use. Freely available alternative formats often exist by which stored data can be accessed without the need for a software license. Removing this cost makes the data available to a wider pool of potential users.

- Licensing: The imposition of ”Terms of Service,” attribution requirements, restrictions on dissemination and so on, acts as barriers to public use of data. Maximal openness includes clearly labelling public information as a work of the government and available without restrictions on use as part of the public domain.

- Permanence: The capability of finding information over time is referred to as permanence. Information released by the government online should be sticky: it should be available online in archives in perpetuity. Often times, information is updated, changed or removed without any indication that an alteration has been made. Or, it is made available as a stream of data, but not archived anywhere. For best use by the public, information made available online should remain online, with appropriate version-tracking and archiving over time.

- Usage Costs: One of the greatest barriers to access to ostensibly publicly-available information is the cost imposed on the public for access–even when the cost is minimal. Governments use a number of bases for charging the public for access to their own documents: the costs of creating the information; a cost-recovery basis (cost to produce the information divided by the expected number of purchasers); the cost to retrieve information; a per page or per inquiry cost; processing cost; the cost of duplication etc.

2.3.3 Case Study Governmental Open-Data

Government of Germany

The government of Germany publishes their open-data on the portal [1]. 16596 datasets have been released by the government until now. Ranked according to the criteria set out by the World Wide Web Foundation in their Open-Data Barometer Report from 2017 Germany is ranked #14 (Davies, 2017).

Several Governmental departments are offering data sets for free use. This data sets

[1] https://www.govdata.de

include:

- Citizens: Households per district, population density, net dwelling area
- Geography and Geobase data: Street Maps, House Maps, Areal Photos, City Boarders as GPS Data
- Health: Registered doctors, density of doctors
- Infrastructure: Street Map, Votes, Public Household per district
- Politics: Voting results
- Education and Science: Patents, Students per year, Museums exhibitions visitors

Frankfurt am Main, Germany

The city of Frankfurt am Main in Germany publishes their open data on the portal [2].

Several governmental departments are offering 44 data sets for free use. This data sets include:

- Citizens: Households per district, population density, net dwelling area

[2] http://www.offenedaten.frankfurt.de

- Geography and Geobase data: Tree Maps, House Maps, Aerial Photos, City Boarders as GPS Data

- Health: Registered doctors, density of doctors

- Infrastructure: Street map, Votes, Public household per district

- Politics: Voting results

- Education and science: distribution of school education per region

UK Government

The Government of the UK publishes their open data on the portal [3]. Ranked according to the criteria set out by the World Wide Web Foundation in their Open-Data Barometer Report from 2017 the UK is ranked #1 (Davies, 2017).

Several UK Governmental departments are offering 26495 data sets for free use. This data sets include:

- Department for Transport: These files provide detailed road safety data about the circumstances of personal injury road accidents in GB from 1979, and the types (including make and model) of vehicles involved

- Highways Agency: This dataset provides information on planned roadworks carried

[3] https://data.gov.uk

out on the Highways Agency network. Roadworks listed cover the period up to 15 days in advance from the date of publication

- Environment Agency: This metadata record is for Approval for Access product AfA458. Light Detection and Ranging (LIDAR) is an airborne mapping technique, which uses a laser to measure the distance

- Driver & Vehicle Standards Agency: List of all Active MOT Vehicle Testing Stations in England, Scotland, and Wales including addresses, contact numbers, and test classes authorised

- Met Office: Monthly historical information for 37 UK Meteorological Stations. Most go back to the early 1900s, but some go back as far as 1853.

- Department for Business, Innovation and Skills: Presents quarterly price and cost indices are a basic tool of the trade to anyone involved in estimating, cost checking and fee negotiation on public sector construction works

Plymouth, UK

The city of Plymouth in the UK publishes their open data on the portal [4].

Several governmental departments are offering data sets for free use. These datasets include:

[4] http://www.plymouth.gov.uk/homepage/councilanddemocracy/information/opendata/dataplay.htm

- Spatial Data: listed buildings, public slipways, green spaces

- Management Structure: salaries

- Education Data: information on UK secondary and primary schools, their locations and much more

- Health and Social Care Data: data on UK healthcare to increase patient choice, improve patient outcome, deliver increased productivity and contribute to the UK's economic growth.

- Ordnance Survey: free maps and services accessible via an API or as a local download. A variety of solutions for geographical boundaries, post codes, place names, streets, original OS maps, contours and more

- UK Postcodes: a web service for postcodes in Great Britain and Northern Ireland, as well as geographical information, it also returns information about council areas and constituencies

2.3.4 Summary

This chapter introduced the context of the research by providing a brief introduction to the main topics surrounding long-term preservation, storage-systems architecture, and open-data.

The first section gave an introduction to the topic of long-term preservation. It moves from physical objects archived by organisations with different ambitions, to explaining what kind of objects are preserved and why. The section closes by explaining the two most widespread strategies for professionally preserving digital items. Section 2 details the main architectural styles of storage systems. As understanding the client-server and peer-to-peer architecture is fundamental for this report, existing architectures are detailed. Providing an introduction to the topic of open data is done in the final chapter. After examining the movement surrounding open data, we focus on the open data movement by governments. The final part of this section is giving an insight into the types of data that are available to the public produced by the Government.

3 Weaknesses in Preservation Approaches

A centralised project philosophy has enabled digital collections to grow, but at the expense of its overall effectiveness of being preserved for the long-term. As preservation of digital collections might become necessary due to a possible closure of the main initiative, certain weaknesses in centralised project philosophies have emerged, which are now beginning to pose a threat to the accessibility of digital collections. This chapter provides a comprehensive analysis of these weaknesses in order to determine the extent of the problems, and describes how the different solutions that have been proposed have not produced the desired effect.

3.1 Introduction

Preservation of digital objects is a new area in the established field of long-term preservation, and many studies have been conducted to cover these new challenges. For example, Thibodeau (2002) has examined challenges regarding policy questions, institutional roles and relationships, legal issues, intellectual property rights, and metadata of digital collections. He summarizes those challenges as substantial at the empirical level; Factor et al. (2009) have shown that strategies have to be implemented to be prepared for changing technologies, formats and communities ; many studies, such as Duranti (1995) report on the challenge of preserving authenticity in the digital age, particularly the problems they face when the physical location of the data is distributed across a number of stewardship organizations running heterogeneous and geographically dispersed repositories (Caplan, Kehoe and Pawletko, 2010) and an equally large number of studies, such as Witt (2008), and Zhong et al. (2013), have reported on the problem of digital curation in distributed environments.

Archives should not merely horde data, they should promote the active use of data. Digital curation is an addition al discipline that follows the pure preservation process. Lord (2004) describes the activity of data curation as managing and promoting the use of data from its point of creation, to ensure it is fit for contemporary purpose. This also means that data has to be monitored for problems that arise on the logical level of preservation, for example legal issues. While digital curation is an important part of the overall preservation process this thesis focuses on the technical validation of data and the problem of hosting in a decentralised environment.

A comprehensive analysis was provided by Hofman (2002), which analysed the new set of digital preservation challenge in great detail in order to asses proper management of digital records and related preservation strategies. These strategies may include developing standards for storage formats or preservation metadata, legislation, prescribing standards, or developing reference models for preservation functions or repositories, or defining minimum sets of metadata that allow the ongoing maintenance and access of these information resources. (Hofman, 2002)

Hofman (2002) identified three main areas of challenges, which derive from the fact that digital information is volatile, intangible and mutable. An overview of these challenges is presented here:

- Missing Authenticity: When a digital collection is archived by an entity that is not part of the core community, distrust might arise if a data set has been altered. The authenticity is associated with the core community building the data set, not with the collection in general.

- Missing Collaboration: For important collections multiple communities arise that have no way of improving a single collection together since all are only improving their own dataset. As importance is measured in size of the community, a dataset might become superseded by another one with the same purpose, solely to have a slightly different focus.

- Missing Accessibility: Datasets all have their own method of accessibility. There is no common way of accessing a dataset with a permanent infrastructure. Once a community decides to abandon a project all ways of contributing or tools around the dataset vanish as well.

The preservation challenge is to provide enough computing capacity, bandwidth, technological knowledge, electricity, and budget to keep datasets accessible to interested parties. This is not a challenge for a single publicly or privately-funded entity, a certain domain like memory organisations, or a country. This challenge has become a global issue only solvable by building partnerships and networks of interested organisations, not only within certain domains, such as a library or archival communities but also across domains (Hofman, 2002).

These three weaknesses pose the greatest threat to the accessibility of digital collections, and must be solved in order to preserve them for future generations beyond the lifetime of the original initiative. However conventional solutions rely on a centralized storage approach. This effectively hinders preservation at multiple locations. As such, the three weaknesses represent the concrete realization of the centralised open-data community projects dichotomy: A centralised hosting philosophy is leading to a trusted project but ineffective preservation system and a distributed hosting philosophy leading to an untrusted project but effective preservation system. This is the key problem that the research described in this thesis attempts to solve: How can digital collections be hosted and preserved decentral and still retain its authenticity?

The following sections provide a detailed examination of the root cause of these weaknesses, revealing why they are so dangerous to a digital collection, and discussing some of the solutions that have been proposed, but have not worked.

3.2 Missing Authenticity

> “One unique aspect of digital information is that it can be copied perfectly and that the perfection of a copy can be verified without human effort or intervention. Yet Archivists, librarians, museum curators, historians, scholars, and researchers in various fields define authenticity in distinct, though often overlapping, ways. Different types of digital informational entities that fall under a given authenticity principle (within a given domain of use) may have different specific authenticity criteria. For example, authenticity criteria for databases or compound multimedia entities may differ from those for simple textual entities.“ (Rothenberg, 2000)

The British Library (2017) stated in their Digital Preservation Strategies that Authenticity is the quality of being genuine and free from tampering or alternation, malicious or otherwise. They also note the fact that is also often necessary to change data, in order to manage and ensure access over time. One unique aspect of digital information is that it can be copied perfectly and that the perfection of a copy can be verified without human effort or intervention. Missing proof of authentication for a digital item lowers the level of trust placed on it as the item might not be in a state the author intended it to be. Specifically authenticity is a requirement when the task requires a high quality of confidence in the data and proven authenticity like it is the case for example for archivists and academic researchers.

3.2.1 Cause of the problem

Missing authentication can arise if the meta-data for an object is lost or the governing body has to abandon a collection. As the following list shows, there may be many reasons why a data set has to move to a new community or infrastructure:

- Lack of funding
- Lack of participants
- Lack of hardware
- Change of focus

However, even if a data set was to remain with the same community, its authenticity could still change. For example, this could occur due to:

- Malicious Hacker: An attempt to change the data set without having the appropriate right to do so.
- Hardware Failure: A data-center containing data to validate the data set fails.
- Human Error: A person deletes all backups of a dataset by mistake

3.2.2 Damaging Effects

Missing authentication for digital items results in loss of trust in using the digital object. Show et al. summarized the usefulness of authentication for digital items as following: Preserved resources and goods are only useful if their integrity and authenticity can be proven at any time (Schott et al., 2008).

With no way of proofing for example the original authorship there is no way of deriving this fact from the source alone.

3.2.3 Measuring

The measuring is binary either there is a proof or no proof exists. What constitutes as a valid proof depends on the governing society. In the legal system copies of a document can be authenticated through a notarial act as the trust has been placed with the notary to only authenticate valid copies. A group determines their specific use case proving the bit accuracy of document is sufficient to validate it as a copy of an existing document than different, automatic means of validation can be taken.

3.2.4 Existing Solutions

Authenticity can be provided in many forms. Either cryptographical methods, e.g. *md5* or *sha256*, are used to calculate a fingerprint of the binary stream or a trusted entity certifies the authentication of the data.

3.3 Collaboration

Collaboration deals with the synchronisation of participating entities either hosting or updating a single collection together. An entity is able to fulfil several roles either by hosting changesets, by creating a changeset or by validating a changeset.

3.3.1 Cause of the problem

Collections at a certain size require the collaboration from specialists from many fields. With different goals for a collection, participants are guided by different viewpoints and rules which result in natural friction in defining clear borders of responsibility.

3.3.2 Damaging Effects

Missing collaboration results in duplication of effort. Instead of combining resources to create, maintain or update a collection, the dataset competes with similar datasets for participants.

3.3.3 Measuring

Measuring collaboration is possible by assessing the number of competing initiatives to achieve a similar goal. Less competition means that resource maintaining the collection are focused on improving a small number of collections.

3.3.4 Existing Solutions

There are two main architectures for enforcing collaboration. Either everything is hosted by a single entity or multiple entities work together under a common rule set to host a library.

3.4 Accessibility

While entities not only host data, they also need to provide access to a collection. This is either for internal use or for the greater public. Access can be provided in many ways, e.g. by using an API mechanism, providing a direct download or similar ways.

3.4.1 Cause of the problem

Unreliable or missing access to a dataset can result from different reasons. For example, the uptime of the hardware that hosts the data plays an important point here. If the hardware is not running the data cannot be accessed. But also, financial considerations like pay for access business models for the dataset or certain data access methods can be a source of not being able to access a dataset. The way to discover a data set plays an important point. If a link to the dataset has to been known upfront and cannot be derived through a mechanism, the access to the data is lost simply because there is no pointer to the data set.

3.4.2 Damaging Effects

Open Data should be open and accessible by definition otherwise it is not open data. The missing access to open data can reduce the rate of discoveries. For example, science projects cannot be verified if the data is not available and have to be recreated from scratch, which sometimes is not possible. Also like in the case of online communities the missing access to data can be loss for the members. The online community Yahoo! GeoCities was founded in 1994 and closed in 2009. Roughly 38 million pages have been created in the meantime by the community. The announcement of closure was issued just 6 months in advance which gave archivist just a short amount of time to save as much as possible (Milligan, 2017).

3.4.3 Measuring

There are nuances in the way of measuring the accessibility of a data set. A dataset can be label if several pointers and methods of accessing the data exists. For example, the dataset is available via an API and also as direct download. While the API might be accessible through different wire formats like xml, json, graphql or similar mechanisms, the direct download can be access also through different wire formats like ftp, http or bittorrent. Due to the different methods it is possible to access the dataset in a way that suits the requestor of the data, which can be either a human or a machine.

3.4.4 Existing Solutions

Access provided to collections is possible in many ways. Either by offering an interface to navigate, search or download a collection or by providing programmatic access to data via the use of an API interface. While with the Open Archives Protocol for Metadata Harvesting standard there is a generic way to query meta-data, there is no standardized way that combines access to meta-data and files at the same time.

3.5 Summary

The chapter has focused on three core challenges faced by digital preservation according to Hofman (2002):

- Accessibility
- Authenticity
- Collaboration

The challenges have been recognized for some time, and various proposed solutions have been described in this chapter. The problem with the existing solutions has been that they are unsympathetic to the philosophy of open-data; the needs and behaviour of storage locations; and the needs and behaviour of communities. Specifically, a system that requires the replacement of existing infrastructure might not be adopted; a system that ignores the needs of the communities might not be used; and a system that assumes malicious user might not attempt to deceive the data will be rendered useless. As such, this research programme has set out to solve the centralised open-data dichotomy.

Most open-data initiatives are an organic system, complex and dynamic, and evolving according to the needs of the community, with the communities engaged in hyper-competition trying to attract as many users as possible. It is more like a society than a rigid information system, but this is to be expected, as it has virtually no barriers to entry, and so all areas of society contribute to it. As Berners-Lee puts it, "the web is a social creation not a technical one" (Berners-Lee and Fischetti, 1999).

As such, this thesis is based on the assumption that in order to contribute to collections effectively, a new and entirely different model of contribution is required, which is sympathetic to the existing architecture, the behaviour of its users, and of its data providers. Rather than hording data in single entity governed vault, the model should focus on bundling resources from many entities to not only share the effort to create and maintain a collection but also keeping save in many locations. Such a model has been developed as part of this research programme, and the remainder of this thesis will discuss its design, development and implementation.

4 A model for synchronising digital collections

Having discussed at length three core flaws of centralised preservation, this chapter presents a model for synchronising open-data between untrusted machines. The model is called Archive Networks, and has been designed to overcome the three flaws of centralised preservation while encourage open sharing of data by a community.

4.1 Introduction

A conceptual overview of deriving the model and the decision taken along the way is presented in this chapter. The remaining chapters discuss how the model has been applied in order to form the resulting protocol. Chapter 6 presents a prototype of the design, which has been developed to validate the design, and to provide performance data to illustrate the practicality of the protocol.

4.2 The Core Components of the Archive Network Model

The Archive Network model has been designed to solve the challenges of trust, collaboration, and formalized accessibility in the domain of long-term preservation. In order to achieve this, the model comprises three components, which together address these three challenges in distributed preservation approaches specified by Hofman (2002). Specifically, the core components of the protocol include:

- Global Timeline (Authenticity): A new validation scheme that allows all participants to validate the data integrity of a dataset

- Distributed Content (Collaboration): A synchronisation method orchestrating contributions to a decentralised dataset.

- Discovery & Access (Accessibility): Formalised access and discovery of decentralised storage locations

The focus of the Network Archive model is on an auditable, tamper proof multi master database replication, redefining distributed preservation, and ensuring the integrity beyond the storage location. The following sections describe how this is achieved.

4.3 Model Concept

The difference from a distributed archiving network to existing synchronisation approaches is the data structure used to hold and distribute the history. An archiving network is synchronised using a tamper proof, cryptographically linked chain of blocks. A block groups single change operations to a local database together. To form a time line, the blocks are linked by hash values. To seal a block, a cryptographic puzzle has to be solved and the solution is signed with cryptographic keys. Every node is able to verify a sealed block by validating the hash chain and the corresponding puzzles. If the sealed block satisfies the validation criteria the log is advanced, broadcasted to the other participants and used as the base for the next puzzle. The solution to the cryptographic puzzle is included in a block in order to make sure the network is not flooded by blocks and the order of the timeline becomes a matter of block creation time.

The block chain algorithm in combination with a cryptographic puzzle was detailed in (Nakamoto, 2008) to provide a distributed ledger for currency transactions. The block chain variant leveraged in this research is a simplified version of the algorithm that drops the mini programming language that allows complicated transactions types in the block chain.

Distributing the content of a web application was also researched by several other teams (Schaffert, 2006), (Urdaneta, Pierre and Van Steen, 2007), (Weiss, Urso and Molli, 2009), (Rahhal et al., 2008), (Oster et al., 2009). Their main focus was to distribute a wiki. Since the type of application was known before hand, it was possible for them to include fine grained conflict resolution mechanisms. A network archive simply applies all transactions from a mainline block, regardless of the semantical meaning a change might have for an application. Since the chain of blocks is the agreed time line of events, conflicts are overwritten by the latest change set in the block chain.

Another system that provides a distributed long-term storage solution is the LOCKSS system described by (Maniatis et al., 2005). The LOCKSS system has a specific focus on providing a closed group of peers' accesses to magazines, journals and books. LOCKSS also deals with access management as a license for the content has to be available per peer from the publisher. A peer with no valid license does not get access or lose access to the content. As an Archive-Network is not concerned with licensed content and does not require a formal membership of a contributor, the focus of the two solutions is on different use cases in the long-term preservation field.

The bootstrap mechanisms from bittorrent are reused (Cohen, 2003). Bootstrapping in the context of bittorrent is the process where a peer begins a download and has no information about other peers that are offering the data. The discovery of the other peers happens by contacting so called tracker servers that have the sole use to connect peers with each other based on the info hash data in the bittorrent protocol. There is an already working infrastructure with these tools that we can repurpose to bootstrap a network. A meta data file as configuration initialisation file for a network can easily be send via email or placed on regular bittorrent index sites as the file format is compatible with each other.

4.3.1 Influencing Factors

Decentralized hosting is truly democratic since there is no single point of authority that decides the validation rules for a block or update operation. The rules are decided by the majority of application users, not the host of the main database. The entry barrier for contributing data is lowered as infrastructure and contributors are separated. Fluctuation of application users does not have an impact on availability, accessibility of the data or the corresponding infrastructure since all machines are connected but can advance independently from each other.

By replacing the cryptographic puzzle with a public-key encryption method it is possible to also cater to communities which are not self-regulating or coalesce around a set of rules by consensus or majority. Only nodes that possess the private key are allowed to contribute data. This is a variation of the protocol that has not been further researched. Many systems already exist for a closed group of collaborators (Maniatis et al., 2005). The focus in this work has been on allowing truly open contributions.

By providing a standardized way of access to open-data, archives are able to become a connected part of the hosting infrastructure instead of stale backups of the main database. By contributing to a decentralized data set, participants prevent knowledge islands. The global data set is improved, instead of only one of many forks. Reviving an abandoned project becomes effortless since the infrastructure is still in place as long as one machine in the network holds the history.

An archive network can be seen as an extension to the 3-tier web architecture, acting independently from the web application. This 4th-tier is responsible for handling the communication with the network, creating blocks and validating blocks or transactions. The web application pushes data into the 4th-tier which becomes part of the network knowledge and data must be pulled from the network and imported back into the web application, to become part of the local knowledge. This approach makes it possible to decide if local data should be overwritten or not with data from the network.

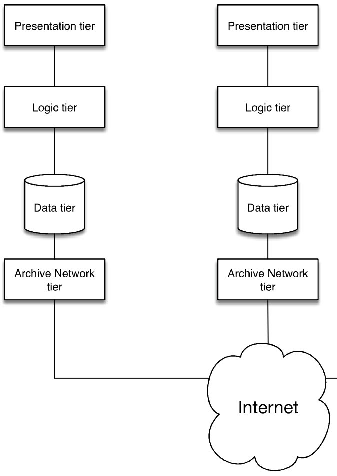

Figure 4.1: 4-Tier Architecture

The creation of new network results in two artefacts: The genesis block which is the root node of the block chain and a meta data file, with all information to distinguish the network from other networks. The genesis block is a special block since it does not have a precedent block and no transaction. All validation must start from this block hence the hash to identify this block is very important and should be well known in the network to make sure everybody validates from the same base. Similar to bittorrent (Cohen, 200) the meta data file contains information about how to identify a network. For example, the root block hash, tracker links where peers might be found, a title of the network and a description of the network.

Bittorrent uses a special purposed software, so called tracker software, to collect information about machines like the IP address and a port who want to participate in a torrent. Machines that want to join a torrent ask the tracker if it has other machines or peers that are running the same torrent. If this is the case information is sent back and the machine is able to contact the network. The archive network protocol piggybacks on this mechanism and thus uses an existing infrastructure to connect machines with each other. It let others find and join a network via a bittorrent tracker found in the meta data file.

When a machine receives information about other peers from the tracker, it downloads all the blocks these machines have stored. It then starts to validate if the genesis block is the same as in the meta data file. Subsequent all blocks are validated and if this succeeds all transactions in these blocks are validated. By validating all blocks from the root block by them self the application can be sure that it has the same block chain as all other machines. When a machine creates a new block, it distributes them to their peer which validates the block. Following a successful validation, the block is passed on until all machines have the new block in their chain.

4.4 Summary

This chapter has described the basic components and philosophy of the Archive Network model, which defines a new way of for synchronising open-data between untrusted machines. The chapter has shown how the model tackles the three identified core challenges faced by digital preservation without coming into collision with the centralised open-data dichotomy. In this way, it provides a new model for managing open-data decentralised that has been designed to work according to the open philosophy, and within existing architectures.

This chapter, however, has only presented a conceptual overview of the Archive Network model. As such, the next chapter presents its design specification in detail. The design is verified in chapter 6 by a prototype implementation of the Open Archives Protocol for Metadata Harvesting on an Archive Network, which has been developed and measured as part of this research programme.

5 Archive Networks – A protocol for synchronising digital collections

The Archive Network model provides a new model for synchronising open-data projects. This chapter describes the design of the building blocks that are necessary for the model. It presents in detail the model's architectural design, which enables the synchronisation of contributions to a decentralised dataset. Furthermore the designs of the global timeline, the distributed content, and the formalised access and discovery methods are discussed.

5.1 Introduction

This chapter presents the building blocks of the Archive Network protocol, a novel protocol for synchronising open-data between untrusted entities in a trusted way. The remaining chapters discuss the building blocks the protocol consists of and the behaviour of the different components. While all examples use the *XML* format (Bray, Paoli and Sperberg-McQueen, 1997), other data representations like *JSON* (Bray, 2014) or *binary* formats are possible. At the time of writing the default exchange format of the Rails Framework, which was used to code the prototype, used *XML* as default wire format of choice. Which wire format is used can be decided by the creator of the network. The corresponding software must simply be able to read and write the data format.

5.2 Data Containers

This subchapter declares the data formats that are used to encapsulate the data intended to be distributed. The order of appearance is from the most inner container, a change set, via a transaction, to the most outer container, a block. A block is able to store multiple transactions and a transaction is able to store multiple change sets. The data format hierarchy allows to parse, validate and access the data in a consistent format.

5.2.1 Change Set

A change set is a single transformation of the applications database wrapped into a predefined form. It is part of data that is exchange between machines to exchange data. The format depends on the data that needs to be exchanged with other machines to recreate the database.

For example, a network is created to let machines share their search index for books. If a book is added to the local database, the corresponding change set would contain the information that it was an addition and the fields that have been added to the database. For book modification the format would consist of the primary key identifying the book and the fields that have changed. A book that was deleted would simply be identified by the primary key. This also means that no data can be deleted once it enters the global timeline. The primary key is generated in the form of a Universally Unique Identifier (UUID) to ensure its uniqueness.

Listing 5.1: Change set segment in *XML*

```
<change-set>
 <addition>
 <book>
 <uuid>550e8400-e29b...</uuid>
 <author>Goehte</author>
 <title>Faust</title>
 </book>
 </addition>
</change-set>
```

With these three operations of create, update, delete, we are able to write a log file of actions and data that when replayed on an empty database, recreates the state on the empty database. The fields that a change set must have are predefined by the creator of the network. Syntax validation is applied to make sure that a change set is in the format the creator of the network intended it to be. Validation rules might include the length, the content, the uniqueness or regular expressions. The change set must obey the format. A machine that receives a change set that is not in the correct format, should drop it and the encapsulating block. Since XML is used in this example predefined DTDs can be used to announce the format for this network.

5.2.2 Transaction

A transaction is a container type that holds a change set. It has additional data information about who created the change set, and when as well as a hash of the content of the fields. To make a change-set temper proof the hash is built from the data of the change-set and the date when the change-set was created. By rehashing the change set data and the date, it is possible for other machines to validate that the content has not been changed.

Listing 5.2: Transaction segment in *XML*

```
<transaction>
 <hash>709813209487</hash>
 <created-at>
  2013-05-30T09:30:10Z
 </created-at>
 <change-set>
```

```
   ...
  </change-set>
</transaction>
```

5.2.3 Block

A block is a container type holding transactions. Additionally, to the transactions, a block contains the following data:

- Hash of the previous block
- Hash for this block
- Date when mining was started
- Creation date of this block
- Difficulty for the cryptographic puzzle
- Offset (nonce) for the cryptographic puzzle
- A date when the correct nonce was found
- A hash build by the hashes of all transactions in this block in lexicographic order

- The public key of the machine

Listing 5.3: Block segment in *XML*

```
<block>
 <previous-hash>
  0023987...
 </previous-hash>
 <hash>0002390d...</hash>
 <transactions-hash>
  3246234...
 </transactions-hash>
 <started-mining-at>
 2013-05-30T09:30:10Z
 </started-mining-at>
 <difficulty>4</difficulty>
 <nonce>439</nonce>
 <found-nonce-at>
 2013-05-30T09:45:47Z
 </found-nonce-at>
 <transactions>
 <transaction>
  ...
 </transaction>
 <transaction>
  ...
 </transaction>
 </transactions>
```

```
</block>
```

5.2.4 Chain of Blocks

The chain of blocks is not an element in itself but is built by the blocks and their connection with each other. The block chain is a tree structure with a single root branch, the genesis block. The number of nodes that a tree possess is not fixed. Blocks can come from all machines in a network and the tree structure accommodates for this by supporting branches.

To seal a new block, the machine takes the previous block with the greatest depth in the tree as a basis for the new block. The branch with the greatest depth in the tree is call the mainline branch. If the final node has two blocks the application should start with the older block. When a new block is received from the network and the depth is greater than the current one the application is working on, it should stop and take the new block as the predecessor. Orphaned blocks, blocks received from the network which have no previous block that is in the chain yet, are stored separately as the missing block might arrive later.

The chain of blocks can be validated by following the hashes and signatures in the chain. This might take some time depending on the number of blocks. Depending on the hashing implementation chosen the validation of a single hash takes between 25 and 80 rounds (CPU cycles) (Bertoni et al., 2013).

To accommodate for orphaned or split branches only the mainline branch should be applied to the application database. The mainline branch is the chain of blocks with the highest depth in the tree minus a predefined number of blocks from the top. As it is not clear when a new block arrives and if it belongs to the mainline, transaction should only be applied to the database if there are blocks in front of them. The buffer of blocks makes sure that no other branch overtakes the lead and changes have to be reverted on the database. Transactions that reside in a block that has been determined as orphaned have to be reschedule for integration into a new block.

Listing 5.4: Chain of blocks segment in *XML*

```
<blocks>
 <block>
  ...
 </block>
 <block>
  ...
 </block>
</blocks>
```

5.2.5 Cryptographic Puzzle

To seal a block with all transactions a cryptographic puzzle must be solved. To solve the puzzle, the hash of the block must begin with zeros. How many numbers must be zero is determined by the difficulty level. If the level is four, the first four numbers of the hash have to be zero. The data for a block is static in general. To generate a different input for the hashing function the nonce value is changed with random characters until the puzzle is solved. This is the only field allowed to change for generating different hash values. Searching for this special hash form takes time which is used to slow down contributions to the network.

The difficulty level must not be a fixed value. A formula containing the average time between the last 10 blocks might suite better if many machines take part in the network. The value or the formula to get the difficulty is decided by the creator of the network.

The root hash of the transactions is built by hashing all hashes of the transactions in lexicographic order. This saves up valuable time while solving the puzzle since only one hash has to be taken into consideration instead of all transaction hashes for every pass.

The hash for a block is built by hashing the values of the previous hash, the root hash for the transactions, date of beginning of puzzle solving, the difficulty and the nonce together. If the hash does not solve the cryptographic puzzle, the nonce value is changed. This creates a new hash that might solve the puzzle. This is done in a loop until the puzzle is solved.

To provide information who created the block, it is digitally signed using asymmetric keys. The block is signed with a private key of the user. The public key is used to verify the signature. If a system like PGP is used, additional information about the user can be stored while creating the keys. PGP maintains a public infrastructure storing public keys. Those can be retrieved from the service and validate that the transaction could only be signed by someone who owns the private key. This makes it possible to identify the source of a transaction. As a backup method the public key should be stored with every block to prevent that a public service for the keys vanishes and hurts the audit ability this way.

5.3 Discovery

A network is distributed by nature. The block chain can be used on a single machine but the true value of exchanging the blocks arises when other machines join a network. When a machine wants to join a new network, it asks a tracker from the meta data file if it possesses information about other peers. The network is identified on the tracker via the hash of the network. The tracker keeps records of which IP address asked for information about the network and relays any other machine that has asked before.

The network application then tries to contact the machines returned from the tracker. If one or more machines are found the block chain is reconstructed by downloading all blocks that exist in the network and validate them. When all blocks are validated the node is a full member of the network.

5.3.1 Meta Data File

The meta data file is the key to finding peers in a network. Based on the idea of the torrent file from bittorrent, the meta data file contains all information that is necessary to join a network. It is also syntax compatible with the bittorrent meta file, to reuse the existing infrastructure. The file consists of the following fields and is encoded in the bencode format:

- tracker links
- the info_hash for the network
- the hash for the genesis block
- a name for the network

With this file it is possible to initialize the network application and connect with other peers in the network.

The hash for the network is the key to identify the network. It is compatible to the info_hash value from bittorrent which allows us to use the existing infrastructure in form of bittorrent trackers to bootstrap a new machine with peer information.

The hash for the network is built by hashing the values of the genesis block hash and the name of the network. Since trackers are subject to change during the lifetime of a network, the remaining fields do not. The validation rules are subject to change and should not be tight to the meta data file.

5.3.2 Tracker

A tracker is a HTTP web server that connects peers interested in a special network. The purpose and syntax is identical to the bittorrent specification. Via a HTTP GET request the client announces that it is interested in finding peers from the identifying network. The tracker saves information about the client like IP address and Port.

- IP address
- Port
- How many blocks have been downloaded so far
- Info_Hash
- Event

The tracker response is a bencode dictionary.

To have an up-to-date representation who is a member of which network, the client should advertise itself every 30 minutes and when a new block is deemed valid. This makes sure that a new machine finds fresh peers quickly in new network.

5.4 Network

This section details the network part of the protocol. It answers the questions: How does a machine find other machines that are sharing the data and how should a machine act to achieve the desired storage behaviour To form the peer-to-peer network that underlies the archiving network, all machines have to react on events. These events follow a callback model and have been determined by the author. They have been derived from the events that happen in the life time of an application. Since a client might leave a network at any time, important information like other peers and the block chain is spread to all machines as soon as they join a network.

5.4.1 boot up event

When the application is started, it checks the local chain of blocks, if available, and all transactions for validity. Then the node up event should be run to ensure that no blocks have been missed since the last boot. Should and error be encountered it sets aside the local data and attempts a recovery by downloading the current time line from the network.

5.4.2 node up event

The node up event is a basic building element. This event is called by a node itself once the network connection is up and running. It announces itself to all trackers that are in the meta data file or have been collected by other peers. This makes sure the trackers have up-to-date information about all peers in the network.

Other peers should be asked if they might have new blocks that have been missed while the network was down. This is done by asking for a list of all blocks a client possesses. This block list is checked against the local block list and missing blocks are requested as necessary from the peers. This routine should be executed every time the machine experiences network downtime. This ensures that the machine has the latest block chain elements.

5.4.3 node up timer event

This is a regular interval event at which the node up routine should be called. Depending on the number of updates in a network, this routine might be called from every hour to once a day.

5.4.4 tracker timer event

The routine announces to the tracker that this machine is still alive. To keep the tracker up to date it should be called every 15 to 30 minutes.

5.4.5 receive block event

Is triggered when a block is received the block itself and all transactions in the block are validated. When a block is valid, it is broadcasted to all known peers. This kind of flooding algorithm makes sure that every machine in the network receives a newly found block as soon as possible. If something is wrong with the block, it is dropped and not broadcasted to avoid further distribution of the block.

5.4.6 asked for block event

The client has to react to this event when it was asked by another peer if it possess a certain block. With the hash as unique identification mechanism the client is able to look up in its local data if there is a block with the certain identification. If the client possesses the block it returns it to the sender otherwise it answers with a 404 file not found header message.

5.4.7 asked for block list event

The client has to react to this event when it was asked by another peer for all blocks it currently has in its local database related to the Archive Network. If the client is new to the network and does not possess any of the blocks yet it answers with a 404 file not found header message. If the client possess block related to the Archive Network it simply returns those.

5.5 Limitations

The protocol using the chain of blocks algorithm at its heart has some limitations that we want to discuss in this chapter. Lin and Liao (2017) conducted a survey of describing attack vectors on a block chain with specific focus on the bitcoin block chain.

5.5.1 Real-Time-Updates

Due to the asynchronous nature of allowing contributions from every machine, the block chain is not a real time data structure. It takes time for a block to travel throughout the network. The data structure is eventual consistent, meaning that at some point in time the whole network converges with the same state of the database. The time it takes to achieve this point is determined by many factors including the network latency, number of machines in the archive network and available bandwidth.

For data that is expected to change very infrequently the limitation of slow updates is not a problem. Most static data does not have the requirements to be available at another machine for quite some time. Since an archiving network is built for sharing long-term data, it might be sufficient when the block chain is updated only once a day, to pack as many transactions in a block as possible.

This makes it possible to push the CPU consuming task of solving the puzzle in a time slot where the application would be idle otherwise, like midnight.

5.5.2 Truncation Attack

As described by Ma and Tsudik (2009), the block chain as a log file, is vulnerable to a truncation or tail cut attack. In this attack the last n blocks are omitted when a client asks for the complete block chain.

Since a malicious attacker would need to control all clients surrounding a victim, the attack has only a small possibility of success in a large network. As long as one client's relays all blocks the attack would be mitigated.

5.5.3 The Sybil Attack

Like all distributed systems the network archives are subject to the Sybil attack described by Douceur (2002). In a system with an open account creation mechanism like generating PGP keys nothing prevents a user to generate a new pair of keys for every transaction and thus gain new identities with every key pair. Depending on the mode the application is running this might become a problem when contributions from all machines are allowed. A malicious user could not get banned by trying to identify him through their key, since he could change the keys for every block.

5.5.4 Block flooding

A malicious user is able to generate fake blocks (Chun, 2007) and occupy CPU time on a machine, since all incoming blocks need to be validated. To prevent this attack, misbehaviour of peers regarding bandwidth consumption could be taken into account which might result in dropping the peer.

5.6 Summary

This chapter has described the Archive Network model's architectural design, which enables it to be integrated into centralised open-data projects without breaking the existing architecture. Specifically, it has presented the design of the Global Timeline, a discovery system for long-term access, and the Distributed Content. The first section gave an introduction to change-sets and how they form the basis of a chain of blocks. The section closes by explaining the many blocks form a global timeline which can be validated by cryptographic hashes. Section 2 details the discovery mechanisms the protocol provides to gain access and identify a single Archive Network. While section 3 highlights the network behaviour that all participants adhere to in order to cope with fluctuating members in the network.

Specifically, the model:

1. provides a solution to verification of objects that have been hosted at different machines through the global timeline

2. introduced formalised access and discovery of decentralised storage locations

3. enables contributions to a decentralised dataset through synchronisation orchestrating

The global timeline is the key to the implementation of the model, as it is the novel mediator between the distributed databases of the machines, ensuring that all systems are able to fully synchronise without requiring a central machine. Because it is based on the Blockchain algorithm, which has been adopted and deployed by cryptocurrency such as bitcoin, it is mature and stable enough for other implementations, and the variant defined for the Archive Network is just as stable as only a subset of the full algorithm is used. The following chapter presents a prototype of the Archive Network that has been tested and measured to further validate its design.

6 Proof of Concept by Research Prototype

This chapter proposes a decentralized registry. It is open for external contributions by design and has no single point of failure. All participants together build up a single global collection of Data Providers. New Data Providers only have to register with a single member and the entry is distributed to all participating Service Providers. Leveraging a single global collection allows Service Providers to refocus efforts on their value-added service for the community and allowing Data Providers to reach a large number of Service Providers with a single entry.

6.1 OAI-PMH

With the introduction of the Open Archives Protocol for Metadata Harvesting (OAI-PMH) automated harvesting of metadata from archives became possible (Rusch-Feja, 2002). This allowed so called service providers to build search engines around the data (McCown et al., 2006). A single query allows users to search the catalogue of different archives at once. The catalogues can be synchronized efficiently with the updates that happen at source in the catalogue once the catalogue is found by a service provider.

The Open Archives Protocol for Metadata Harvesting has been widely adopted as an approach to allow harvesting of metadata from archives. To determine how much of the current OAI-PMH corpus search engines index, McCown et al. (2006) harvested nearly 10M records from 776 OAI-PMH repositories. From these records they extracted 3.3M unique resource identifiers and then conducted searches on samples from this collection. Of this OAI-PMH corpus, Yahoo indexed 65%, followed by Google (44%) and MSN (7%). Automated discovery of Providers is not part of the protocol. Service Providers have the additional burden of searching the web for new Data Providers. This leads to duplicate effort since every Service Provider maintains a private collection of Data Providers.

OAI-PMH is a distributed protocol at the core that allows every service provider to connect with every data provider. A global access point for registration is not defined by the protocol. With many service providers this leads to a situation where a new data provider has to register its archive with every service provider in order to distribute his data. On the other side, service providers want to find as many data providers as possible since they want to provide a search over a complete body of knowledge.

The OAI hosts a centralized registry under their domain. With a pre-validation of a submitted URL only genuine archives are allowed into the registry. The registration process is an optional step and there is no way for service providers to automatically query the data via a defined interface. Since service providers want to spread their data as far as possible, they began to register with service providers directly and the registry under the OAI domain became one of many. Every service provider maintains their own list of archives and no formalized structure for exchange of this information exists as highlighted by (Shreeves et al., 2005).

The goal of the prototype is a novel approach to building a collection of information about OAI data providers. The approach does not need a centralized server to exchange data and is based on Archive Networks as defined by Goebert et al. (2013). Members connect into a distributed network of participants and exchange information with other members through a defined interface as it becomes available.

Like the OAI-PMH, the network is based on a distributed protocol at the core to avoid single point of failures and become resilient against member fluctuation. Archive Networks are best suited for building a collection with many participants but not rely on a centralized infrastructure for synchronization.

6.2 Prototype Concept

The protocol allows everybody to participate in a network but Service Provider will have a special interest in running a node that collects URLs from Data Providers as they can be seen as interface for the end user to the data. Service Providers are rather stable services with a stable number of providers who take great effort in order to be discovered.

The Service Provider runs software that implements the handling of the Archive Network Protocol (ANP) independently from the existing search engine infrastructure. The protocol implementation provides a REST (Fielding, 2000) programming interface in order to query data. The integration between these two services has to be done by the service provider.

In order to connect data providers a key has to be provided. The key for discovery in Archive Networks is a meta data file that contains information about how to find other interested nodes. Many Archive Networks exists for different collections but an Archive Network is always scoped to the usage of the same key. This allows several installations who serve different collections to coexist next to each other on one installation. In this paper we will focus only on the usage as a registry for the OAI-PMH protocol.

The key contains the URL to a tracker service (Cohen, 2003). The tracker service maintains information about other machines that registered themselves for the same key. The format of the key file is compatible with the already existing bittorrent infrastructure (Cohen, 2003). The tracker service returns IP addresses of machines who participate in the same network.

The key is brought into the Archive Network Protocol implementation and used to contact the tracker. After receiving information about other members of the network, the implementation starts to contact those in order to validate that they are really members of the network with the interested of building a collection.

For Service Providers the local registry provided by the Archive Network Protocol implementation is not different to the one found on the OAI domain. A simple HTML form provides an interested user with the ability to enter the URL to his service. The web service run by the service provider that receives the request for inclusion onto the list of known repositories, validates the URL by calling it and making sure that the protocol is honoured by the service behind the URL. This avoids including a repository which is not behaving correctly defined by protocol or including a wrongfully entered URL into the list of know repositories.

So far, the data provider is only stored locally. In order to share it with the other members of the network it is send to the Archive Network Protocol implementation. The implementation tests the URL again and after successful validation wraps the URL in a transaction. A transaction is an xml structure that provides information about the URL. Transactions have to be wrapped in an xml structure called a block in order to be send to other members. A block can contain more than on transactions and a link to the previous block to build a chain.

The block is sent to all connected machines who verify the information in the block and accept it as an advancement of the local timeline. The block is then sent on until every member of the network has received the block. The usage of a cryptographic puzzle to seal a block was described by Nakamoto (Nakamoto, 2008). The solution to this cryptographic puzzle has to be included in a block in order to make sure the network is not flooded by blocks and the order of the timeline becomes a matter of block creation time. The computational puzzle takes time to be solved depending on the difficulty level chosen and has to be recalculated for every block.

New members of the network download all blocks in order to sync with the network. To validate that the blocks have not been tempered with the key to the network also contains the first block. The second block in the network has to link to the first block in the key. Subsequent blocks have to link the previous block. This way a new member can verify the chain of blocks he received from the network back until the first block. If the chain of blocks is valid until the first block contained in the key the timeline is valid and has not been tempered with.

Periodically the Service Providers should query the Archive Network Protocol implementation if new change sets have arrived. If yes, the service validates a new URL locally and if successful adds it to the local database in order to prepare for meta data harvesting of the newly found provider.

A reference implementation can be found at: [1]

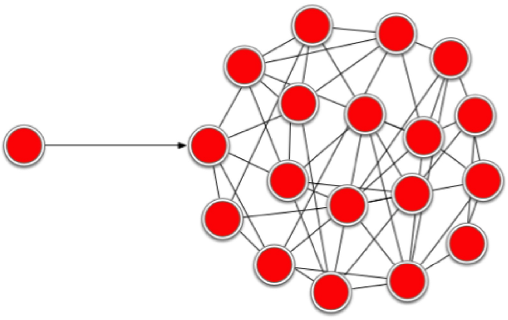

Figure 6.1: Distribution of data in an Archive Network

[1] https://github.com/bigcurl/oai-registry-distributed

6.3 Integration

This chapter describes the communication between the existing search engine infrastructure and the protocol implementation. A node running the Archive Network Protocol implementation is totally independent from the system that wants to harvest information. The application can talk to the protocol implementation via a REST interface. The protocol itself is based on the friend's container described by Lagoze and Van de Sompel (2003). The node is running a webserver internally that supports the following commands:

- GET /urls
- POST /urls

6.3.1 GET /urls

Function: Returns the entries of all URLs in order of appearance in the friend's container format specified by the OAI-PMH.

Return Status:

- 200: OK
- 400: The request could not be understood by the server due to malformed syntax

- 500: Internal server error

Return values if return status 200:

- id: A unique alphanumeric value for the URL.

- baseURL: A URL to an OAI repository.

Listing 6.1: Example of a BaseURL segment

```
<BaseURLs>
 <baseURL id='234987234...'>
   http://url.to.repository.com
 </baseURL>
</BaseURLs>
```

6.3.2 POST /urls

Function: Sends a candidate URL for inclusion into the chain of blocks

Post Parameters:

- baseURL: The URL to a repository

Return Status:

- 200: OK

- 400: The request could not be understood by the server due to malformed syntax
- 409: Conflict. Resource already exists.
- 500: Internal server error

Example:

- baseURL=http://url.to.repository.com

Integration: The protocol format internal to the Archive Network must be tailored to situation of handling URLs. We adapt the sections in the following way. A transaction segment contains a single baseURL and additional metadata.

Listing 6.2: Example transaction segment with BaseURL and meta data

```
<transaction>
 <tx-hash>390d0002...</tx-hash>
 <created-at>2014.05.29T09:30:10Z</created-at>
 <baseURL>http://url.to.repository.com</baseURL>
</transaction>
```

- tx-hash: is a SHA512 hash taken from the URL contained in link.
- creation-at: Date when the URL was created in ISO:8601 format
- baseURL: A URL to an OAI repository

6.3.3 Block

A block contains one or more transactions and provides the fields to validate it later.

Listing 6.3: Example block segments containing multiple transaction segments

```
<block>
 <previous_hash>0023987...</previous_hash>
 <hash>0002390d . . .</hash>
 <transactionshash>3246234...</transactionshash>
 <nonce>439</nonce>
 <transactions>
 <transaction>
  ...
 </transaction>
 <transaction>
  ...
 </transaction>
 </transactions>
</block>
```

- previoushash: The hash of the previous block this block wants to be linked to

- hash: A hash of the current block. SHA512(previoushash + transactionshash + nonce)

- transactions hash: A hash taken from the content of the transaction's fields.

SHA512(<transaction >... <transaction >+ ...)

- nonce: The proof of work (Nakamoto, 2008).

6.4 Discussion of Prototype

This section discusses the experience of running the software on a local network with simulated participants. . In order to determine what was measured the simulation was split into the component's hardware, software, and network. For the hardware part the CPU Utilisation and RAM Utilisation was logged. For the software part the time to solve the puzzle was logged, the time it took until a block was validated, and the time it took from sending a block until all nodes confirmed the block was taken. For the network part the utilization of the port was measured.

6.4.1 Client Setup

A total of 8 clients had been involved in the experiment. All machines had the same hardware characteristics stemming from the Shiva Plug. A plug computer consisting of 1.2 GHz ARM Marvell Kirkwood 88F6281 processor including 100MB Ethernet Card and 20GB of internal storage. All machines were running the Debian Linux operating system and where setup with the Archive Network Software.

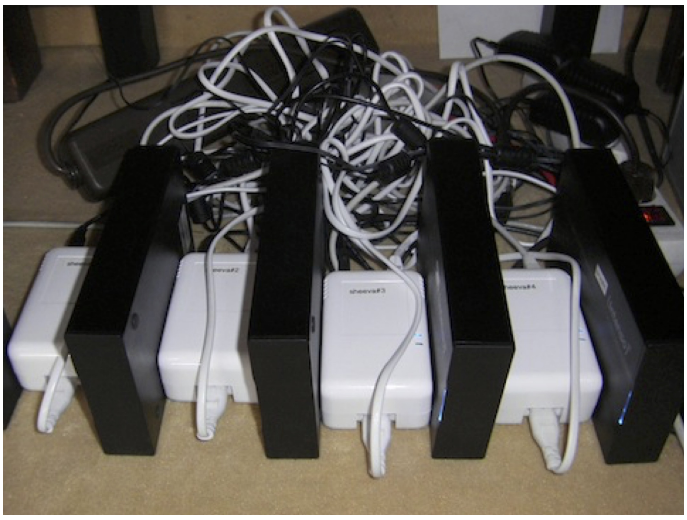

Figure 6.2: Set of Sheeva Plugs with Harddrive and Network connections

6.4.2 Network Setup

All machines were linked on a local network with a switch providing a capacity of 100MB per port. The machines have not been connected to the internet and have not experienced delays resulting from intermediate hardware needed to connect the machines. No firewalls or similar protection mechanism have been configured for the simulation.

6.4.3 Protocol Settings

The system was tested with a level of 6 for the cryptographic puzzle. Level 6 indicates that a hash having a 0 for the first 6 positions has to be found. It took around 5 seconds on an average to solve the puzzle and find the correct nonce. The timestamps have been taken on the client using the logging facilities of the programming language ruby (in version 2.5.0) and were written to a log file and later analysed.

6.4.4 Simulation Setup

In a true peer-to-peer fashion all machines were setup to be clients and servers. The clients were either using the API to ask for new additions to the library using an internal timer which was set to ask every 3 seconds or sent randomly generated urls to the API to keep a pipeline of new blocks and simulating users interacting with the system. All machines were connected via Ethernet through an 100MB Switch. They simulated a single archive network in order to stay in sync and were configured to generate a new block in a random time interval between 1 and 60 seconds. Also, to simulate fluctuation in the network nodes killed them self in a randomly selected time interval between 1 minute and 15 minutes resulting in a redownload all blocks including validating the chain on this node.

On an operating system level the CPU level was measured with parsing the output of the command line tool top every 4 seconds, written to a log file on disk and which was later analysed.

6.4.5 Simulation

The prototype ran as planned without hitting a major bottleneck in terms of network or processor capacity. What became apparent was that no requeuing of transactions was scheduled if a block did not end up in the mainline branch which resulted in data loss.

Also, the storage size played an important role as in this setup the storage capacity was fixed which could result in full disk if not paid attention to.

The network traffic coming from the protocol was not able to saturate the 100MB connection for the nodes. While it shows that optimization on this level is possible it might take a long time until the effect of the optimization is recognized in a setup. A script was running on each machine to check the processing capacity used and if a certain threshold (25% CPU capacity) was reached it would reboot in order to recover and simulate additional fluctuation in the network.

6.5 Improvements

Depending on the use case at hand there are many opportunities in the prototype to improve the overall performance. From optimizing the programming language from faster development time to quicker execution time to optimize the wire protocol from a text-based format to a binary format to save bits when sending data over the network. This section focuses on three items that have been raised during feedback that was given as result of the presentation of the research at the iPres conference in Berlin.

6.5.1 Build Robust Community

The prototype was only tested under lab conditions and not on the internet with different entities involved. The next step is to create a community around the software and build the necessary artefacts that enable other people to install and run the software.

Installation documentation, network setup guides and similar help has to be provided in order to give the software to interested people.

Every member has a full copy of the data and acts as an entry point into the network balancing the capacities needed to support an archive network. The more members the network has the more resilient it becomes. The ability to run a node in the network is not limited to service providers. Data Providers and even end users can run a node in the system and strengthen the data set against member fluctuation. Recruiting work has to be done to encourage people to run a node themselves.

6.5.2 Extending the encapsulated OAI-PHM data Format

The format that the prototype encapsulates is rather limited. It provides only access to the information that is contained in the friend container description of the OAI-PHM format. The friend container format is flat and only contains a link to a repository. The OAI-PHM has more information to offer and an assessment which other important fields that contain more information about the repository itself should be moved into the Archive-Network without bloating the protocol.

6.5.3 Multi Archive-Network Support

The prototype has been implemented as a proof-of-concept to validate the model and design decisions. It supports only a single Archive-Network at a time. This is rather wistful in terms of computing resources. The support of multiple Archive-Networks running within a single program mitigates this effect drastically and enables a single setup to contribute to multiple collections without an overhead in administration effort.

6.6 Summary

This chapter concludes the work that has been performed in order to validate the concepts that have been defined as part of the thesis. The chapter has presented the design and specification of the Open Archives Protocol for Metadata Harvesting on an Archive Network, as well as a prototype implementation that demonstrates the power and flexibility of the system. In addition, the chapter has proven the scalability of the design, which has been tested with performance data from the prototype. More specifically, the chapter has presented:

- the design and specification of the Archive Network Protocol
- the design and specification of the adaption of the protocol to host the Open Archives Protocol for Metadata Harvesting
- a fully working prototype of the Open Archives Protocol for Metadata Harvesting on an Archive Network;
- demonstration applications of the Archive Network
- performance figures of the system.

In addition, further validation of the system's design has been provided by the publication of a paper that describes the Open Archives Protocol for Metadata Harvesting on an Archive Network at iPres 2014 (Goebert, Harriehausen-Mühlbauer and Furnell, 2014).

7 Conclusions and Directions of Further Research

This chapter concludes the thesis by summarising the work that has been achieved, including a new Archive-Network protocol for decentralised hosting of digital collections, the global timeline of Archival Information Packages as a fundamental entity for synchronisation and its associated algorithms and protocols, a formalised standard for discovery and access to digital collections, and the implementation of a prototype for the Open Archives Initiative Protocol for the Metadata Harvesting Registry dataset. The chapter concludes by discussing limitations of the research, and describing new areas of research that can be performed to further enhance and refine the work.

7.1 Achievements of the research

The research programme has met all of the objectives originally specified in chapter 1. New conceptual and practical work has been presented in a number of areas, as listed below:

- The development of the new Archive-Network protocol allowing decentral hosting of digital collections, which solves the three identified weaknesses of central preservation systems. The Archive-Network protocol comprises the meta-data file containing the genesis hash, the specification of Archival Information Packages suitable for distribution, and the global timeline of Archival Information Packages. The protocol is backwards compatible with existing web application architecture and enables the storage of digital collections at unknown storage locations while retaining data authenticity.

- The design, development and testing of a meta-data file format. The file format is the access key that identifies the dataset and connects storage locations to a specific Archive-Network. The meta-data file builds the foundation of the discovery mechanism for the Archive-Network protocol leveraging existing infrastructure provided by the Bittorrent community.

- The design, specification and development of Archival Information Packages modified to suit the domain of decentralised long-term preservation. With knowledge of the genesis hash contained in the meta-data file, the packages are inter-linked providing a tamper-proof, cryptographically verifiable timeline of events.

- The design, specification and development of the peer-to-peer protocol that provides accessibility to exchange the Archival Information Packages between the storage locations, complete with the implementation of a prototype to test the concept and measure its performance. It provides the basis for permanent accessibility as long as one storage location remains in the peer-to-peer network.

Several papers relating to the research programme have been presented at refereed conferences, where the work received praise and recognition for its novel approach to decentralised preservation. In conclusion, it is believed that the research has made valid and useful contributions to the fields of long-term preservation, decentralised systems and open-data.

7.2 Limitations of the research

Despite having met the overall objectives of the research programme, and despite the functional success of the prototype, the work inevitably has a number of limitations. The principal points are presented below.

- Insufficient time and resources were available to further develop the different acceptance-modes responsible for the acceptance of a foreign Archival Information Package by a storage location. The scope of acceptance-modes could easily form another research programme in itself, including experiments, implementation and further design. However, the design and implementation of the anonymous acceptance-mode was the priority in order to test one possible configuration of the Archive-Network protocol.

- The prototype was deployed and tested on rather underpowered computers, and on a LAN rather than across the Internet, due to resource constraints. As such, the prototype was not run under real-world conditions, which will inevitably have had an impact on the performance figures. Despite this, the performance and scalability proved sufficient to validate the feasibility of the architecture and provides a starting point to scale to a system the size of common open-data projects like Wikipedia or Open Street Map in future iterations.

- Although important for every application that is connected to the Internet, the large topic of application and protocol security was only considered to a point until working results were achieved . It was explained why certain cryptograph parameters were chosen, but application security best practices, e.g. OWASP (OWASP, 2010) were not taken into account during this work. While security should be built in and not bolted on, it was a top priority to show the feasibility of a protocol that synchronises data items among unknown storage locations. When necessary the work relies on well-known algorithms which might provide attack vectors in themselves. As security is an evolving topic and related measures need to be specifically suited to the given problem, this work assumes no malicious user in this iteration of the concept. This assumption provides a platform for future work in the area of decentralised preservation to address the unique combination of algorithms that are used.

7.3 Suggestions and scope for future research

Throughout the thesis, areas where future work is possible or desirable have been identified. These areas could be used as a reference to build upon and enhance the stable infrastructure for decentralised preservation. These areas, together with others, are summarised below.

- While the tracker infrastructure used to discover storage locations in a certain Archive-Network still exists, the general usage of this method of discovery has been abandoned. While for the experiment it was faster to replicate the deprecated method, relying on a tracker to store knowledge about storage locations, an alternative mechanism can be used to provide anonymity and reduce the dependency of available trackers on the Internet, with the sole purpose of providing discovery to clients. So-called magnet links, e.g. [1] provide links into an overlay network where the meta-data file is retrieved from other storage locations. This network needs only a small number of bootstrap machines to provide a stable mechanism of discovery. Depending on the number of active storage locations, bootstrap servers might not be needed at all.

- The meta-data file used cryptography that provides a surface for attack vectors. Depending on the goal of the attack different scenarios can be imagined, providing further platforms for further research. A predictable hash collision for example would enable a malicious user to exchange data that has already been accepted by some storage locations with data that might contain malicious content or false information, depending on what is shared in the Archive-Network. Establishing

[1] magnet:?xt=urn:btih:5ec62ede21a12d271da2e99afe1df2eb2794a87e

more sensible defaults with certain attack vectors in mind could future-proof the system.

- The XML format used in the peer-to-peer protocol to structure information could be reimagined using a different, more efficient format, for example JSON, to foster universal adoption in places where a format that is not natively supported adds unnecessary computation time. The JSON format encoding has its roots in the programming language JavaScript, which brings dynamic elements to the browser. Also, on the server side, with web frameworks like node.js and the npm ecosystem, JavaScript is a dominant player. JSON itself is also natively supported in many database management systems like MySQL, PostgreSQL and Oracle.

- While xml provides a readable format which is widely supported, it is not the most efficient way of encoding information in a readable way. The wire protocol could be sent as binary data. Bencode is a data serialisation format transforming native data types into binary code. Supported are byte strings, integers, lists and dictionaries. To adhere to this small list of natively supported data types the Archive-Network protocol would need adaption in its current form.

- Depending on the content that is shared in an Archive-Network, it might be suitable to have true anonymous contributions or participants. Naturally, in a peer-to-peer network where data is exchanged with other participants, a network address of some sort has to be discovered where an interested peer can be reached. In the case of the current state of the protocol this is the IP address. The IP address can be used to gain the identity of a participant. The Tor project defends against traffic analysis, a form of network surveillance that threatens personal freedom and privacy, confidential business activities and relationships, and state security (Dingledine, Mathewson and Syverson, 2004). The combination of an Archive-Network with mechanisms to collaborate without revealing personal information about participants is an interesting area of future research.

- A central component of the timeline algorithm is the concept of using a proof-of-work algorithm to decelerate the speed of contributions to an Archive-Network. Depending on the purpose of the Archive-Network, it might be preferable to use a different mechanism than the one suggested in the current implementation. Researching different methods for the proof-of-work algorithm, and controlling the amount of work that is added per Archival Information Package, is a wide field.

- Depending on the algorithm used for proof of work it is possible to change the enforced policy of acceptance for contributions. Four so-called acceptance-modes have been identified that control the level of outside contribution accepted by an Archive-Network. The first acceptance-mode allows contributions that have been signed by a certain root certificate. The root certificate is embedded into the genesis hash. The second acceptance-mode allows contributions from certificates that have been signed by the root certificate. As in the first acceptance-mode the root certificate is embedded into the genesis hash value. The third acceptance-modes allows blessed certificates to bless other certificates. The fourth acceptance-mode does not rely on a root certificate to restrict contributions and only requires the proof of work to contribute to the Archive-Network. While the other acceptance-modes also fulfil certain purposes, this work only explores the third acceptance-mode that allows contributions from everybody. Only the last acceptance-mode was discussed in detail while the other modes still have their use-case in the long-term preservation field.

- In the current form the protocol has no way of updating its validating algorithms once the first Archival Information Package has been created. If one of the algorithms that constitute the protocol is deemed unsafe because of a new mathematical method, the timeline that constitutes an Archive-Network needs to be recalculated from scratch using a stronger algorithm. The protocol needs an extension that allows different algorithms with different output sizes to coexist in the linked timeline of Archival Information Packages. This would provide a way to mitigate the problem of the growing number of attacks against cryptographic algorithms and growing computing performance.

- While the protocol to exchange messages between storage locations is formalised, there is no formalisation of acceptance rules to validate the content of the packages. A limitation of the current state of the Archive-Networks protocol is that for every special purpose archive a specific client adhering to all the validation rules of the schema format has to be coded. A book collection schema requires a field that represents an International Standard Book Number (ISBN); it is part of the client code to make sure that this field is validated according to the specification of an ISBN. This limitation could be lifted if the validation were also embedded in the meta-data file, paving the way to a single universal client that is able to differentiate Archive-Networks without needing a separate client for each network.

- Despite its importance, no particular attention has been given to the selection of sensible defaults for the hashing algorithms used in the Archive-Network protocol. Determining which length of hash is sufficient given the current state of computing power and mathematical knowledge is a constant process that needs to be part of the preservation lifecycle. With a weakened hash algorithm multiple attacks become possible, undermining trust in the blockchain.

7.4 The future of decentralised preservation

Decentralised preservation is here to stay, and will remain part of best practices for digital long-term preservation for many years to come. Since budgets are limited, novel ways of finding storage space have to be acquired. Safely storing and validating data without the need to own and control the storage location enables this. As such, the Archive-Network protocol provides a solution to the challenges with which the thesis is concerned. However, even if it does not become a new part of digital collection projects, it has contributed to the field in many ways:

- The Archive-Network protocol is a novel means of decentralised preservation. Its collaborative collaboration treatment of digital collections in a network of storage locations differs substantially from existing models, enabling preservation to easily address the problems that emerge from the nature of digital archiving. As such, it can be used in a wide range of applications, from enabling entities that are unknown to each other to build a collection, to enabling website providers to build a website that cannot be shut down or provide the foundation for a new version of the bittorrent protocol that enable continuous collections of torrent files instead of isolated files.

- Decentralising collaboration through the Archive-Network protocol is a novel approach to let the public participate in the creation of a digital collection. Individuals who deem a collection worthy can focus their efforts into a single project, instead of having many small groups that all want to achieve the same goal.

- The Archive-Network protocol offers a novel approach for memory organisations to utilise public resources without sacrificing authenticity because storage locations do not belong directly to the organisation. Outsourcing the storage and administration of a digital collection, even on a read-only basis, is currently not an option for memory organisations.

- The Archive-Network protocol offers a novel approach to preserve achievements of communities by allowing storage locations to become a vital part of the hosting infrastructure, not only storing snapshots of the data. The event-sourcing approach allows storing of individual data changes and rebuilding of the dataset by re-running the events from the event log on an empty application.

- The Archive-Network protocol features a novel approach in preservation to create a public readable order of events, which is not coordinated by a single source. Current approaches concentrate on backing up data from a single storage location; however, with the new approach it is possible to read the data from multiple sources at the same time and still have the guaranteed order of events. This is especially helpful when providing enough bandwidth is an issue or availability of a single system is limited.

- By introducing different acceptance-modes the Archive-Network protocol offers a way to restrict the number of contributors, in extreme cases to only one contributor.

This change was not possible previously, as the only possible acceptance-mode allowed anybody to contribute.

- The Archive-Network protocol offers a novel reuse of the existing tracker infrastructure provided by the Bittorrent community. Backwards-compatibility with the Bittorrent file format allows reliance on a proven infrastructure for discovery of storage locations that want to preserve a certain dataset.

As such, the research work that has been completed for this PhD has contributed to many fields, and has provided new avenues for future research that will provide many more contributions in the future.

References

3sat Nano Redaktion (n.d.), 'Ein einheitlicher standard für die flut digitaler daten'.
URL: *http://www.3sat.de/page/?source=/nano/bstuecke/119799/index.html*

Adam, S. (2010), 'Preserving authenticity in the digital age', *Library Hi Tech* **28**(4), 595–604.

Akrimullah, A., Sima, B. S. & Akhadi, Y. (2018), 'Openstreetmap infrastructure mapping and its usage on flood impact assessment using inasafe in surabaya', *IPTEK Journal of Proceedings Series* (1), 9–1.

ALLEA (2012), 'Open science for the 21st century: Declaration of all european academies'.
URL: *http://www.allea.org/Content/ALLEA/*

Anderson-Tarver, C. (2015), 'Crisis mapping the 2010 earthquake in openstreetmap haiti'.

Andrews, G. R. (1999), *Foundations of parallel and distributed programming*, Addison-Wesley Longman Publishing Co., Inc.

Arunkumar, K. & Devendran, A. (2019), Digital data preservation—a viable solution, *in* 'Data Management, Analytics and Innovation', Springer, pp. 129–141.

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R. & Ives, Z. (2007), Dbpedia: A nucleus for a web of open data, *in* 'Proceedings of the 6th International The Semantic Web and 2Nd Asian Conference on Asian Semantic Web Conference', ISWC'07/ASWC'07, Springer-Verlag, Berlin, Heidelberg, pp. 722–735.

URL: *http://dl.acm.org/citation.cfm?id=1785162.1785216*

Ballatore, A. (2014), 'Defacing the map: Cartographic vandalism in the digital commons', *The Cartographic Journal* **51**(3), 214–224.

URL: *http://dx.doi.org/10.1179/1743277414Y.0000000085*

Bandeira, L. A. M. (2019), Crimea back to russia and economic sanctions against russia, *in* 'The World Disorder', Springer, pp. 211–221.

Bandulet, C. (2011), 'The file systems evolution'.

URL: *http://www.snia.org/sites/default/education/tutorials/2011/spring/file/Christian_Bandulet_The_File_Systems_Evolution_final.pdf*

Beaver, D., Kumar, S., Li, H. C., Sobel, J., Vajgel, P. et al. (2010), Finding a needle in haystack: Facebook's photo storage., *in* 'OSDI', Vol. 10, pp. 1–8.

Berger, S. (n.d.), ‘Client-/server-architektur und internet’.
URL: *http://www.server-client.com*

Bergsma, M. (2007), ‘Wikimedia architecture’, *Wikimedia Foundation Inc* .

Berners-Lee, T., Fielding, R. & Frystyk, H. (1996), ‘Hypertext transfer protocol–http/1.0’.

Berners-Lee, T. & Fischetti, M. (1999), ‘Weaving the web. 1999’, *New York: Orion Business* .

Bertoni, G., Daemen, J., Peeters, M. & Van Assche, G. (2013), Keccak, *in* ‘Annual international conference on the theory and applications of cryptographic techniques’, Springer, pp. 313–314.

Bizer, C., Heath, T. & Berners-Lee, T. (2011), Linked data: The story so far, *in* ‘Semantic services, interoperability and web applications: emerging concepts’, IGI Global, pp. 205–227.

Bizer, C., Heath, T., Idehen, K. & Berners-Lee, T. (2008), Linked data on the web (ldow2008), *in* ‘Proceedings of the 17th International Conference on World Wide Web’, WWW ’08, ACM, New York, NY, USA, pp. 1265–1266.
URL: *http://doi.acm.org/10.1145/1367497.1367760*

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R. & Hellmann, S.

(2009), ‘Dbpedia-a crystallization point for the web of data’, *Web Semantics: science, services and agents on the world wide web* **7**(3), 154–165.

Böhm, C., Naumann, F., Freitag, M., George, S., Höfler, N., Köppelmann, M., Lehmann, C., Mascher, A. & Schmidt, T. (2010), Linking open government data: What journalists wish they had known, *in* ‘Proceedings of the 6th International Conference on Semantic Systems’, I-SEMANTICS ’10, ACM, New York, NY, USA, pp. 34:1–34:4.
URL: *http://doi.acm.org/10.1145/1839707.1839751*

Brabham, D. C. (2012), ‘The myth of amateur crowds’, *Information, Communication & Society* **15**(3), 394–410.
URL: *http://dx.doi.org/10.1080/1369118X.2011.641991*

Bray, T. (2014), ‘The javascript object notation (JSON) data interchange format’, *RFC* **7158**, 1–16.
URL: *https://doi.org/10.17487/RFC7158*

Bray, T., Paoli, J. & Sperberg-McQueen, C. M. (1997), ‘Extensible markup language’, *World Wide Web J.* **2**(4), 29–66.
URL: *http://dl.acm.org/citation.cfm?id=274784.273625*

Brin, S. & Page, L. (1998), ‘The anatomy of a large-scale hypertextual web search engine’, *Computer networks and ISDN systems* **30**(1), 107–117.

Bundesarchivgesetz (n.d.), ‘Bundesarchivgesetz’.

URL: *https://www.bundesarchiv.de/bundesarchiv/rechtsgrundlagen/bundesarchivgesetz/index.html.de*

Caplan, P., Kehoe, W. & Pawletko, J. (2010), Towards interoperable preservation repositories (tipr), *in* 'Proceedings of the 2010 Roadmap for Digital Preservation Interoperability Framework Workshop', ACM, p. 16.

Ccsds (2002), Reference Model for an Open Archival Information System (OAIS). Blue book, Technical Report 1.
URL: *http://public.ccsds.org/publications/archive/650x0b1.pdf*

Choy, S. C., Crofts, N., Fisher, R., Choh, N. L., Nickel, S., Oury, C. & Ślaska, K. (2016), 'The unesco/persist guidelines for the selection of digital heritage for long-term preservation'.

Chun, B.-G. (2007), Mechanisms to tolerate misbehavior in replicated systems, PhD thesis, Citeseer.

Cohen, B. (2003), Incentives build robustness in bittorrent, *in* 'Workshop on Economics of Peer-to-Peer systems', Vol. 6, pp. 68–72.

Cohen, N. (2012), 'Travel site built on wiki ethos now bedevils its owner'.
URL: *https://www.nytimes.com/2012/09/10/business/media/once-a-profit-dream-wikitravel-now-bedevils-owner.html*

Crowley, J. (2013), 'Connecting grassroots and government for disaster response'.

Culham, P. (1989), 'Archives and alternatives in republican rome', *Classical Philology* pp. 100–115.

Davies, T. (2017), 'Open data barometer: 2017 global report', *World Wide Web Foundation and Open Data Institute* .

Davies, T. (2018), 'Open data barometer: 2018 global report - leadership edition', *World Wide Web Foundation and Open Data Institute* .

De Lusenet, Y. (2007), 'Tending the garden or harvesting the fields: Digital preservation and the unesco charter on the preservation of the digital heritage', *Library Trends* **56**(1), 164–182.

Dingledine, R., Mathewson, N. & Syverson, P. (2004), Tor: The second-generation onion router, *in* 'Proceedings of the 13th Conference on USENIX Security Symposium - Volume 13', SSYM'04, USENIX Association, Berkeley, CA, USA, pp. 21–21.
URL: *http://dl.acm.org/citation.cfm?id=1251375.1251396*

Douceur, J. R. (2002), The sybil attack, *in* 'Peer-to-peer Systems', Springer, pp. 251–260.

Duranti, L. (1995), 'Reliability and authenticity: the concepts and their implications', *Archivaria* **39**.

Engelfriet, A. (2010), 'Choosing an open source license', *IEEE software* **27**(1).

Englev, E. & Petersen, K. (2003), 'Ich-gcp guideline: quality assurance of clinical trials.

status and perspectives', *Ugeskrift for laeger* **165**(16), 1659–1662.

Erway, R. & Research, O. (2010), *Defining "born Digital": An Essay*, OCLC Research. **URL:** *https://books.google.de/books?id=xLSTAQAACAAJ*

Factor, M., Henis, E., Naor, D., Rabinovici-Cohen, S., Reshef, P., Ronen, S., Michetti, G. & Guercio, M. (2009), Authenticity and provenance in long term digital preservation: Modeling and implementation in preservation aware storage., *in* 'Workshop on the Theory and Practice of Provenance'.

Fiedrich, F. & Fathi, R. (2018), Humanitäre hilfe und konzepte der digitalen hilfeleistung, *in* 'Sicherheitskritische Mensch-Computer-Interaktion', Springer, pp. 509–528.

Fielding, R. T. (2000), Architectural styles and the design of network-based software architectures, PhD thesis, University of California, Irvine.

Foster, J. & Sheppard, J. (2016), *British Archives: A Guide to archive resources in the UK*, Springer.

Ghemawat, S., Gobioff, H. & Leung, S.-T. (2003), *The Google file system*, Vol. 37, ACM.

Giles, J. (2005), 'Internet encyclopaedias go head to head', *Nature* **438**(7070), 900–901. **URL:** *http://dx.doi.org/10.1038/438900a*

GNU General Public License, version 3 (2007), http://www.gnu.org/licenses/gpl.html. Last retrieved 2012-05-10.

Goebert, S., Harriehausen-Mühlbauer, B. & Furnell, S. (2014), Towards a unified oai-pmh registry, *in* ‘Archiving Conference’, Vol. 2014, Society for Imaging Science and Technology, pp. 97–100.

Goebert, S., Harriehausen-Mühlbauer, B., Wentzel, C. & Furnell, S. (2013), Decentralized hosting and preservation of open data, *in* ‘Archiving Conference’, Vol. 2013, Society for Imaging Science and Technology, pp. 264–269.

Graham, P. M. (2019), ‘Guest editorial: Reflections on the ethics of web archiving’, *Journal of Archival Organization* **0**(0), 1–8.
URL: *https://doi.org/10.1080/15332748.2018.1517589*

Handelsgesetzbuch (n.d.), ‘§ 257 hessisches archivgesetz (harchivg)’.
URL: *http://dejure.org/gesetze/HGB/239.html*

Haslhofer, B., Warner, S., Lagoze, C., Klein, M., Sanderson, R., Nelson, M. L. & Van de Sompel, H. (2013), Resourcesync: leveraging sitemaps for resource synchronization, *in* ‘Proceedings of the 22nd international conference on World Wide Web companion’, International World Wide Web Conferences Steering Committee, pp. 11–14.

Hendler, J., Holm, J., Musialek, C. & Thomas, G. (2012), ‘Us government linked open data: Semantic.data.gov’, *IEEE Intelligent Systems* **27**(3), 25–31.

Hofman, H. J. (2002), A global issue: preservation of digital objects, *in* ‘Proceedings International Conference on Conservation and Digital Preservation of Archives and

Records', pp. 59–76.
URL: *http://eprints.erpanet.org/38/*

Hopper, G. M. (1952), The education of a computer, *in* 'Proceedings of the 1952 ACM national meeting (Pittsburgh)', ACM, pp. 243–249.

Howe, J. (2008), *Crowdsourcing: Why the Power of the Crowd Is Driving the Future of Business*, 1 edn, Crown Publishing Group, New York, NY, USA.

Ilic, D., Erman, D., Popescu, A. & Nilsson, A. A. (2004), Traffic measurements of p2p systems, *in* 'Second Swedish National Computer Networking Workshop'.

Jaffe, E. & Kirkpatrick, S. (2009), Architecture of the internet archive, *in* 'Proceedings of SYSTOR 2009: The Israeli Experimental Systems Conference', ACM, p. 11.

Javanmardi, S., Ganjisaffar, Y., Grant, S. & Lopes, C. (2010), 'Scientific mashups: The issue of trust in the aggregation of web 2.0 content'.

Kalusopa, T. & Zulu, S. (2009), 'Digital heritage material preservation in botswana: problems and prospects', *Collection building* **28**(3), 98–107.

Kargl, F., Schlott, S. & Nagler-Ihlein, J. (2003), 'P2p-technologien'.
URL: *http://medien.informatik.uni-ulm.de/lehre/courses/ws0304/semp2p/SeminarP2P.pdf*

Kirchhoff, A. (2011), *eBooks: The Preservation Challenge,*, 1 edn.

Krabina, B., Prorok, T., Lutz, B. et al. (2012), 'Open government implementation model', *Implementation of open government. Vienna: KDZ, Centre for Public Administration Research and Office of the CIO, Chief Executive Office of the City of Vienna* .

Lagoze, C. & Van de Sompel, H. (2003), 'The making of the open archives initiative protocol for metadata harvesting', *Library hi tech* **21**(2), 118–128.

Lee, K.-H., Slattery, O., Lu, R., Tang, X. & McCrary, V. (2002), 'The state of the art and practice in digital preservation', *Journal of research of the National institute of standards and technology* **107**(1), 93.

Leskovec, J., Huttenlocher, D. P. & Kleinberg, J. M. (2010), Governance in social media: A case study of the wikipedia promotion process., *in* 'ICWSM', pp. 98–105.

Library, B. (n.d.), 'Digital preservation strategy'.
URL: *https://www.bl.uk*

Lin, I.-C. & Liao, T.-C. (2017), 'A survey of blockchain security issues and challenges.', *IJ Network Security* **19**(5), 653–659.

Lord, P., Macdonald, A., Lyon, L. & Giaretta, D. (2004), From data deluge to data curation, *in* 'Proceedings of the UK e-science All Hands meeting', Citeseer, pp. 371–375.

Ma, D. & Tsudik, G. (2009), 'A new approach to secure logging', *ACM Transactions on*

Storage (TOS) **5**(1), 2.

Maniatis, P., Roussopoulos, M., Giuli, T. J., Rosenthal, D. S. H. & Baker, M. (2005), 'The lockss peer-to-peer digital preservation system', *ACM Trans. Comput. Syst.* **23**(1), 2–50.
URL: *http://doi.acm.org/10.1145/1047915.1047917*

March, S. T. & Smith, G. F. (1995), 'Design and natural science research on information technology', *Decision support systems* **15**(4), 251–266.

Mauthner, N. S., Parry, O. & Backett-Milburn, K. (1998), 'The data are out there, or are they? implications for archiving and revisiting qualitative data', *Sociology* **32**(4), 733–745.

McCown, F., Liu, X., Nelson, M. L. & Zubair, M. (2006*a*), 'Search engine coverage of the oai-pmh corpus', *IEEE Internet Computing* **10**(2), 66–73.

McCown, F., Liu, X., Nelson, M. L. & Zubair, M. (2006*b*), 'Search engine coverage of the oai-pmh corpus', *Internet Computing, IEEE* **10**(2), 66–73.

Milligan, I. (2015), Finding community in the ruins of geocities: distantly reading a web archive, Institute of Electrical and Electronics Engineers.

Molloy, J. C. (2011), 'The open knowledge foundation: open data means better science', *PLoS biology* **9**(12), e1001195.

Nakamoto, S. (2008), 'Bitcoin: A peer-to-peer electronic cash system', *Consulted* **1**(2012), 28.

nestor Handbuch: Eine kleine Enzyklopädie der digitalen Langzeitarchivierung, Version 1.2 (2008), nestor - Kompetenznetzwerk Langzeitarchivierung und Langzeitverfügbarkeit digitaler Ressourcen für Deutschland.
URL: *http://nestor.sub.uni-goettingen.de/handbuch/nestor-handbuch.pdf*

Norheim-Hagtun, I. & Meier, P. (2010), 'Crowdsourcing for crisis mapping in haiti', *Innovations: Technology, Governance, Globalization* **5**(4), 81–89.

Oldenburg, S. (n.d.), 'Die vergänglichkeit von daten: Das lunar orbiter image recovery project'.
URL: *https://scilogs.spektrum.de/clear-skies/die-verg-nglichkeit*

Open archival information system (OAIS) Reference model (2012), Technical Report ISO 14721:2012, International Standards Organisation.
URL: *http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=57284*

Oram, A. (2003), 'Peer-to-peer: Harnessing the power of disruptive technologies', *SIGMOD Record* **32**(2), 57.

Oster, G., Molli, P., Dumitriu, S. & Mondéjar, R. (2009), Uniwiki: A collaborative p2p system for distributed wiki applications, *in* 'Enabling Technologies: Infrastructures for Collaborative Enterprises, 2009. WETICE'09. 18th IEEE International Workshops on', IEEE, pp. 87–92.

OWASP, T. (2010), 'Top 10-2013: The ten most critical web application security risks', *The Open Web Application Security* .

Palfrey, J. G. & Gasser, U. (2013), *Born digital: Understanding the first generation of digital natives*, Basic Books.

Panek, J. & Brychtova, A. (2015), Russian–ukrainian conflict over crimea on the map, *in* 'Proceedings of the2nd International Multidisciplinary Scientific Conference on Social Sciences and Arts SGEM2015, Albena', pp. 307–314.

Poiani, T. H., dos Santos Rocha, R., Degrossi, L. C. & de Albuquerque, J. P. (2016), Potential of collaborative mapping for disaster relief: A case study of openstreetmap in the nepal earthquake 2015, *in* 'System Sciences (HICSS), 2016 49th Hawaii International Conference on', IEEE, pp. 188–197.

Rahhal, C., Skaf-Molli, H., Molli, P. et al. (2008), 'Swooki: A peer-to-peer semantic wiki'.

Rothenberg, J. (2000), 'Preserving authentic digital information', *Authenticity in a digital environment* **5168**.

Rusch-Feja, D. (2002), 'The open archives initiative and the oai protocol for metadata harvesting: rapidly forming a new tier in the scholarly communication infrastructure', *learned Publishing* **15**(3), 179–186.

Schaffert, S. (2006), Ikewiki: A semantic wiki for collaborative knowledge management, *in* 'Enabling Technologies: Infrastructure for Collaborative Enterprises, 2006. WET-ICE'06. 15th IEEE International Workshops on', IEEE, pp. 388–396.

Schollmeier, R. (2001), A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications, *in* 'Peer-to-Peer Computing, IEEE International Conference on', IEEE Computer Society, pp. 0101–0101.

Schott, M., Dittmann, J., Vielhauer, C., Krätzer, C. & Lang, A. (2008), Integrity and authenticity for digital long-term preservation in irods grid infrastructure, *in* 'The 6th International Workshop for Technical, Economic and Legal Aspects of Business Models for Virtual Goods incorporating The 4th International ODRL Workshop', pp. 90–104.

Schrimpf, S. & Steinke, T. (2018), 'Long-term preservation policy of the german national library'.
URL: *https://d-nb.info/1163683787/34*

Schrock, N. (1996), 'Preservation factors in the appraisal of architectural records', *The American Archivist* **59**(2), 206–213.

Seadle, M. (n.d.), 'The elements of authenticity in digital preservation'.
URL: *https://www.vatlib.it/moduli/Seadle_EWASS2012.pdf*

Sebastian Claudius Semler, A. R.-R. (n.d.), ‘Archivierung von klinischen forschungsunterlagen’.
URL: *http://www.telemedizinfuehrer.de/free/2006/ss11_353_356.pdf*

Shirky, C. (2000), ‘Clay shirky on p2p’.

Shreeves, S. L., Habing, T. G., Hagedorn, K. & Young, J. (2005), ‘Current developments and future trends for the oai protocol for metadata harvesting’.

Sistla, A. P. & Welch, J. L. (1989), Efficient distributed recovery using message logging, *in* ‘Proceedings of the eighth annual ACM Symposium on Principles of distributed computing’, ACM, pp. 223–238.

Soden, R. & Palen, L. (2014), From crowdsourced mapping to community mapping: The post-earthquake work of openstreetmap haiti, *in* ‘COOP 2014-Proceedings of the 11th International Conference on the Design of Cooperative Systems, 27-30 May 2014, Nice (France)’, Springer, pp. 311–326.

Steinke, T., Schoger, A., Padberg, F. & Rechert, K. (2016), ‘Project emil–emulation of multimedia objects’, *iPRES 2016* p. 282.

Thibodeau, K. (2002), ‘Overview of technological approaches to digital preservation and challenges in coming years’, *The state of digital preservation: an international perspective* pp. 4–31.

UNESCO (n.d.), ‘Charta zur bewahrung des digitalen kulturerbes’.

URL: *http://www.unesco.at/kommunikation/basisdokumente/charta_digitales_kulturerbe_dt.pdf*

Ung, H. & Dalle, J.-M. (2010), Project management in the wikipedia community, *in* 'Proceedings of the 6th International Symposium on Wikis and Open Collaboration', ACM, p. 13.

Urdaneta, G., Pierre, G. & Van Steen, M. (2007), A decentralized wiki engine for collaborative wikipedia hosting., *in* 'WEBIST (1)', pp. 156–163.

Vangoor, B. K. R., Tarasov, V. & Zadok, E. (2017), To fuse or not to fuse: Performance of user-space file systems.

Weiss, S., Urso, P. & Molli, P. (2009), Logoot: A scalable optimistic replication algorithm for collaborative editing on p2p networks, *in* 'Distributed Computing Systems, 2009. ICDCS'09. 29th IEEE International Conference on', IEEE, pp. 404–412.

Wieringa, R. (2009), Design science as nested problem solving, *in* 'Proceedings of the 4th international conference on design science research in information systems and technology', ACM, p. 8.

Witt, M. (2008), 'Institutional repositories and research data curation in a distributed environment', *Library Trends* **57**(2), 191–201.

Zhong, C., Shah, S., Sundaravadivelan, K. & Sastry, N. (2013), Sharing the loves: Understanding the how and why of online content curation., *in* 'ICWSM'.

Zimmermann, H. (1980), 'Osi reference model–the iso model of architecture for open systems interconnection', *IEEE Transactions on communications* **28**(4), 425–432.

Zitzer, L. (n.d.), 'Bericht zur schlusspräsentation der gruppe 'theorien zum gedächtnis ii (assmann)''.
URL: *http://blogs.mewi.unibas.ch/asg/kategorie/thema/assmann-thema*

Published Papers

1. Goebert, Samuel, et al. 'A non-proprietary RAID replacement for long term preservation systems' Conferences, iPRES 2011 - Proceedings of the 8th International Conference on Preservation of Digital Objects, ISBN: 978-981-07-0441-4

2. Goebert, Samuel, Bettina Harriehausen-Mühlbauer, and Steven Furnell "Decentralized Hosting And Preservation Of Open Data." Archiving Conference. Vol. 2013. No. 1. Society for Imaging Science and Technology, 2013, ISBN: 978-1-63266-642-0.

3. Goebert, Samuel, Bettina Harriehausen-Mühlbauer, and Steven Furnell "Towards A Unified OAI-PMH Registry." Archiving Conference. Vol. 2014. No. 1. Society for Imaging Science and Technology, 2014, ISBN: 978-0-89208-309-1.