Spring 2014

Web Archive Services Framework for Tighter Integration Between the Past and Present Web

Ahmed AlSum
Old Dominion University

WEB ARCHIVE SERVICES FRAMEWORK FOR TIGHTER INTEGRATION BETWEEN THE PAST AND PRESENT WEB

by

Ahmed AlSum
B.S. May 2003, Mansoura University, Egypt
M.S. July 2009, Arab Academy for Science and Technology, Egypt

A Dissertation Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

COMPUTER SCIENCE

OLD DOMINION UNIVERSITY
May 2014

Approved by:

Michael L. Nelson (Director)

Michele C. Weigle (Member)

Hussein Abdel-Wahab (Member)

M'Hammed Abdous (Member)

Herbert Van de Sompel (Member)

ABSTRACT

WEB ARCHIVE SERVICES FRAMEWORK FOR TIGHTER INTEGRATION BETWEEN THE PAST AND PRESENT WEB

Ahmed AlSum
Old Dominion University, 2014
Director: Dr. Michael L. Nelson

Web archives have contained the cultural history of the web for many years, but they still have a limited capability for access. Most of the web archiving research has focused on crawling and preservation activities, with little focus on the delivery methods. The current access methods are tightly coupled with web archive infrastructure, hard to replicate or integrate with other web archives, and do not cover all the users' needs. In this dissertation, we focus on the access methods for archived web data to enable users, third-party developers, researchers, and others to gain knowledge from the web archives. We build ArcSys, a new service framework that extracts, preserves, and exposes APIs for the web archive corpus. The dissertation introduces a novel categorization technique to divide the archived corpus into four levels. For each level, we will propose suitable services and APIs that enable both users and third-party developers to build new interfaces. The first level is the content level that extracts the content from the archived web data. We develop ArcContent to expose the web archive content processed through various filters. The second level is the metadata level; we extract the metadata from the archived web data and make it available to users. We implement two services, ArcLink for temporal web graph and ArcThumb for optimizing the thumbnail creation in the web archives. The third level is the URI level that focuses on using the URI HTTP redirection status to enhance the user query. Finally, the highest level in the web archiving service framework pyramid is the archive level. In this level, we define the web archive by the characteristics of its corpus and building Web Archive Profiles. The profiles are used by the Memento Aggregator for query optimization.

To my wife, Yasmin.

ACKNOWLEDGMENTS

First and foremost, I am grateful and thankful to Allah who has blessed me with guidance, patience, determination, and good people that has supported me all through my life. I thank Allah for helping me and giving me the incentive to go on and finish this work.

I would like to acknowledge many people who helped me during my study. I would like to express my special appreciation and thanks to my advisor, Dr. Michael L. Nelson, for his help and support during my study at Old Dominion University. In addition to the academic mentoring and research guidance, Dr. Nelson opened endless opportunities for me to grow and succeed.

Also, I would like to thank my PhD committee members, every one of them has a special contribution to this work and to my life. Dr. Hussein Abdel-Wahab has been my mentor since I joined the program. His office was always open for me to discuss any issue that I might face. Dr. Michele Weigle always pushes me to achieve the highest quality standard in my research. I was fortunate to work with Herbert Van de Sompel and Robert Sanderson from Los Alamos National Laboratory. Our collaboration around the Memento project helped me in understanding the nature of world-class research. I benefited a lot from discussions with them on many occasions. I have a special appreciation for Dr. M'hammad Abdous. I dealt with Dr. Abdous in many occasions, and every time my respect for his noble character is increased. He was one of the persons who affected me a lot during my study at Old Dominion University.

I would like to thank Kris Carpenter Negulescu and the Internet Archive team for allowing me to intern with them in early 2012. The work accomplished at the Internet Archive was the basic foundation for my research. They provided me with the data collection and the computation environment that were required to complete this research.

I'm grateful to the web science and digital library group's former and current members: Martin, Hany, Justin, Scott, Chuck, Mat, Sawood, Mohamed, Amara, Lulwah, and Shawn. I spent wonderful time with them full of work, research, and fun.

I would like to acknowledge the Egyptian student community at Old Dominion University. For more than four years, we have been together facing many challenges including the hard time during the Egyptian revolution in 2011. I would like to thank each one of them, especially, Hisham Sayed, Ahmed Hesham, Mahmoud Kamel, Mahmoud Aly, and Ahmed Nagib.

I cannot forget my best friends who encouraged me during my past events and who will continue to in the future, Mohammed Seyam, Ahmed Abu-Zaid, and Mostafa Aly.

Regular words of thanks are not enough for my parents, who sustained my absence from my home country for this time. I felt the acceptance of their prayers and best wishes in every successful step in my work. Their support and encouragement helped me to recover from many disappointing moments through my research.

Finally, I would like thank the one who bore the most in the hard time during my study and to whom I dedicate this dissertation, my beloved wife Yasmin. During my course of study, she did her best to make our life easier, warmer, and more successful.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

"The Best Way to Predict the Future is to Invent it.", Alan Kay [166].

The web is the largest publication system today. Web archives are a way to preserve copies of the web materials before they change or disappear forever. Much of the current research efforts focus on the collection and preservation of web content. There is less focus on how to retrieve and use this content, and moreover how to integrate the current and the past web.

For example, news websites cover important events with details about how, why, and when they happened. So we are not in danger of "forgetting" that important events like Hurricane Katrina or the Virginia Tech shootings occurred, but we may forget the evolution of these stories. The evolution of the stories and the context in which they were reported are an important part of our cultural heritage and the footprints of history.

To review the evolution of these stories, you have to go back to the archived copies of the news websites during these periods, but not all the pages are archived, or they have been archived with slower rate than the events evolution. Even though the web archived data are preserved, users need novel methods to access it [139]. Moreover, the Internet Archive has the most and the earliest archived copies, but it does not have all the archived copies in the world, because there are different web archives with varying coverage of the web [42].

The UK Web Archive conducted research that showed web archives in general are being used by journalists and investigative reporters, litigants and detectives, civil servants, web designers, and academic researchers [253]. Bibliothèque nationale de France (BNF) studied the users of their web archive and discovered that researchers in political and social sciences are the most frequent visitors, in addition to professional and other personal searches such as web design and ebusiness [262]. The National Library of the Netherlands (Koninklijke Bibliotheek) [234] categorized web archive users as historians, sociologists, linguists, journalists, owners/designers of websites, public institutions, and the general public. The International Internet Preservation Consortium (IIPC) [3] proposed different use cases for access to the Internet Archive [149]; these use cases covered different levels of users with different interests. With this diversity of users' background, web archives should provide different types of interfaces to support this wide range of use cases. Currently, most of the web archives are managing their own interfaces, without any support for third-party applications due to the limitation of available Application Programming Interfaces (APIs). The development of APIs for web archives will attract third-party developers to implement customized applications on the top of web archive collections that will enable the usage of the web archives for general and special users and allow web archives to focus on other tasks such as crawling and preservation.

In this research, we build *ArcSys*, a new service framework for a tighter integration between the current and the past web by leveraging the set of methods and approaches to access and use the archived web data. The research starts with a review of the current trends in web archiving and then a detailed discussion about the service framework. The study will contain the challenges related

to the temporal nature of the archived web with a cost model related to the size of the archive in addition to the implementation prototype with quantitative evaluation.

1.1 MOTIVATION

The lack of standard APIs for web archives and the needs for such APIs in web archiving use-cases motivated this work. The emerging of BigData techniques in the last few years made the implementation of new services more efficient. In this section, we will discuss the motivation behind this research, in addition to an overview about some related projects.

1.1.1 LACK OF APIS

An API extends the functionality of a website to third-party developers to create new services and new applications on the top of the original system infrastructure. Google, Bing, and Twitter provide a set of APIs that enable users and developers to build their own applications and services. For example, Google released the Google Maps API[1] that is used to build interactive data visualizations on maps and location-based applications. Twitter provides a Search API[2] to query the tweets programmatically. In addition to commercial services, the US government calls on agencies to make APIs the new default method for making government information accessible to the public [193]. Several agencies have released APIs such as the US Census Bureau[3] and the Federal Communications Commission[4]. Flanders et al. [118] discussed the importance of having APIs to facilitate the design of innovative online services and to cover the information sharing gap.

Unfortunately, most current web archives do not provide APIs to access their data. Third-party developers cannot build new interfaces due to the absence of an API. As a user, you are limited to the functionality that is provided by the web interface. As a researcher, you could use the web interface with page scraping techniques [83, 173, 222, 286] or get the data directly from the web archive based on a partnership agreement. For example, you cannot get raw web archived data (i.e., in ARC/WARC formats) from the Internet Archive. Even though the Internet Archive has a simple API in the Wayback Machine, it is not enough to do intensive research. So you may need to get a copy of WebArchive Metadata format (WAT) files [123] for some specific collections based on agreements.

1.1.2 WEB ARCHIVING USE CASES

In 2006, IIPC [3] published a report entitled "Use Cases for Access to Internet Archives" [149]. In this report, the IIPC access group illustrated some web archive usage scenarios and defined the required interfaces to satisfy these use cases. The use cases report has been supported with a prototype report [148] that defined the implementation of these interfaces. The proposed use cases varied between a journalist who prepared a report using a historical information from the web archive to a lawyer who extracted an evidence in a civil case from the archived web data. In addition,

[1] `https://developers.google.com/maps/`
[2] `https://dev.twitter.com/start`
[3] `http://www.census.gov/developers/`
[4] `http://www.fcc.gov/developers`

there are use cases on the collection level such as analyzing the evolution of the web technology. More details about the use cases and the suggested prototype can be found in [149, 148]. The Portuguese web archive showed that searching for a known page or site was the most frequent need, and collecting information about a subject written in the past was the second most frequent need [101].

Dougherty et al. [109] conducted a set of interviews with archivists, technicians, and researchers between 2008 and 2010. They summarized the web archiving practices in the relationship to researchers and research needs. They prepared a set of recommendations in three groups: building community, building tools and resources, and building practices. For building tools and resources, they highlighted the importance of sharing the tools among researchers and librarians. They also suggested the usage of web services and APIs with standard and relational movable metadata using Resource Description Framework (RDF) [176, 209] and Linked Data [71, 73]. Thomas et al. [272] identified a set of challenges and opportunities in the web archives that should be supported by funding agencies. His first point was the creation of advanced set of tools to help the archivist to control and validate the crawlers; tools and APIs to build workflows or specific applications; and tools to capture, analyze, and search metadata. Ashley [47] mentioned the importance of having web archives with APIs, not just document-centered (sometimes the content, presentation or both). APIs allow many views and uses to emerge from the data.

In 2011, IIPC published "Harvesting Practices Report" [199], a survey among the web archiving members in IIPC. In this report, they asked about the areas of the user interface that they would like to improve. The answers listed the required services that will enable enriched interaction with the web archive such as:

- Enable temporal navigation between different versions of a website.
- Full-text search with weighted ranking.
- Use language filters (e.g., English/Welsh).
- Move some site analysis features for curators out to the public archives (e.g., "show me all the PDFs published on a particular site since a particular date").
- Provide raw WARC [16] format download.
- Import of metadata records into other repositories, so that pointers to web archived content can be found side-by-side with scanned materials.

The Internet Archive forum[5] receives requests from the broad user community about the availability of APIs to access the archived data. Here are two examples quoted from the Internet Archive forum:

- Ponguru asked on May 17, 2010:

  "*Hi All - I am new to Archive.org. A few quick questions (1) Is there any API or tools available to access the Archive.org contents programatically? (2) Are there any research papers where Archive.org*

[5] `http://archive.org/iathreads/forums.php`

was used for data collection / analysis (e.g. studying a particular topic over time, etc.)? I digged a little bit, could not find much, so checking with the group. "

- Redth asked on Feb 16, 2011:

  "*Hi, I am helping to develop a service which catalogs domains and collects various statistics about them. One of those statistics we are trying to capture is the oldest date that a domain has a page archived for. Currently, we can see this data by using the new beta for example:* `http://waybackmachine.org/*/digg.com` *The data on this page we are interested in is "going all the way back to November 11, 1998.", so the date, Nov. 11, 1998. I'm wondering if there's any way we can access this via an API, preferably something that would support batching requests. Scraping the data is proving to be a bit unstable, and it seems as if the server is blocking our requests fairly regularly.*

  I think it would be better for both of us to have some sort of API to use. We'd definitely be willing to pay some amount for such an API (or donate). Is this a possibility?

  Thanks!"

The previous examples emphasize the importance of APIs to enable the development of customized applications. It will help the web archives to delegate the development of the access interfaces to third parties while the web archivists can focus on harvesting and preservation.

1.1.3 WEB ARCHIVING AS BIG DATA

The web is growing rapidly, as is the archived web. Everyday, 250 million pictures are uploaded to Facebook, 300 billion emails are sent, and 340 million tweets are posted to Twitter [185]. The current Internet Archive web archive corpus reached 5+ PetaBytes with 360 billion mementos [237]. Alexandria Bibliotheca needs one year of continuous computation to recompute the checksum for its corpus using its current infrastructure. From the data point of view, the web archive is a huge amount of unstructured documents with a tree hierarchy and temporal relation. This huge corpus is an example of big data [288] which describes data sets so large and so complex that they cannot be processed with normal database management tools.

The Apache Hadoop [1] software library is a framework that allows distributed processing of large data sets across clusters of computers using a simple programming model. Hadoop became the main computation solution for Internet Archive and UK Web Archive [28] in order to process the large amount of data contained within the web archives. The Internet Memory Foundation [235] used Hadoop for computation and HBase [122] for data storage. In this research, we use Hadoop/MapReduce for preprocessing the archived web data by running various tasks that extract important information from the archived web pages. The output is stored in a NoSQL database which fits our needs.

1.1.4 RELATED PROJECTS

The current web archives have a limited set of Graphical User Interfaces (GUIs) that cover the basic functionality to browse, search, and retrieve the archived web. Section 2.4 gives a complete overview of the current trends in accessing the archived web; most of these interfaces display the web page as it appeared in the past, but it did not help them to gain knowledge from the archived

web. Even though we have a few projects that tried to get benefits from the archived web, there is no way to reuse this information. For example, the Stanford WebBase project [8] has a repository with more than 260 TB of 7 billion pages. WebBase does not have an access point to access their material, the user can only download the entire collection via FTP.

There are a few initiatives that extract value from the available archived web data. The *Living Web Archives* (LiWA) [13, 80] is an European funded project that focused on six use cases: archive fidelity [220], web spam filtering [114, 113], archive coherence [259, 63], crawl coherence (Online Quality) [107], archive interpretability [267, 294, 266], streaming application [230], social web application.

LiWA was followed by the *Longitudinal Analytics of Web Archive data* (LAWA) [17], LAWA is a European funded project that developed an experimental testbed for analyzing the data in large-scale. The main goals for the project are building stable infrastructure supported with the methods and tools to implement the main data analyzing tasks for the unstructured web data. They focused on four areas: web scale data provision, which discusses the optimal data structure for large-scale computation to reach "Web scale" limit [257]; web analytics, which studies the methods and tools for aggregating, querying, mining, and analyzing "Web scale" data [293, 160]; distributed access to large scale data sets, which investigates for the methods to keep the large-scale data accessible to the researchers [287, 246]; and virtual web observatory, which implements a set of core analysis solutions [290, 260, 284].

Web Archive Retrieval Tools (WebART) [29, 147] is a joint project between University of Amsterdam[6], National Research Center for Mathematics and Computer Science (CWI)[7], and National Library of the Netherlands (KB)[8]. WebART addressed the research question why does the web archiving concentrate on preservation more than using and accessing the data? WebART collects and builds prospective use cases for web archiving, then it supports these use cases with the required software and tools that maximize the usage of web archives.

Common Crawl [23] is a non-profit organization that aims to build and operate an open web archive repository that the researchers can use easily. The crawling data is around 5 billion pages that are available on Amazon's S3 service. The researchers could build, download, or access the data through map/reduce tasks on Amazon EC2. The Common Crawl corpus was used in various research [72, 227, 212, 110].

The Memento project [278, 279, 282] provides the dimension of time to the HyperText Transfer Protocol (HTTP) [117]. Memento introduces special resources called "TimeGates" that redirect the user-agent to the "closest" possible version of an archived resource. A Memento TimeGate is a place that is aware of prior versions of web resources. A Memento Aggregator aggregates multiple TimeGats. Memento is explained in detail in Chapter 3.

1.2 RESEARCH QUESTIONS

The success of the web is based on easy access to information. We can assume the success of the web archiving will be measured by the means of access the archive provides to the preserved

[6] `http://www.uva.nl`
[7] `http://www.cwi.nl`
[8] `http://www.kb.nl`

material [149]. This research aims to leverage the integration between the past web with the current web by building enriched access mechanisms. So the main question is: "*How can we enrich the web archive access interface with the conjunction of the live web?*". Our study shows that supporting web archives with a set of APIs may encourage third-party developers to integrate the web archive in their products. This research direction leads to more research questions:

- What are the required services for web archiving user community? (Chapter 2).
- Shall we work on the web archive collection as one entity or on different levels? (Chapter 4).

We found that we can divide the collections on four levels. For each level, we define more research questions:

- How can we use the web archive content beyond the full-text search service? (Chapter 5).
- What are the metadata fields that could enhance the user browsing experience? (Chapter 6).
- How can we develop access interface to the temporal web graph? (Chapter 6).
- How can we optimize the thumbnails creation in the web archive? (Chapter 6).
- How can we use the HTTP redirection to enhance the URI-lookup query? (Chapter 7).
- How can we optimize the query routing mechanism across the web archives? (Chapter 8).

1.3 CONTRIBUTIONS AND PUBLICATIONS

To enhance the access model for web archives, we develop *ArcSys*. ArcSys is a new service model that exposes an API interface for the web archive collections. ArcSys has a set of subsystems that are divided into four levels: Content, Metadata, URI, and Archive.

- **Content services:** We develop ArcContent, a complete system to extract, preserve, and access web archive content. The content services focus on the actual page content. ArcContent provides a set of filters and wrappers that expose the original web archive content in different presentation that based on the user requests.
- **Metadata service:** We develop two systems ArcLink and ArcThumb. ArcLink [39, 40] is a distributed system to construct, preserve, and deliver the temporal web graph for large-scale web archives. ArcLink exposes an API interface to access the link structure metadata on fine-grained level. ArcLink optimization techniques reduce the input corpus to 29%, extract efficiently from the WARC files with 61% of the regular page scraping, and deliver the link structure in RDF/XML format. ArcLink provides an adequate platform for new applications such as Temporal PageRank which has a weak relationship between the rank at each month and Time-Indexed Inlinks that gives information about URI through the time.

  In ArcThumb [41], we explore different algorithms to optimize the thumbnail creation procedure for the web archive based on information retrieval techniques. We study different features based on HTML that correlate with changes in rendered thumbnails so we can know in advance

which archived pages to use for thumbnails. We find that SimHash correlates with changes in the thumbnails ($\rho = 0.59, p < 0.005$). We propose different algorithms for thumbnail creation suitable for different applications, the number of thumbnails to be generated is reduced to 9% – 27% of the total size.

- **URI service:** URIs are typically used as lookup keys to archived versions, or "mementos" in a web archive. This is straightforward until the URI on the live web issues a redirect: $R \rightarrow \overline{R}$. Then it is not clear if R or $\overline{R}$ should be used as the lookup key to the web archive. We run a quantitative study to conclude a set of policies that will help the client to reach the memento in the existence of HTTP redirection. The first policy covers the URIs with HTTP redirection status. It successfully resolves 17 URIs out of 77 URIs that do not have mementos but have HTTP redirection status. The second policy covers the mementos with the HTTP redirection status. It helps the client to get the nearest memento to the requested date/time especially in the multi-archive environment.

- **Archive service:** This level covers the individual web archive characteristics and the global web archiving activities around the world. For the global archiving activities, we calculate "How much of the web is archived?" [35, 36]. We study the question by approximating the web via sampling URIs from DMOZ, Delicious, Bit.ly, and search engine indexes and then measuring how many of copies of these sample URIs exist in various public web archives. Each sample set provides its own bias. In 2011, we calculated an archive percentage of 90% for DMOZ, 97% for Delicious, 88% for search engine, and 35% Bit.ly. The results from our sample sets indicate that from 35%-90% of the web has at least one archived copy, 17%-49% has between 2-5 copies, 1%-8% has 6-10 copies, and 8%-63% has more than 10 copies in public web archives. The number of copies varies as a function of time but no more than 31.3% of URIs are archived more than once per month.

  For the individual web archive characteristics, we develop an automatic technique to construct profiles for web archives [42]. The results show that the Internet Archive is the largest and has the broadest coverage that ranges from 77% to 98% for the general samples. The national archives have good coverage of their domains and languages, and some of them extend their selection policies to cover more domains. We use the profiles in query routing optimization for the Memento Aggregator that selects the most probable web archives by matching the profiles with the query request. We propose $Recall_{TM}@n$ to evaluate the success of the web archive selection algorithm. We achieve $Recall_{TM}@3 = 96\%$, even when the IA is excluded, we achieve $Recall_{TM}@3 = 64.7\%$.

1.4 ORGANIZATION

The dissertation is organized into the following chapters, separated by topic and contribution.

- **Chapter 2: Web Archiving Trends** - gives an overview about the web archiving paradigm in general, focusing on the main web archiving tasks (selection, harvesting, access, and the quality assurance).

- **Chapter 3: Memento Framework** - discusses the Memento protocol concepts and terminology.
- **Chapter 4: Web Archiving Services Framework** - proposes the new service framework that will enable the user to access web archiving data. The chapter starts with a general discussion about the services layer, then a complete catalog of the proposed services.
- **Chapter 5: Content Service** - discusses the ArcContent subsystem with an overview about the supported filters.
- **Chapter 6: Metadata Service** - gives an overview about web metadata, then it explores two systems: ArcLink, for retrieving link structure; and ArcThumb, for thumbnail creation in the web archive.
- **Chapter 7: URI Service** - studies the HTTP redirection in the web archive and how it can be used to enhance the URI-lookup mechanism in the web archive.
- **Chapter 8: Web Archive Service** - discusses the creation of web archive profiles of the distribution of the archived web between the institutions based on the Top-Level domains, and content languages. The Memento Aggregator will use these profiles to optimize the query routing techniques.
- **Chapter 9: Conclusions and Future Work** - concludes the research results and contributions. We point to other future work opportunities.

CHAPTER 2

WEB ARCHIVING TRENDS

In this chapter, we will give an overview of the current trends in the web archiving including: crawling, preservation, and retrieval.

2.1 WEB AND WEB ARCHIVING

Web archives preserve the web for future access. In this section, we give an overview about the web and the relationship with web archive.

2.1.1 WORLD WIDE WEB

The World Wide Web (WWW, or simply web) is "an information space in which the items of interest, referred to as resources, are identified by global identifiers called Uniform Resource Identifiers (URI)" [152]. The first proposal of the web, was published by Berners-Lee in 1989 [65], described the web as an elaborated information management system. Figure 1 shows the relation between the main parts of the web: *URI*, *Resource*, and *Representation*.

Uniform Resource Identifiers (URIs) [67] provide a mechanism to identify the resources across the web. URI is characterized by the following definitions:

- *Uniform*: Different types of identifiers can be used in the same context however the dereference mechanisms are different. It is achieved by allowing uniform semantic interpretation across all the types of resource identifiers.
- *Resource*: A resource can be anything that has identity.
- *Identifier*: An identifier is an object that can act as a reference to something that has identity. In the case of URI, the object is a sequence of characters with a restricted syntax [68].

Dereferencing the URI is the process of using the URI to access the resource by retrieving a representation of the resource (e.g., HTTP GET) or adding or modifying a representation of the resource (e.g., HTTP PUT). Dereferencing is defined by the URI scheme (e.g., HTTP and FTP).

A representation is "Data that encodes information about resource state" [152]. Representations of a resource may be sent or received using interaction protocols. The protocol is responsible for determining the form of the representation that should be delivered upon request. HTTP protocol uses the content negotiation mechanism to select the best representation of the resource.

Content negotiation [117] is the mechanism to deliver multiple representations through the same URI. Negotiation is between the requesting agent and the server. Based on this negotiation, the server will come to a decision about which representation should be served. HTTP defines two kinds of content negotiation: server-driven and agent-driven negotiation. These two kinds of negotiation are orthogonal and may be used separately or in combination.

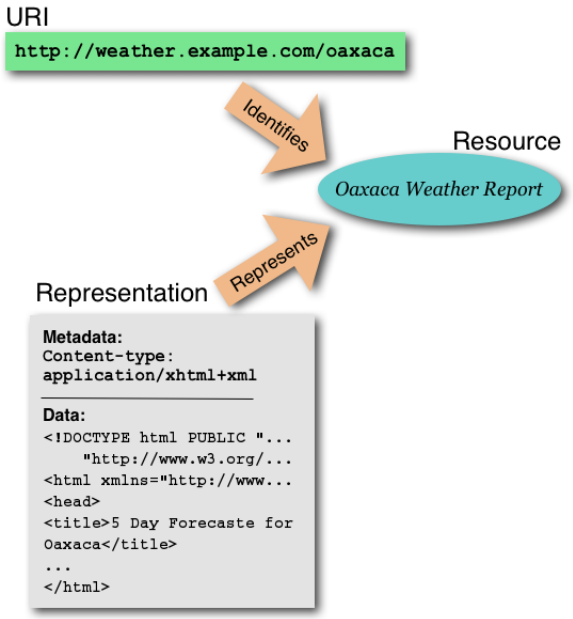


Fig. 1 The relationship between URI, Resource, and Representation (figure taken from [152]).

Table 1 Content negotiation request headers

Header	Description
Accept	It is used to specify certian media type in the response
Accept-Charset	It is used to specify acceptable character sets in the response
Accept-Encoding	It is used to specify acceptable content-codings in the response
Accept-Language	It is used to specify preferred natural languages in the response

HTTP/1.1 includes various request-header fields to enable the user agent to send its capabilities and user preferences (as appeared in Table 1). The Vary response-header field can be used to express the parameters the server uses to select a representation that is subject to server-driven negotiation.

For example, the URI http://foo.edu/paper may have different versions [145]:

1. HTML, English
2. HTML, French
3. Postscript, English

The server will respond with the first representation (HTML, English), if the user agent's request headers include:

```
Accept: text/html;q=0.8
Accept-Language: en-US,en;q=0.5
```

Even though Berners-Lee [66] added time as one of the dimensions in his original design issue document (that was the root of the content negotiation in HTTP protocol [117]), content-negotiation

in the time dimension did not become available until Memento protocol appeared in 2009 [278]. Chapter 3 discusses the Memento protocol in detail.

2.1.2 WEB ARCHIVING

Library of Congress defined Web Archiving [10] as "The process of creating an archival copy of a website. An archived site is a snapshot of how the original site looked at a particular point in time." In recent years, there have been various studies about web archiving.

In 2006, Masanès [197] reviewed the state-of-the-art of methods, tools, and standards to build a web archive. Masanès discussed the web archiving paradigm from the computer science and librarian point of view. Masanès stated the main challenges for web archiving with an overview about the current web role in the society, then he explained the main procedures of the web archiving such as selection, crawling, access, analyzing, and preservation. Masanès discussed two case studies: Internet Archive and DACHS, a research-driven selective Web archive. Brown [81] provided a practical guide for archiving the Web; the book examined the process of archiving from selection, collection, storage, and delivery to the user. The book also covered legal issues and quality assurance. Brown illustrated the web archiving process as a workflow (figure 2) with the following steps:

- **Selection:** A selection policy should be determined to define a set of web resources for collection, with clearly defined boundaries and collection frequencies.
- **Maintenance:** The selection policy should not be static, but must be updated to reflect changes in internal and external factors to ensure its continuing relevance and fitness for purpose.
- **Collection:** collection is the process of collecting the website content from the web to be ready for preservation. There are different collection methods such as: client-side (remote harvesting), server-side (direct transfer and database archiving).
- **Quality Assurance and Cataloging:** one must ensure that the selection policy is being implemented correctly and the collected content is described adequately to support its long-term preservation and use.
- **Storage and Preservation:** Digital information is stored in the form of bits that have no inherent meaning. So preservation is the process of saving the digital content to ensure the continued accessibility over time. There are three basic strategies: Refreshing, Migration, and Emulation. *Refreshing* is the process of transfering the data between two types of the same storage medium [285]. Refreshing is an effective preservation technique when the bit stream could be encoded independent of the software or the hardware. *Migration* is the process of converting the original object to a new form which can be accessed using current methods [285]. Migration includes refreshing but it differs in not always copying the exact bit stream but it may convert the resource from hardware/software configuration to another or from computer generation to another. *Emulation* is the process of developing new methods for accessing the object in its original form [238].

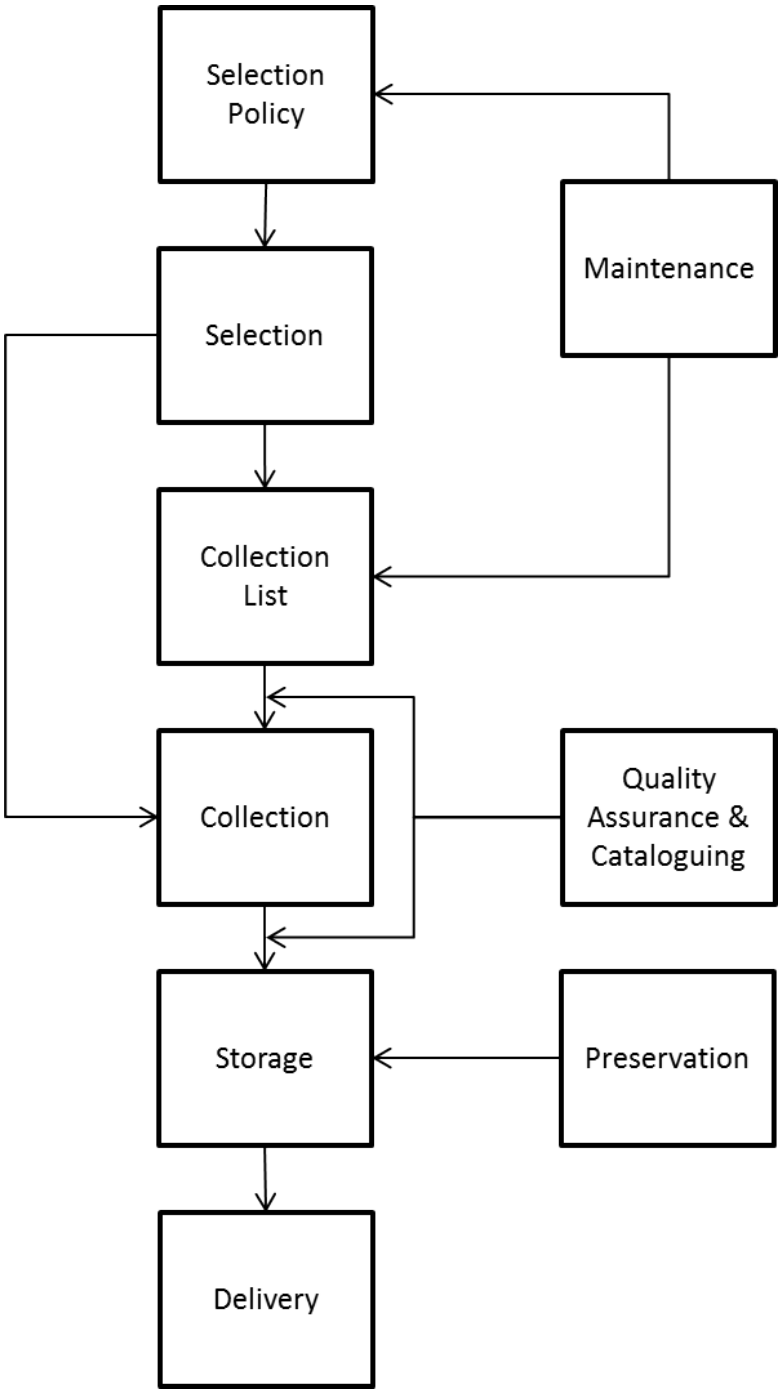


Fig. 2 The web archiving process (figure is taken from [81]).

- **Delivery:** the community of users must be able to access the web archive content. The delivery system should guarantee two basic functions: a means for users to discover the content and a means to deliver that content in a meaningful form.

Brügger [82] provided a step-by-step guide to archive a website based on the framework of the Danish research project Media and Democracy in the Network Society (MODINET)[1]. Brügger focused on "micro-archiving" which is archiving carried out on a small scale in space and time with a limited experience on the archiving. In other words, he focused on the "Website" and not the "Web".

Shiozaki and Eisenschitz [249] published the results of a questionnaire survey that was conducted between 16 national libraries designed to clarify how national libraries justify their web archiving activities. The survey results concluded that:

(a) There are benefits from running these web archiving initiatives that exceed the overall costs.

(b) The libraries bear the costs for these initiatives more than the stakeholder.

(c) The national libraries try to work on the legal risks (e.g., legislation, contracting and opt-out policies) although there are trade-offs in terms of costs for negotiation, scope of access, and size and scope of the web archive.

Kelly et al. [167] published a report about "PoWR: The Preservation of Web Resources" project. The handbook aimed to give a suggestion for best practices to enable the archiving of the web and web-based resources. Gomes et al. [125] performed a survey on the web archiving initiatives identifying 42 web archives around the world. Gomes found that the number of web archives has grown since 2003. They studied the archived data and the means of access to these archives. Niu [218, 217] evaluated several web archives to study the selection, acquisition, and access techniques of the web archives. Niu limited her study to web archives with an English interface.

National libraries have published their web archiving initiatives in various studies, for example, National Library of France [48], Portuguese web archive [126], UK web archive [53], National Library of the Czech Republic [283], National Taiwan University [93], National Archives of Australia [141], Netarkivet web archiving [98], National and University Library of Slovenia [165], and China Web InfoMall [291].

2.1.3 TYPES OF WEB ARCHIVES

There are different ways to categories the web archives. We can categorize based on the harvesting strategy to client-side, database, and transactional archiving (section 2.2); and based on the selection to specialized and general purpose archives (section 2.3). In this section, we will address other criteria that we could use to categorize the web archives.

Masanès [197] defined three categories of web archiving based on the scope: site-centric, topic-centric, or domain-centric. In site-centric archiving, the crawler focused on a specific site, it is common in private archives that some organization are developed and used its own web archive internally for their own purposes. It may be archived using web copier or special archiving service

[1] `http://www.modinet.dk`

provider, for example, Hanzo Archive[2] is archiving the Coca Cola website [30]. In topic-centric archiving, the archiving process is related to specific topic or event, not just the website. The selection of seed URIs that are related to the topic could be manual (e.g., Archive-It) or automatic (e.g., Twittervane [20]) An extra effort should be done to select the seed URIs that are related to the topic, for example, Minerva project[3] from the Library of Congress. Finally the domain-centric archiving, the archiving process is not driven by the content but by the location, for example, National Archives and Records Administration (NARA)[4] has a focus on .gov Top-Level Domian (TLD).

The source of the seed URIs is another criteria. First, the seed URIs are retrieved from a public directory such as DMOZ, which is where the Internet Archive draws their seed URIs. In this case, the scope of the web archive may range from specific domain to world wide domains. The crawler expands this list by discovering more URIs. Second, the seed URIs may be promoted by the user directly. Some social bookmarking websites enable the user to archive the URI content. Third, it may be there are no seed URIs. This is the case of versioned documents systems such as Content Management Systems (CMS) and Wikipedia that used to save copies of all the versions. So all the pages are subject to archiving, then the user could browse the history through the same interface.

Copying the websites to another place before sunset could be considered as a web archive that preserve the final version of the websites that are no longer available online, however there is no machine-readable interface about copying. For example, the conferences websites used to be copied to another destination after the completion of conference. JCDL 2002 is not accessible on its original URI[5] but is accessible through the JCDL website[6].

Institutions running web archives can be divided into three types:

- Nonprofit organizations such as Internet Archive and CommonCrawl.
- National libraries such as UK Web Archive and Bibliothèque nationale de France.
- Private companies such as Hanzo Archive and REED Archives.

2.2 SELECTION

Selection is the process of determining a set of web resources that should be collected and preserved. This set of web resources is defined in "Selection Policy". Selection policy is a general guiding document that defines the collection development policy [196].

The IIPC Harvesting Practice Report [199] surveyed between 17 members in IIPC [3], and 13 of them responded to the current archiving policy question.

- **All domains:** the Internet Archive includes all content published to any domain or host that is not excluded by robots.txt [9, 178]. Internet Archive ignores robots.txt for all embeds resources.

[2] http://www.hanzoarchives.com/
[3] http://www.loc.gov/minerva/
[4] http://www.archives.gov/
[5] http://www.jcdl2002.org
[6] http://jcdl.org/archived-conf-sites/jcdl2002/

- **Partner customized policy:** the California Digital Library (CDL) and Archive-It do not have their own policy but they allow the curatorial partners to determine their own policies and setup their own crawler using their favorite seeds URIs and crawling frequency.
- **National cultural heritage:** some national archives perform the selection based on their legal deposit law. Subjects for selective harvesting include media, governmental, academics, technology and medicine and important events on nation-wide. For example, Bibliothèque nationale de France (BnF) is mandated to harvest websites of the "French national domain" under the French Heritage Law, or "Code du patrimoine". The law has been published on August 2006. The law allowed the national libraries to cover the internet communications: "is also liable to legal deposit every sign, signal, writing, image, sound or messages of every kind communicated to the public by electronic channels" (clause 39) [48].
- **User submission:** some services enabled the users to archive their favorite URIs such as WebCite and Archive.is. So the selection policy depends on the users' selection. For example, WebCite (which is an IIPC member) enables the user to push a copy to the WebCite which returns a new URI suitable for citing the unchanging archived version. Recently, Internet Archive provided a "Save Page" feature [237] to allow the user to submit to the Wayback Machine.

The source of the seed URIs may vary between the archives. Internet Archive suggests adding your website to Open Directory[7] (DMOZ) to be included in IA's seed list [5]. UK Web Archive selection process is manual, dependent upon internal subject specialists or external experts to contact them and nominate websites for archiving in the UK Web Archive [2]. Twittervane [20] is a new tool developed by UK Web Archive to automate the selection of websites for archiving. Twittervane uses the crowdsourcing [146] approach that compliments the manual selections provided by subject specialists and other experts. The project develops a tool for analyzing twitter content that determines which websites are shared most frequently around a given theme over a given time period; and link to UK web archive infrastructure to support harvesting of sites that fall within the UK domain.

Masanès [196] divided the selection process into three phases: preparation, discovery, and filtering. Figure 3 shows the relation between the phases with the main output of each phase. In preparation, the curator should determine the target of the collection, capture policy, and the tools that will be used to implement it. The preparation phase will differ based on the type of the archive. Discovery phase determines the list of entry points, the capture frequency, and the scope of the capture. The discovery has two types based on the collection building approach: manually or automatically. Finally, the filtering phase limits the discovery phase to be with in the selection policy that was defined in the preparation phase.

2.3 ACQUISITION

The term "acquisition" (also known as harvesting or collecting) designates the various technical means used to get the content into the archive [197]. The next step in the web archiving life cycle is how to capture the web site content for future preservation. In general, there are three

[7] `http://www.dmoz.org`

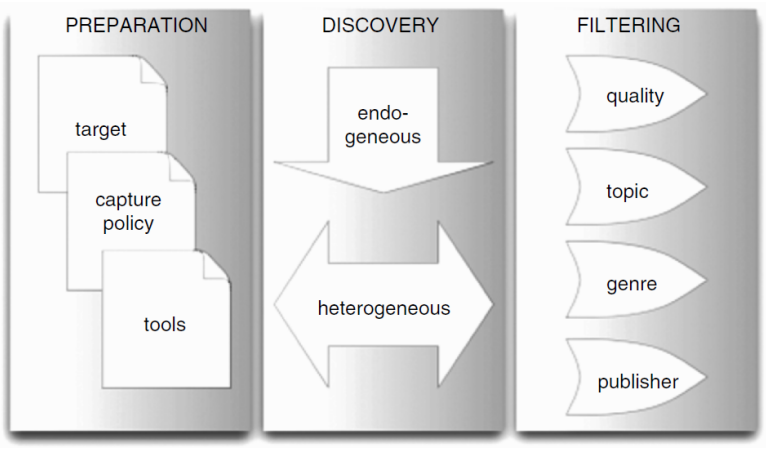


Fig. 3 The phases of the selection process (figure is taken from [196]).

main techniques for harvesting the web site content: remote harvesting, database archiving, and transactional archiving.

2.3.1 REMOTE HARVESTING/CLIENT-SIDE ARCHIVING

This is the simplest and the common way to crawl the web for archiving purposes and was adopted from search engine crawling techniques. In this method, the web crawler works as a web browser and accesses the web as a normal user. Crawlers start from seed pages, parse them, extract links and fetch the linked document. Figure 4 shows a block diagram of how the web archive crawler accesses the web sites.

In this technique, the crawler aims to get a complete snapshot of the web page by downloading the HTML page with all its embedded resources. However, the crawler may have to hit the same web site many times, so crawlers apply a politeness rule by keeping delay between each HTTP request (typically 1 to 3 seconds). For the larger web sites, the crawler may take days to capture a complete snapshot of the web site which may cause a coherence problem [259], whic is an inconsistency between the web pages due to the crawling in different times.

Crawling Strategies

The crawl depends on crawling strategy that determines what the order of pages to be crawled. The *Important Page First* strategy has been studied in different ways. Cho et al. [96] proposed a policy to visit the most important page first based on re-ordering the visited URI. They evolved the ordering schema based on some importance metrics such as: similarity to a query, backlink count, and PageRank. Baeza-Yates [51] compared different strategies based on the amount of available information about the crawling cycle (no-information, partial information, or all the information).

Ben Saad and Gançarski [61, 62] adapted new crawling strategies to increase the quality of the web archive for the completeness and coherence. Completeness measures the ability of the archive to contain the largest amount of useful versions. Coherence measures how much the archive reflects

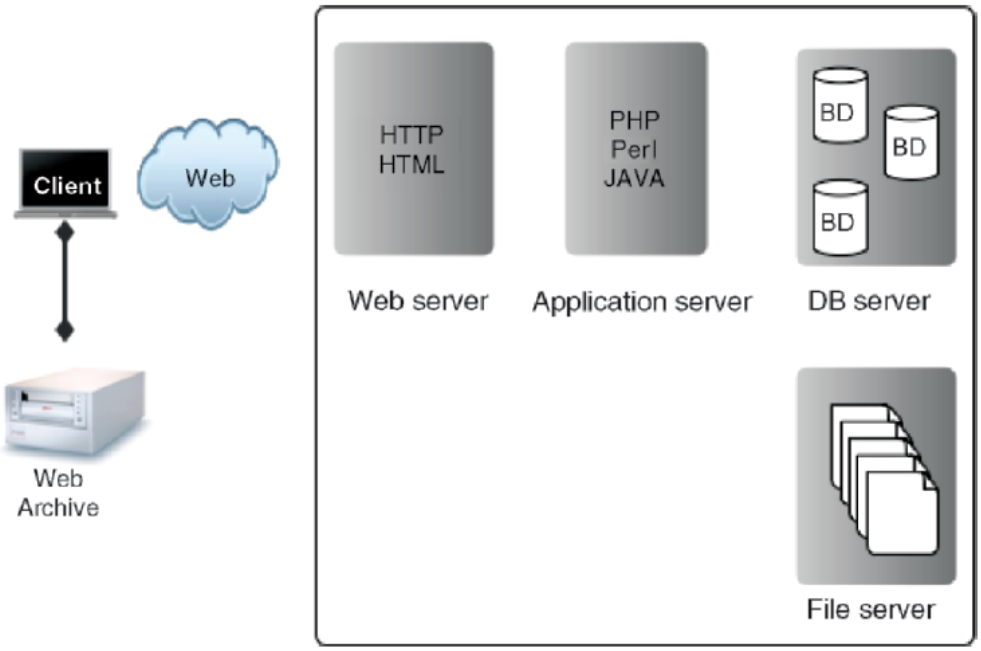


Fig. 4 Client-side archiving (figure is taken from [197]).

the snapshot of web sites at different points in time. One strategy based on recording the pattern of changes for the web pages and configure the crawler frequency to visit the web pages based on this computed pattern. Their experiment showed a completeness gain up to 20% in case of limited resources. Another strategy focused on downloading *the most important versions*. An important version is a version that has important change with respect to the last one archived of the same page. Hence, unimportant changes in the page (e.g., advertisements and decoration) can be ignored and useful information is captured by a single crawl, maximizing the use of resources.

Heritrix crawler executes a primarily breadth-first, order-of-discovery policy for choosing URIs to process, with an option to prefer finishing sites in progress to beginning new sites (i.e., *site-first* scheduling) [210].

Tools

The IIPC Harvesting Practices Report [199] showed that 19 out of 21 web archives are using "Heritrix" for harvesting the web content. Heritrix is the Internet Archive's open-source, extensible, web-scale, archival-quality webcrawler project. IA started the development in 2003 [24, 210] to be able to do crawling internally for its own and other partners (before that Alexa[8] Internet used to donate bi-monthly snapshots to IA).

[8] `http://www.alexa.com/`

Challenges

As new technologies start to appear, the web archiving crawler should adapt itself to capture it. It results a list of harvesting challenges [236]:

- *Ajax and Web 2.0/3.0*: In Ajax (Asynchronous JavaScript and XML) [121] applications, the state of the user interface is determined dynamically, through changes in the DOM that are only visible after executing the corresponding JavaScript code [208]. In this case, the crawler should be able to walk through these different states based on navigational path.
- *Streaming Media*: Streaming websites, such as YouTube.com and Vimeo.com, became a challenge to the normal web crawling techniques. Web site owners used to discourage bulk/distributed downloads via bots.
- *URI challenges*: URI is the key of the crawling procedure. The URI model has been altered by some web sites [270]. For example, the dynamic generated URI where you may have different URIs for the same resource.
- *Mobile*: Various websites provided a special interface for the handheld devices. Even though this interface may differ from the original website, web archives typically do not archive to this different format [245].

2.3.2 DATABASE ARCHIVING

Many tools and technologies support the usage of the database to deliver the website interface. Database archiving refers to methods for archiving the website by archiving the underlying content database. The basic idea behind database archiving is exporting the data in standard format that could be imported later into the archive and providing a general access method for the data.

McCown and Nelson [203] discussed three methods to inject the server components, including database, into the web archive. The first method is exposing the raw components in compressed files that area ready to be crawled, but it is not recommended because the search engines avoid crawling the binary content [206]. The second method is called robots vaults [274] where the server components is encoded into special pages that are for crawlers only, however the search engine may avoid them because it may be considered as spam. The third method is called dispersion through preexisting content where the server components are injected into special tags in the crawled pages such as HTML comment tags in HTML pages.

Tools

Software Independent Archiving of Relational Databases (SIARD) [15] is an archiving solution for relational databases, and was developed by Swiss Federal Archive. SIARD suite is based on international standards such as XML, SQL:1999 and UNICode.

DeepArc [12] was developed by National Library of France (BnF) to transform relational database content into XML for archiving purposes. DeepArc allows the users to map an existing relational

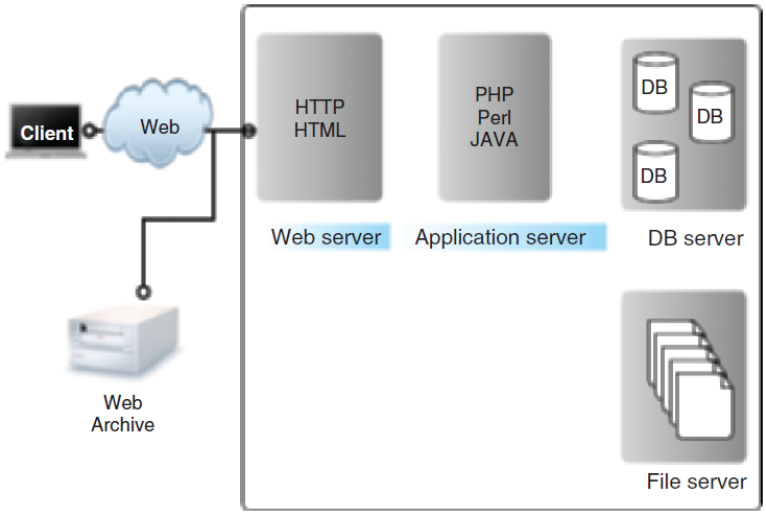


Fig. 5 Transactional archives (figure is taken from [197]).

data models to one or more data models specified by XML Schema. DeepArc can export the database content into the specified schema.

2.3.3 TRANSACTIONAL ARCHIVING

Transactional archiving intercepts the transactions that occurred between the web server and the web browser, then it stores and archives this pair of request/response. This archival procedures depends on event-driven not content-driven. It could be implemented by applying a filter to the web server to record each input/output (request/response) flow [197].

Tools

SiteStory [55] is an open-source transactional archive that runs on ApacheWeb servers. SiteStory provides an access through Memento compatible interface. The archival data can be exported to WARC format that can be uploaded into Wayback Machine instance. Brunelle et al. [85] discovered that SiteStory has low effect on the content server performance. Figure 6 shows SiteStory's architecture diagram.

pageVault [11] is a client-side transactional archive. It archives and indexes every unique HTTP response. The filter component of pageVault sits inside the web-server's address space. It inspects each HTTP request, and if the pageVault configuration specifies that the request is of a type which should be considered for archiving, it then inspects the response. pageVault consists of four components: *filter* to intercept the calls, *distributor* to check the potential unique responses, *archiver* to archive the unique responses, s and *query servlet* to search and retrieve the archival database. Figure 7 shows the architecture diagram for pageVault.

Transaction-Time Apache (TTApache) [112] is a server side transactional archive. It is an extension of the Apache web server that supports document versioning. TTApache automatically archives

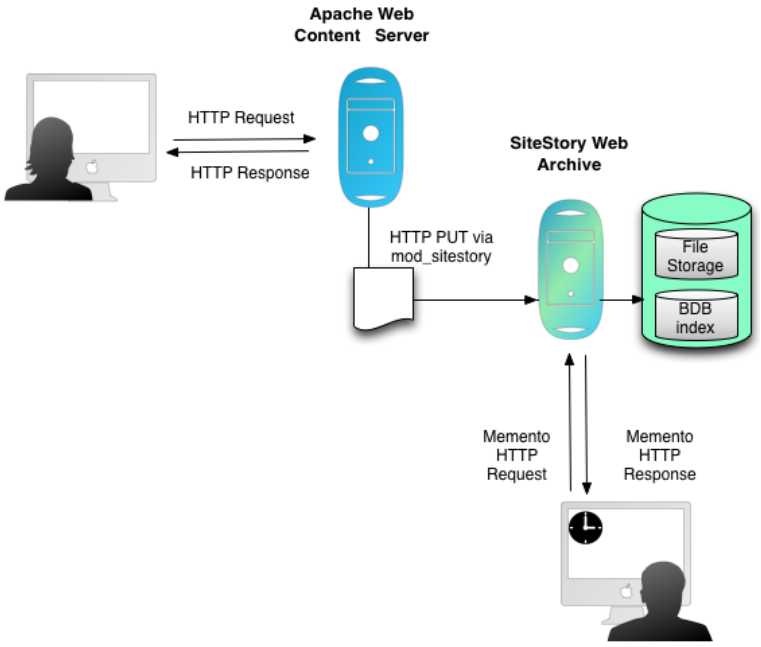


Fig. 6 SiteStory's architecture diagram (figure is taken from [55]).

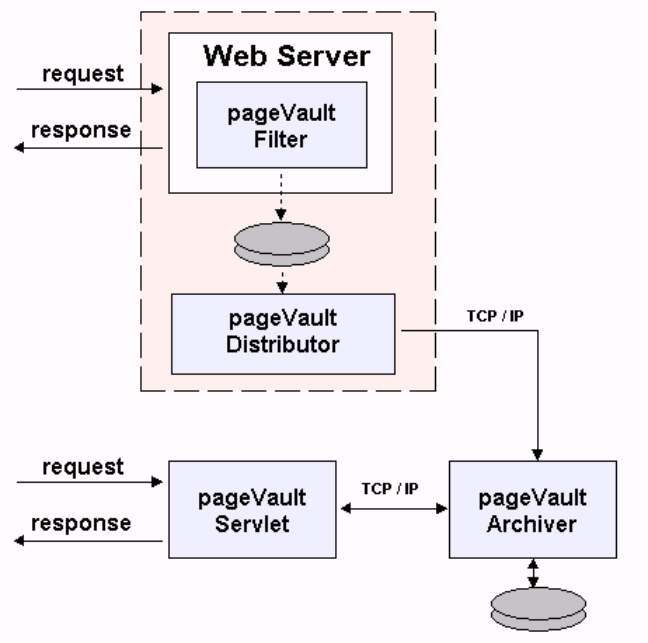


Fig. 7 Client-side TA - pageVault architecture diagram (figure is taken from [11]).

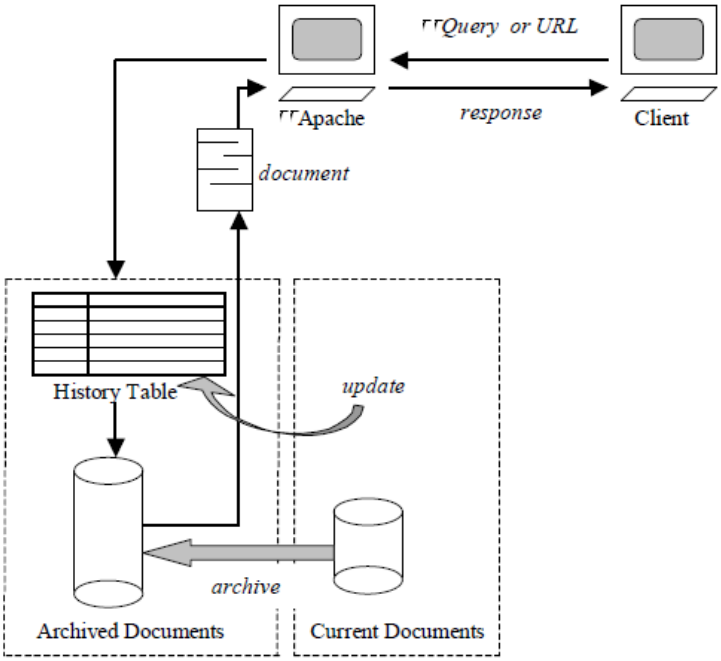


Fig. 8 Server-side TA - TTApache Server model (figure is taken from [112]).

the document version during the HTTP GETs. TTApache is fully compatible with HTTP queries, an Apache server can seamlessly migrate to TTApache server at any time without affecting anything else on the web. TTApache stores full copies, not differences between versions. Figure 8 shows the server model for TTApache.

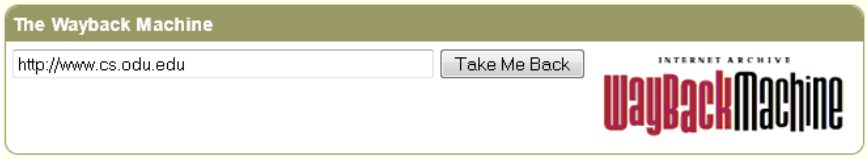
IIPC funded Live Archiving HTTP Proxy (LAP) [25]. LAP is an HTTP proxy that captures the HTTP traffic, then LAP sends the captured traffic to dedicated writers, which may be WARC or ARC formats writers.

2.4 ACCESSING THE ARCHIVED WEB

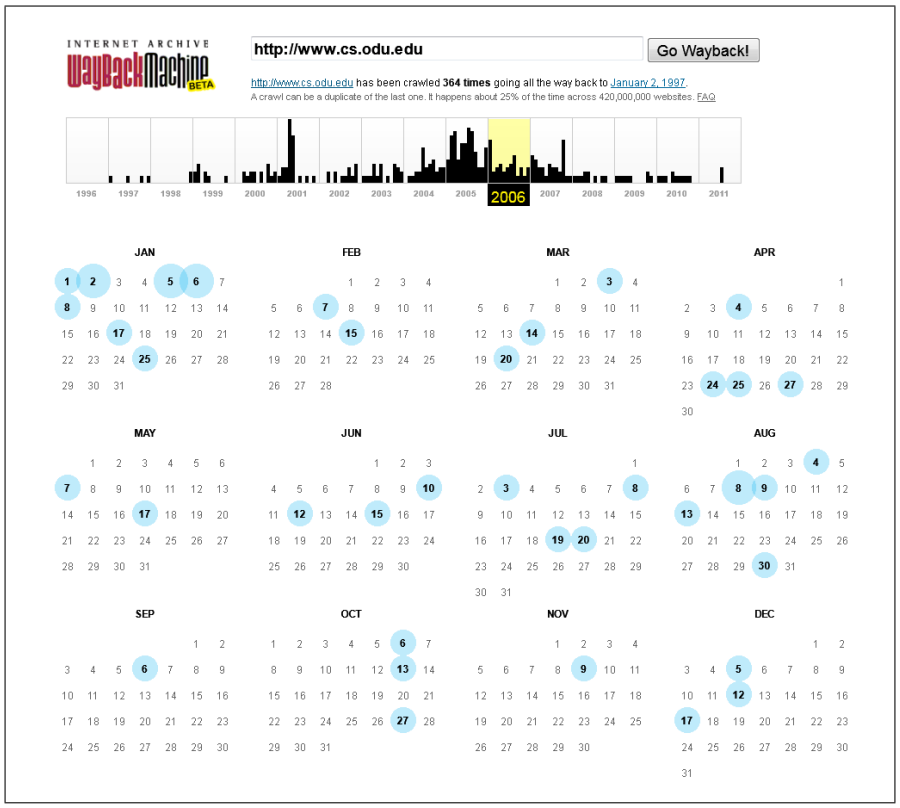
There are two main access models to the archived web content. First, the user accesses the web archive through the web interface. Second, the user use some services that have been built on the top of the archived web content. In this section, we will cover both approaches in the current web.

2.4.1 WEB ARCHIVE WEB INTERFACE

Usually the web archives provide three common user interfaces to enable the users to browse the archive content: URI-based, collection-based, and full-text search. They could be used together or separately. Internet Archive implemented the URI-based access though the "WayBack Machine". It depends on entering the URI on the text box and click on "Take Me Back" button (Figure 9(a)), then the user will get a list with the different archived copies on Internet Archive for this specific URI (Figure 9(b)). Gomes et al. [125] reported that 89% of the web archives provide URI-lookup



(a) Internet Archive Wayback Machine Query.



(b) Internet Archive Wayback Machine Results.

Fig. 9 Internet Archive Wayback Machine Interface.

interface. WebCite and the UK National Archives are another two examples of this category.

The second category is collection-based. Web pages were crawled based on specific events or topics. For example, you can browse the "September 11, 2001"[9] collection at the Library of Congress or "2011 Egyptian Revolution"[10] at Archive-It. Figure 10 shows the main page of the "September 11, 2011" collection on the Library of Congress Web Archive.

The third category is full-text search. Costa and Silva [102] found that the users preferred the full-text search to the URL-search. Gomes et al. [125] reported that 67% of the web archives supported full-text search. Costa et al. [100] surveyed the different architectures to build full-text search for the web archive. They discussed the concept of "Temporal Inverted Files" where the index is partitioned by time then by document or term; or partitioning by term or document first

[9] http://lcweb2.loc.gov/diglib/lcwa/html/sept11/sept11-overview.html
[10] http://www.archive-it.org/public/collection.html?id=2358

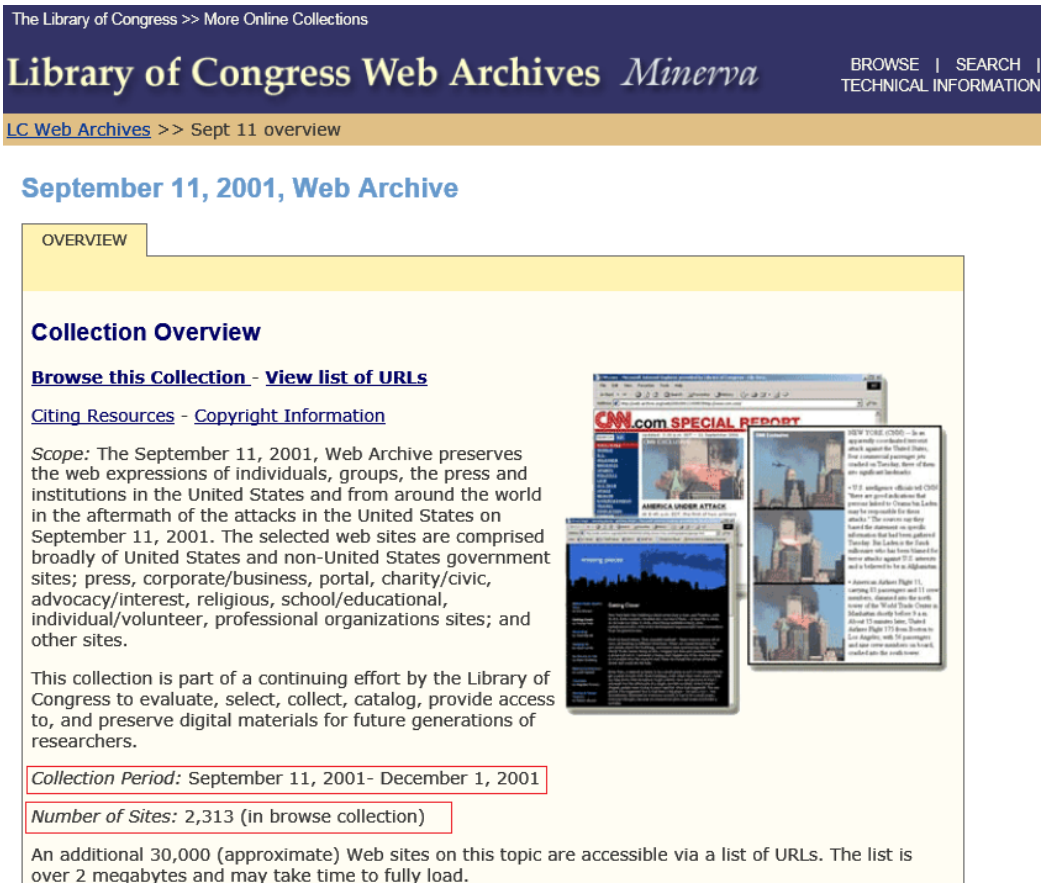


Fig. 10 September 11, 2001 collection on the Library of Congress archive

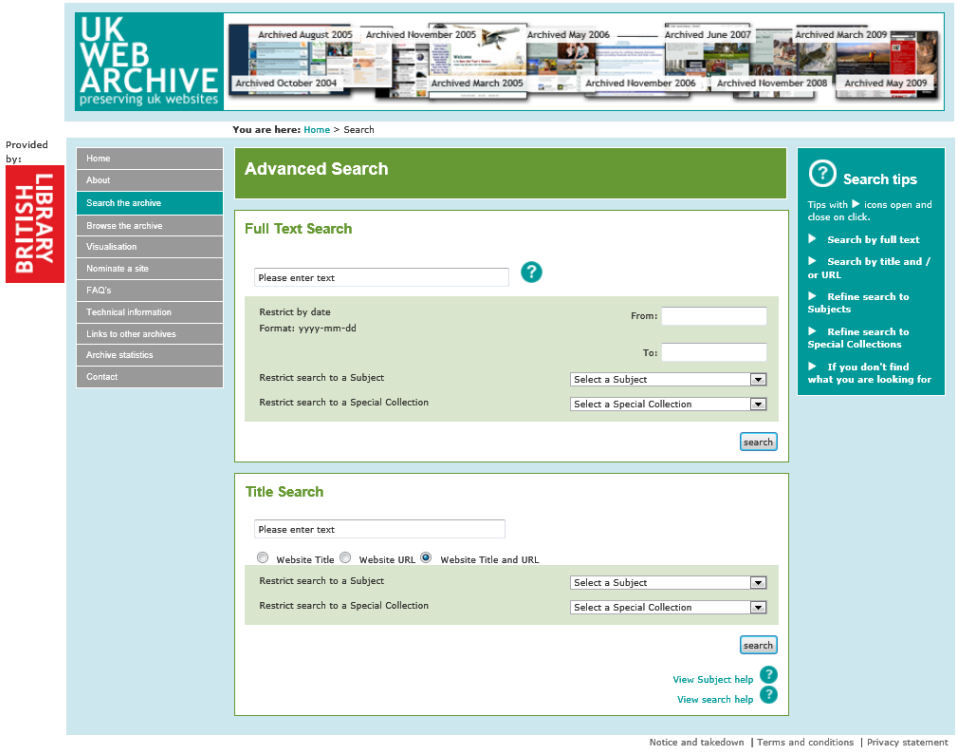


Fig. 11 UK Web Archive advanced search

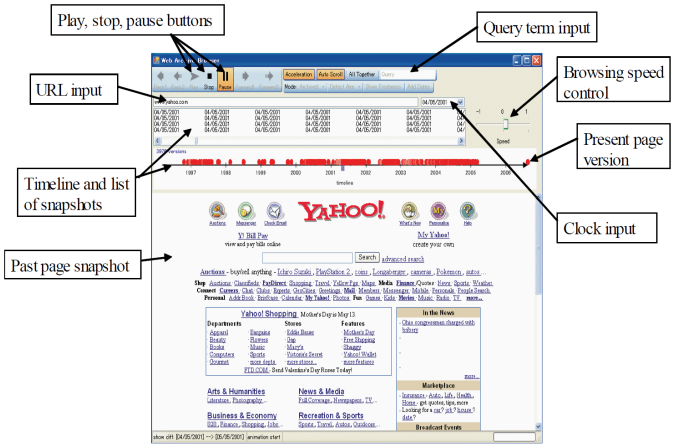
then by time.

Berberich and Anand et al. [64, 44, 43] studied building indexing methods to support text-search over web archives, e.g., "computer science" @ [06/2009]. He et al. [137, 138] proposed different techniques to compress and query temporal versioned documents. They partitioned the index into different time ranges, and access only the requested part. This technique showed better performance than they keyword-only queries. UK Web Archive[11] provides a full-text and title search in the Web Archive materials. Figure 11 shows the advanced search page in the UK Web Archive. Internet Archive does not have full-text search service due to the limitation of the computation and storage requirements because of the huge corpus that exceeds 5PB [237].

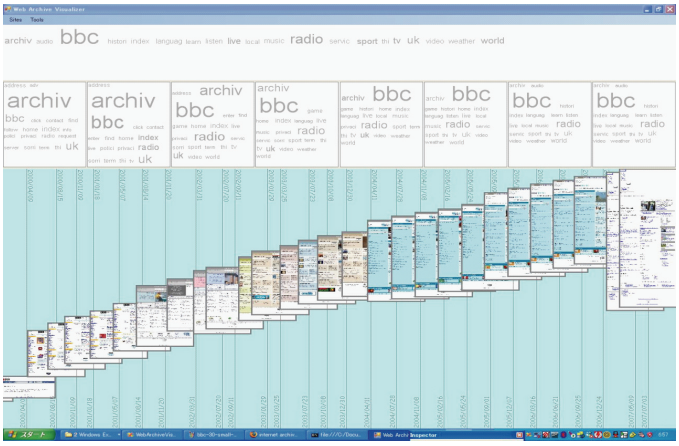
Some research has focused on exploring different archive access methods. Jatowt et al. [154, 155, 156, 157, 158] proposed different models to browse the past web. Figure 12(a) shows the "Past Web browser" that merged past snapshots from different collections and linked between the present and the past. The Past Web Browser provided the ability to move forward and backward in time. Figure 12(b) shows the history summary of a page that can generate short visual summaries of document histories. The user can select a URL and a time period for the analysis. Finally, Figure 12(c) maps the document's history to the current page by detecting the age of the page elements and displaying it to the user.

Adar et al. [33] proposed Zoetrope, a system that enables the interaction with the historical

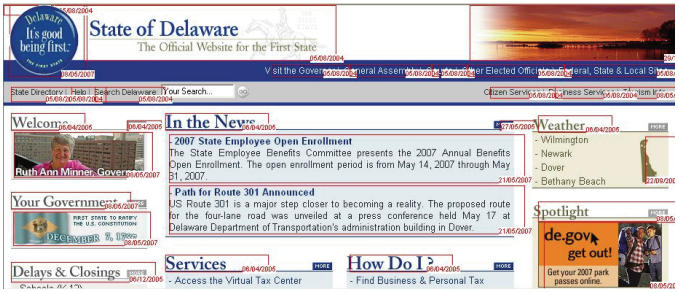
[11] `http://www.webarchive.org.uk/ukwa/advancedsearch`



(a) Interface of Past Web Browser.

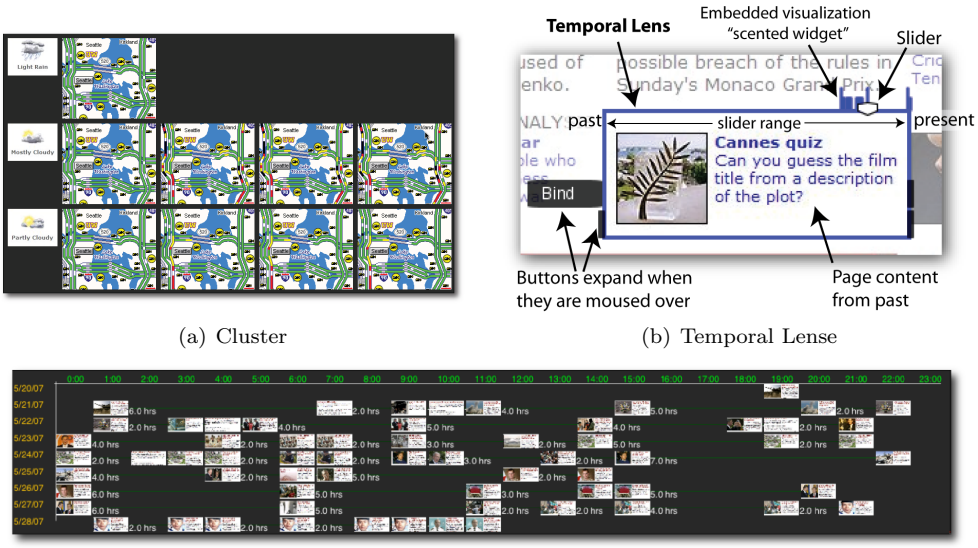


(b) Example of Page History Explorer.



(c) Example of page annotated with content-creation dates.

Fig. 12 Different Models to Interact with Document Histories (figure are taken from [156]).



(a) Cluster

(b) Temporal Lense

(c) Timeline visualization

Fig. 13 Zeotrope, different visualization techniques (figure is taken from [33]).

Web. They discussed different techniques for specifying interesting portions of the current page and visualizing the relevant historical information. Even though Zoetrope provides an easy way to browse the past, it can not be applied on the normal archives for two reasons. First, it depends on a large number of copies that were preprocessed to be ready for the user interface manipulation. Second, it depends on a special file format, so it is not compatible with WARC and Wayback Machine architecture. Figure 13 shows different visualization techniques in Zeotrope.

Teevan et al. [269] proposed DiffIE an Internet Explorer browser plug-in that cached the visited pages and highlighted how those pages have changed when the person returned to them. Figure 14 shows a snapshot of a page while DiffIE is in action. The goal is to get the users attention to the newly added content to the page to understand how the page was changed. DiffIE could be used in different ways such as: monitoring the websites (for example, stock quotes), finding expected/unexpected new content, understanding a web page, helping in editing for the publisher, and using the archived content.

2.4.2 WEB ARCHIVE SERVICE INTERFACE

Unfortunately, most of the current web archives do not provide APIs to access its data, and limit their interfaces to GUIs only. In Section 2.4.1, we gave some examples about the user interfaces for web archives and the different actions that the user could perform (e.g., search by URI, search by word, browse collection). In this section, we will discuss the current web archives' APIs.

In addition to the web interface, the Wayback Machine has various query interfaces which provide APIs interface [237, 31]. For example, Wayback Availability JSON API tests if the URI is archived and accessible on the Wayback Machine or not.

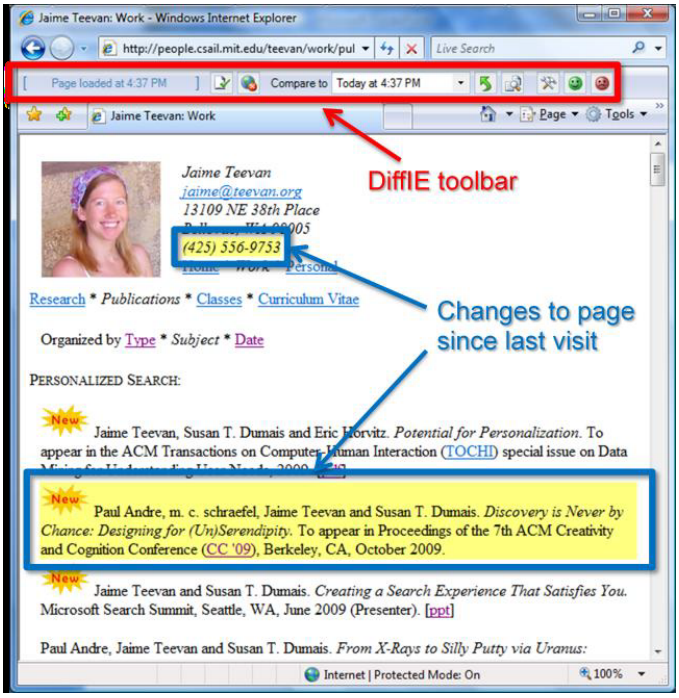


Fig. 14 DiffIE in action. Changes to a publication page since the user previous visit are highlighted (figure is taken from [269]).

```
> curl http://archive.org/wayback/available?url=cs.odu.edu
{
  "archived_snapshots":{
    "closest":{
      "available":true,
      "url":"http://web.archive.org/web/20140116122050/http://www.cs.odu.edu/",
      "timestamp":"20140116122050",
      "status":"200"
      }
   }
}
```

Wayback CDX Server API allows the user to retrieve a complete list of the CDX records for specific URI.

```
> curl http://web.archive.org/cdx/search/cdx?url=cs.odu.edu&output=json&limit=3
[
  ["urlkey","timestamp","original","mimetype",
     "statuscode","digest","length"],
  ["edu,odu,cs)/", "19970102130137", "http://cs.odu.edu:80/", "text/html",
     "200", "QLH32HMLFUSZ3VSSFWPNKTWSKC4LFSMY", "377"],
```

```
  ["edu,odu,cs)/", "19970606105039", "http://www.cs.odu.edu:80/", "text/html",
     "200", "VWXQ7BORGXNHDL4DL3HSAKJ7ANWNTITZ", "627"],
  ["edu,odu,cs)/", "19971010201632", "http://www.cs.odu.edu:80/", "text/html",
     "200", "VWXQ7BORGXNHDL4DL3HSAKJ7ANWNTITZ", "625"]
]
```

The Memento API could be used to get the same list by using a Memento Timemap. Listing 3 shows the Memento Timemap for `http://a.example.org`. Memento implementation has been integrated in Wayback Machine since release 1.6 [21].

Portuguese Web Archive developed [27, 124] OpenSearch API to query the full-text interface for the Portuguese Web Archive and to receive the response in XML-based file. UK Web Archive [32] proposed the "UK Web Archive Open Data" which is a set of datasets, tools, and APIs to access archived web data. Due to the size and legal issues, UK Web Archive gives access to metadata samples only instead of the actual data.

Warrick

Warrick [84, 202, 206] is a tool for recovering a website from the available public web repositories. Warrick 1.0 covered four Web repositories: Internet Archive, Google, Live Search (now Bing), and Yahoo, which were entitled Web Infrastructure (WI). Warrick 2.0 incorporated the Memento protocol to retrieve a complete TimeMap from the Memento Aggregator. Warrick helps the web administrators to retrieve their websites where there is no back-up available. The recovering process starts with a seed URI for the lost website and searches through the available repositories for any available copy. The most recent copy is stored, then Warrick starts with extracting the links and embedded resources from the saved copy, then repeating the process to recover the rest of the website.

Synchronicity

Synchronicity [171, 172] is a Mozilla Firefox add-on that enables the user to (re-)discover his missing web pages. Synchronicity used the Memento protocol to get the latest available memento and used it to build lexical signature for this URI. Then, Synchronicity will use Yahoo APIs to query with this lexical signature to search for a similar content.

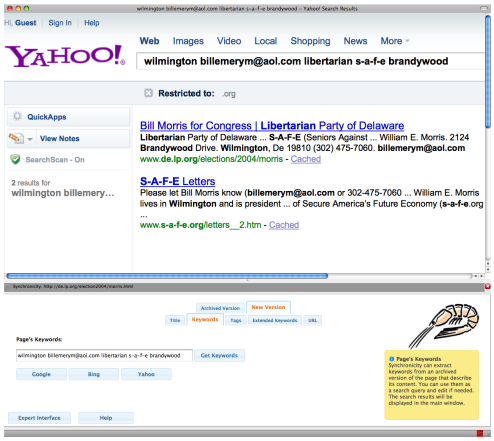
Figure 15 shows applying Synchronicity on a missing page (Figure 15(a)). Figure 15(b) shows the latest memento of the page and the generated timeline with the dates of all mementos in the bottom of the page. The generated lexical signature of the latest Memento is queried against Yahoo! (Figure 15(c)). And finally, Figure 15(d) shows the new URI of the missed page with the same content.

2.5 THE QUALITY OF THE WEB ARCHIVE

The goal of any web archive is to preserve the web page history for future generations. In an ideal world, the web archive should be able to take a complete snapshot of the website when any of the pages have been changed. Of course, this is not the case. Thus, we could define the "Quality of Web

(a) Lost at: www.de.lp.org/election2004/morris.html

(b) Memento from April 2005



(c) Query with extracted keywords

(d) Discovered Page at its New URI www.de.lp.org/elections/2004/morris

Fig. 15 Synchronicity's discovery process (figures are taken from [172]).

Archive" as "How much the archived website was similar to the original site at t_i?". To quantify this definition, we are using three main measurements: *Completeness*, *Coherence*, and *Coverage*.

2.5.1 COMPLETENESS

As introduced in Section 2.3.1, the completeness of the archived website is defined as the amount of useful page versions from the website by the archive [62, 195]. The completeness could be measured in two ways: horizontal and vertical [195]. Horizontal completeness is measured by the number of relevant entry points found within the designated perimeter. The horizontal completeness is required in site-oriented archiving. Figure 16(a) shows a graphical explanation of the extensive collection to increase the horizontal completeness. The vertical completeness is measured by the number of relevant linked nodes found from this entry point. Figure 16(b) shows an illustration for the intensive collection to increase the vertical completeness.

2.5.2 COHERENCE

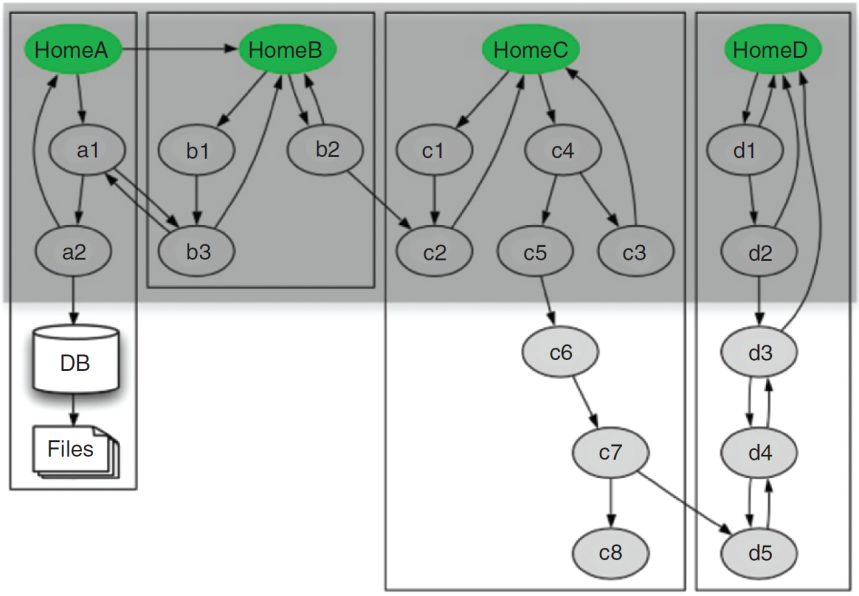
The coherence is the ability to render the original form of the site. Spaniol et al. [259] defined coherence as "Web archiving coherence has a temporal dimension: contents are considered to be coherent if they appear to be as of time point x or interval [x; y]". Ben Saad et al. [62] defined coherence as "the ability of the archive to reflect the states (or snapshots) of a web site at different points in time". Figure 17 shows an example of the coherence defect. Figure 17(a) is the Computer Science Department homepage[12] at Old Dominion University as it appeared in the Internet Archive on date June 23, 2003. If you click on the featured student link for "Chutima Boonthum", the Internet Archive will redirect you to "Alaa Youssef" featured alumni page (Figure 17(c)) which definitely is not the required page. The problem happened because these two snapshots were taken in two different dates then linked together. So the current archived navigation behavior does not reflect the real behavior of the website on this date, this is what is called "coherence defect".

Spaniol et al. [259, 107] proposed the SHARC Framework for assessing the data quality in the web archives and for tuning capturing strategies toward coherence and completeness quality measures. The framework proposed two phases, phase one "Single-visit" crawls download every page in a site exactly once to cover the whole site (completeness). Phase two "Visit-revisit" revisit the pages after the initial download to cover any intermediate changes during the first crawl. The aim of the second phase is to maximize the coherence of the site. The coherence defects could be detected using different visualization techniques [259]. Figure 18 shows a coherence defect visualization of a single crawl-recrawl pair of the `mpi-inf.mpg.de` (about 65,000 Web pages). Depending on the nodes' size, shape, and color the user gets an immediate overview on the success or failure of the capturing process.

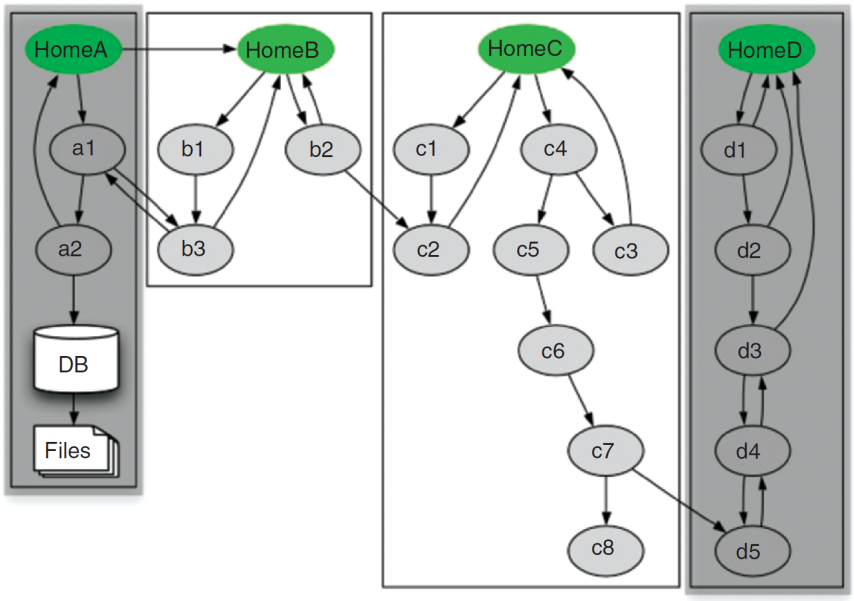
2.5.3 COVERAGE

The coverage quality measure [244] of a list of archived copies describes how accurately the archived copies reflect the important versions of a web page in a time interval. Also, the coverage

[12] `http://www.cs.odu.edu`



(a) Extensive archiving to support the horizontal completeness.

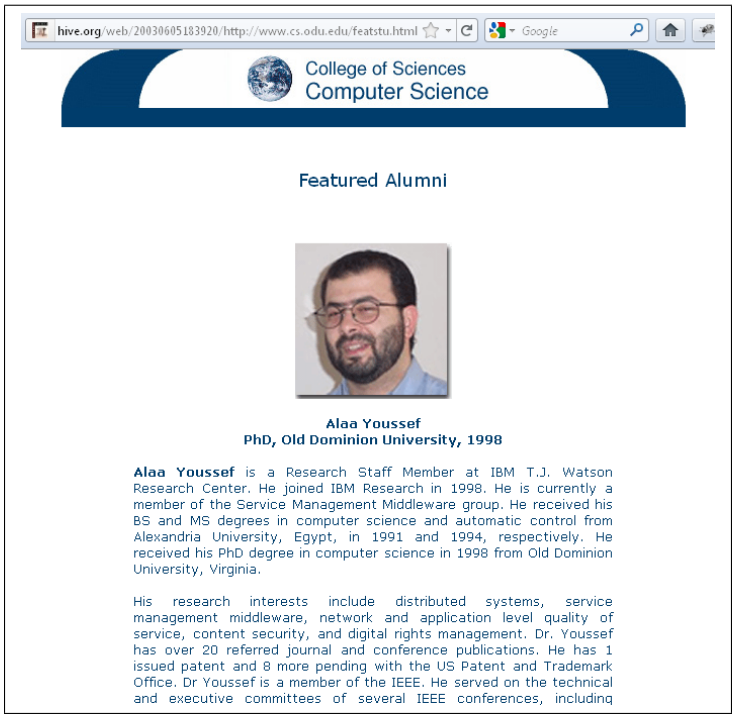


(b) Intensive archiving to support the vertical completeness.

Fig. 16 Different Completeness Archiving levels, the shaded area is the archived content (figure is taken from [197]).

(a) CS-ODU Home Page, then click on "Chutima Boonthum" link.

`http://web.archive.org/web/20030623201227/http://www.cs.odu.edu/`



(c) Internet Archive forwards me to "Alaa Youssef" page.

`http://web.archive.org/web/20030623201227/http://www.cs.odu.edu/`

Fig. 17 Coherence Defect Example in www.cs.odu.edu.

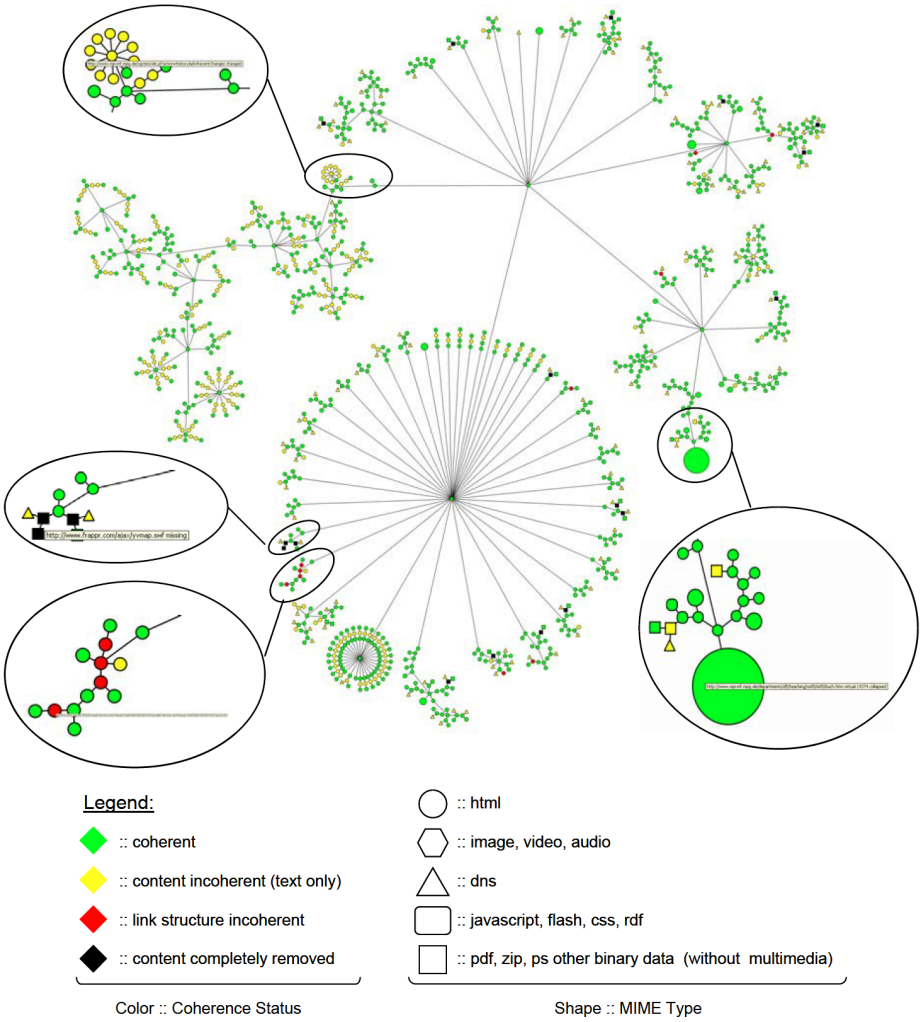


Fig. 18 Coherence defect visualization of a single crawl-recrawl pair of mpi-inf.mpg.de (figure is taken from [259]).

may differ from the common freshness concept of search engine techniques [94] in that it is limited to the important changes [244]. Ben Saad et al. [61, 62] provided different techniques to adapt the web crawler for the Web Archive to estimate and capture the important changes only.

Hence, most of the quality studies focused on the quality from the archivists perspective by proposing new crawling strategies to enable the web archive to fulfill three measures: completeness, coherence and coverage. To increase the completeness, the breadth-first search crawling [214] could be used. Gomes et al. [127] suggested a technique to detect the duplicates pages before storing in the archive. Increasing the coherence has been discussed in [258, 107], and increasing the coverage has been studied in [94]. However, the previous techniques provided mechanisms to assure pre-crawling quality measures, few techniques covered post-crawling and post-delivery quality measurements from the user perspective. Brown [81] suggested post-collection testing steps to ensure the quality of the collected pages.

2.6 SUMMARY

In this chapter, we discussed the main aspects about web archiving. We focused in the archiving process on four steps: selection, acquisition, access, and quality. The selection process described the scope of the web archive. In the acquisition process, we listed various techniques such as: remote harvesting, database archiving, and transnational archiving. We explored different access methods and applications for web archives. Mainly there are two categories, web interface and APIs. The web interface depends on three methods, URI-lookup, full-text search, and collection browsing. APIs are limited to few and non-standards APIs such as Wayback APIs and Memento.

This chapter explained the current challenges in web archiving. It emphasized on the limitation of the access methods, especially non-standards and rich APIs that are the main focus of this research.

CHAPTER 3

MEMENTO FRAMEWORK

Memento [7, 215, 216, 278, 279, 280, 282] is a protocol-based solution which aims to achieve a tighter integration between the current and the past web. Memento is an extension for the popular HTTP to allow the user to browse the past web as the current web. Memento extends HTTP content negotiation [145] to include the datetime dimension.

3.1 MEMENTO NOTATIONS AND HEADERS

The Memento framework depends on a set of new notations [278]:

- $URI - R$, or R, is used to denote the URI of an *Original Resource* as it appeared or used to appear on the live web.
- $URI - G$, or TG, is used to denote the URI of a *TimeGate*. A TimeGate is the resource that has the negotiation capability to select of the representation of the URI as it appeared in the closest to the requested datetime.
- $URI - M$, or M, is used to denote the URI of a *Memento*. A Memento is a representation of the original resource as it appeared in the past. We denote M_i as $M_i = M(R)$ *at* t_i.
- $URI - T$, or TM, is used to denote the URI of a *TimeMap*. A TimeMap is a list of the available mementos of the original resource:

$$TM(R) = \{M_1, M_2, \ldots M_n\}$$

The Memento Framework [278] introduces two new headers ("Accept-Datetime", "Memento-Datetime") and reuses two existing headers ("Vary", "Link"):

- **Accept-Datetime** request header that is used by a user agent to indicate the datetime to retrieve the memento of the original resource near this time.

  ```
  Accept-Datetime:  Thu, 31 May 2007 20:35:00 GMT
  ```

- **Memento-Datetime** response header is returned by a server to declare the datetime when the memento was captured.

  ```
  Memento-Datetime:  Wed, 30 May 2007 18:47:52 GMT
  ```

- **Vary** response header is used by the server to indicate that content negotiation happened in the datetime dimension.

  ```
  Vary:  accept-datetime
  ```

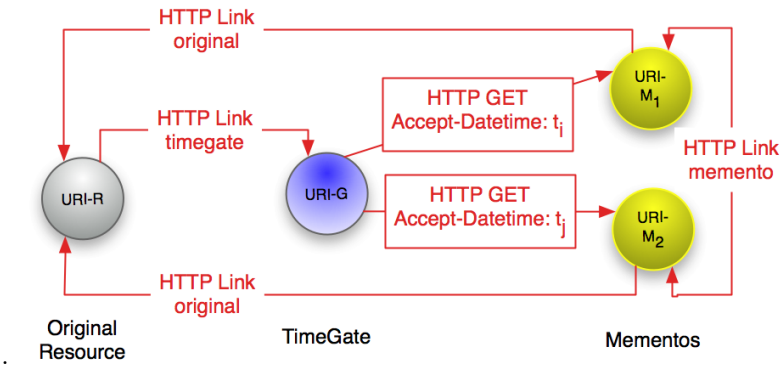


Fig. 19 The Memento Framework (figure is taken from [282]).

- **Link** response header is updated with new relation types to expose links among original resources, TimeGates, mementos, and TimeMaps. Listing 1 shows an example of Link Header response for $URI-M$ as defined in the Memento protocol. The "Link" header carried information about the $URI-R$, TimeMap URI with the format, and first, last, next, and previous memento to this $URI-M$.

Figure 19 illustrates the Memento framework. Only the *TimeGate* is capable of negotiation with the web browser in the time dimension to allow selective, datetime-based, access to an archived copy of the requested URI. Multiple TimeGates may exist for any given resource.

Listing 1 Memento Protocol Link Header Example.

```
Link: <http://a.example.org>; rel="original",
 <http://arxiv.example.net/timemap/http://a.example.org>
   ; rel="timemap"; type="application/link-format",
 <http://arxiv.example.net/web/20000915112826/http://a.example.org>
   ; rel="first memento"; datetime="Tue, 15 Sep 2000 11:28:26 GMT",
 <http://arxiv.example.net/web/20080708093433/http://a.example.org>
   ; rel="last memento"; datetime="Tue, 08 Jul 2008 09:34:33 GMT",
 <http://arxiv.example.net/web/20010911203610/http://a.example.org>
   ; rel="memento"; datetime="Tue, 11 Sep 2001 20:36:10 GMT",
 <http://arxiv.example.net/web/20010911203610/http://a.example.org>
   ; rel="prev memento"; datetime="Tue, 11 Sep 2001 20:30:51 GMT",
 <http://arxiv.example.net/web/20010911203610/http://a.example.org>
   ; rel="next memento"; datetime="Tue, 11 Sep 2001 20:47:33 GMT"
```

3.2 MEMENTO TIMEMAP

As previously defined in Section 3.1, we can consider the TimeMap as an inventory of mementos for an original resource that the responding server is aware of. It lists at least:

- URI of original resource.

  `<http://a.example.org>;rel="original",`

- URI and datetime of all known Mementos.

```
<http://arxiv.example.net/web/20010911203610/http://a.example.org>
    ; rel="memento"; datetime="Tue, 11 Sep 2001 20:36:10 GMT",
```

- URI of TimeGate.

```
<http://arxiv.example.net/timegate/http://a.example.org>; rel="timegate",
```

- URI of the TimeMap itself.

```
<http://arxiv.example.net/timemap/http://a.example.org>
    ; rel="self"; type="application/link-format",
```

  In a paged TimeMap where the available mementos are divided between multiple TimeMap, additional attributes could be added to the TimeMap URI, such as `from` and `until` that express the temporal span of mementos listed in each TimeMap.

```
<http://arxiv.example.net/timemap/1/http://a.example.org>
   ; rel="self";type="application/link-format"
   ; from="Tue, 20 Jun 2000 18:02:59 GMT"
   ; until="Wed, 09 Apr 2008 20:30:51 GMT",
 <http://arxiv.example.net/timemap/2/http://a.example.org>
   ; rel="timemap";type="application/link-format"
   ; from="Thu, 10 Apr 2008 20:30:51 GMT"
   ; until="Tue, 27 Oct 2009 20:49:54 GMT",
```

There are different possible TimeMap serializations such as application/link-format, introduced in CoRE Link Format [248], that is the default implementation of the TimeMap. In Listing 3, lines 1-4 are the HTTP request headers to retrieve the TimeMap for $URI - R$. Lines 7-15 are the HTTP response headers. Lines 17-30 are the TimeMap content.

3.3 MEMENTO AGGREGATOR

A TimeMap Aggregator [243] (for simplicity, Aggregator) harvests and merges TimeMaps; the Aggregator exposes its own TimeGates and TimeMaps. The Memento Aggregator provides a TimeGate and TimeMap from various archives with finer datetime granularity after merging the different TimeMaps. It can become a shared target for redirection for many web servers that do not have a ready TimeGate. The Memento Aggregator depends on a set of Memento complaint archives such as Internet Archive, Archive-It, UK Web Archive, UK National Archive, and Archive.is. For the non-compliant archives, the Memento Aggregator uses scripts [239] that can be used to implement by-proxy Memento support for third-party servers such as web archives and content management systems. Figure 20 illustrates the Aggregator relatives to the different TimeGates. The original resource (R) redirects the request with Accept-Datetime header to the Aggregator TimeGate. The

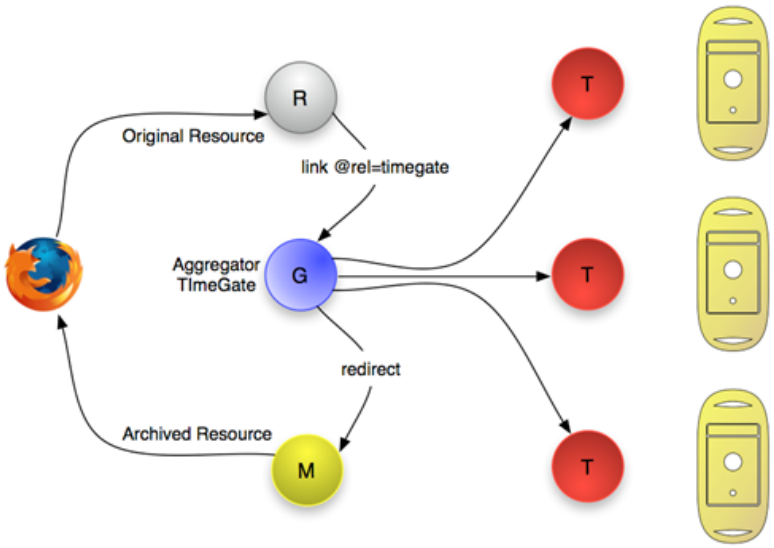


Fig. 20 Memento Aggregator (figure is taken from [281]).

Aggregator will query different TimeMaps to find the nearest memento to the requested date/time. Then, it will redirect the request to the selected memento. The Aggregator tasks for merging and selection are completely transparent to the user-agent.

In practice, we have two active Aggregators: at the Computer Science Department at Old Dominion University (CS-ODU)[1] and Los Alamos National Lab (LANL)[2]. Each one of these Aggregators merges different TimeGates for different archives. Appendix A lists the different TimeGates/Proxies in the current Aggregator implementation.

3.4 MEMENTO CLIENTS

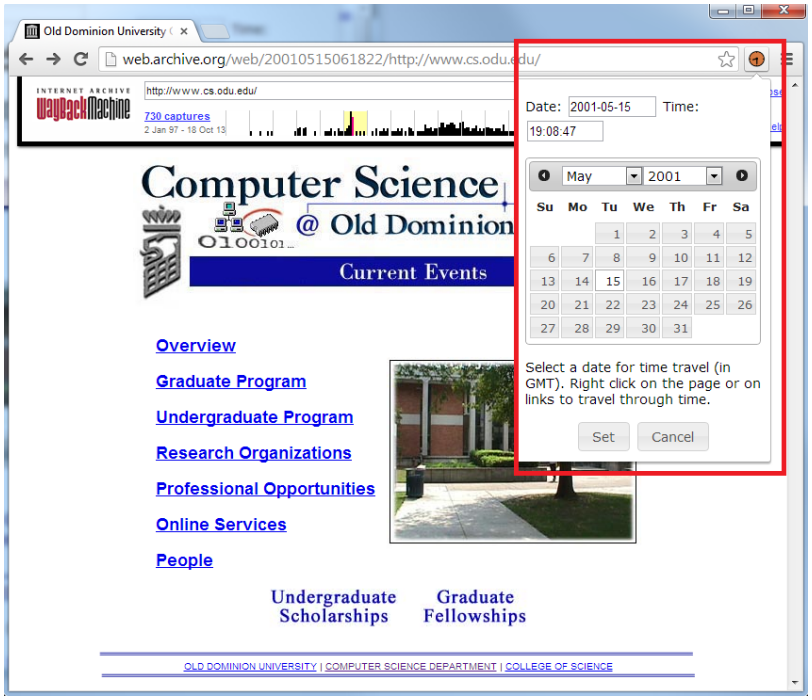
In order to fully benefit from the Memento Framework, it was important to have a specialized Memento Client which will be able to use the Memento request/response headers. *MementoFox* [242, 241] is a Firefox addon that implements the Memento protocol that links resources with their previous versions automatically, so can you see the web as it was in the past. MementoFox created a toolbar for user interaction to select the date in the past to travel to. Firefox provides different events and methods to monitor and change HTTP requests. MementoFox's toolbar allowed the user to enable and disable Memento, specify the target date and time, and view the results of Memento operations. Figure 21(a) shows a screenshot of MementoFox.

Memento for Chrome [247] is Chrome extension that works as Memento client. It has a datetime picker to set the preferred date/time in the past. The Memento extension updates the quick list (i.e., the list that appears when you right-click on item) with "Get near" option to navigate the user to the page as it appeared on the requested date.

There are two Memento clients for mobile devices. For the Android operating system, *Memento*

[1] http://mementoproxy.cs.odu.edu/aggr/timegate/{URI}
[2] http://mementoproxy.lanl.gov/aggr/timegate/{URI}

(a) MementoFox - Firefox addon.



(b) Memento for Chrome extension.

Fig. 21 Screenshot for www.cs.odu.edu in 2001 from Internet Archive..

Fig. 22 Screenshots of Andriod App:Memento Browser (figure is taken from [201]).

Browser [242, 201, 277] was released as an open source project on Google Code in September 2010 [18]. In Memento Browser, the user has the ability to select a new date, and the Memento TimeGate will return the nearest Memento to this date. Also, it lists all the available Mementos by selecting the "List Mementos" option. Figure 22 shows screenshots for the Memento Browser. For iOS, Memento Browser [26], provides time travel mode for both iPhone and iPad. Figure 23 has a screenshot for Memento Browser on iPhone.

Finally, `mCurl` [38] is a command-line memento client. mCurl is a wrapper for the UNIX `curl` command that is capable of doing content negotiation in the datetime dimension with Memento TimeGates. mCurl supports all curl parameters in addition to the new parameters that are Memento related.

3.5 SUMMARY

Memento protocol provides the basic mechanism for the communication between the present and the past. Memento proposes new resources such as TimeGate and TimeMap that facilitate the integration between the original URIs and its recorded snapshots from the past (mementos). The easy of use for Memento APIs supports the development of third-party clients that help the user to browse the past web easily. We use Memento terminology through the dissertation. In this research, we have the same assumption of having enrich APIs will facilitate the creation of new web archives application that may attract more users.

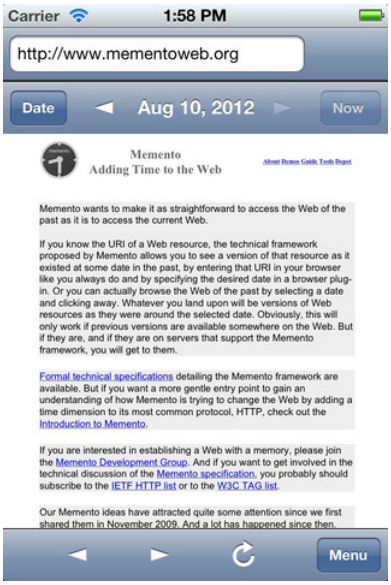


Fig. 23 Screenshot of Memento Browser for iPhone (figure is taken from [26]).

CHAPTER 4

WEB ARCHIVING SERVICES FRAMEWORK

In the previous chapters, we explained the current trends of the web archiving and the limitation of its access layer. In this chapter, we will discuss our solution to enable a service framework to provide enriched access to the archived web. Section 4.1 gives an overview about the service framework levels, then it summarizes the four-level framework: *Content*, *Metadata*, *URI*, and *Archive*. Section 4.2 discusses the datasets that are used in this research. Appendix C has a list of the possible services that could be useful for the web archive supported with a summary for the input, output, and main challenges to implement these services.

4.1 WEB ARCHIVES SERVICE MODEL

The IIPC [3] was formed in 2003 to improve the tools, standards, and best practices of web archiving. Even though they formed standards for web archiving activities (for example, the WARC format became an international standard in 2009 [133]), they did not address a method for communication between different web archives. The current web archive architecture does not support rich APIs, as described in Section 2.4.2, with only a limited set of APIs available to return the TimeMaps in non-standard formats. Even when APIs exist, it is not standardized between the various web archive entities. In this section, we will define the main parts of the web archive in order to build a suitable service for each part.

We will look to the web archive as four layers. Figure 24 shows an example of an archived web page that could be represented in these four layers.

1. *Content*: This is the basic level of the model and it contains the raw data of the crawled web pages. The data could be text, images, videos, or other MIME types [120].

2. *Metadata*: It describes a structured way of storing and accessing metadata. In this level, the web archive exposes some information about the archived web page such as title, thumbnail, content-length, etc. We focus on the derived metadata, which requires additional processing and analysis to be created more than the extracted metadata that can be read directly from the resource content.

3. *URI*: Unlike most web services, most large web archives do not have keyword search and use the URI as the primary key for access.

4. *Archive*: This is the most abstract layer of the archived web page when you look to the archive itself. This level describes the main characteristics that distinguish each web archive from the others such as age and supported top-level domains.

In this research, we build *ArcSys*, the name is borrowed from IIPC Access Working Group report [148]. ArcSys is a service framework that integrates with the current web archive architecture to

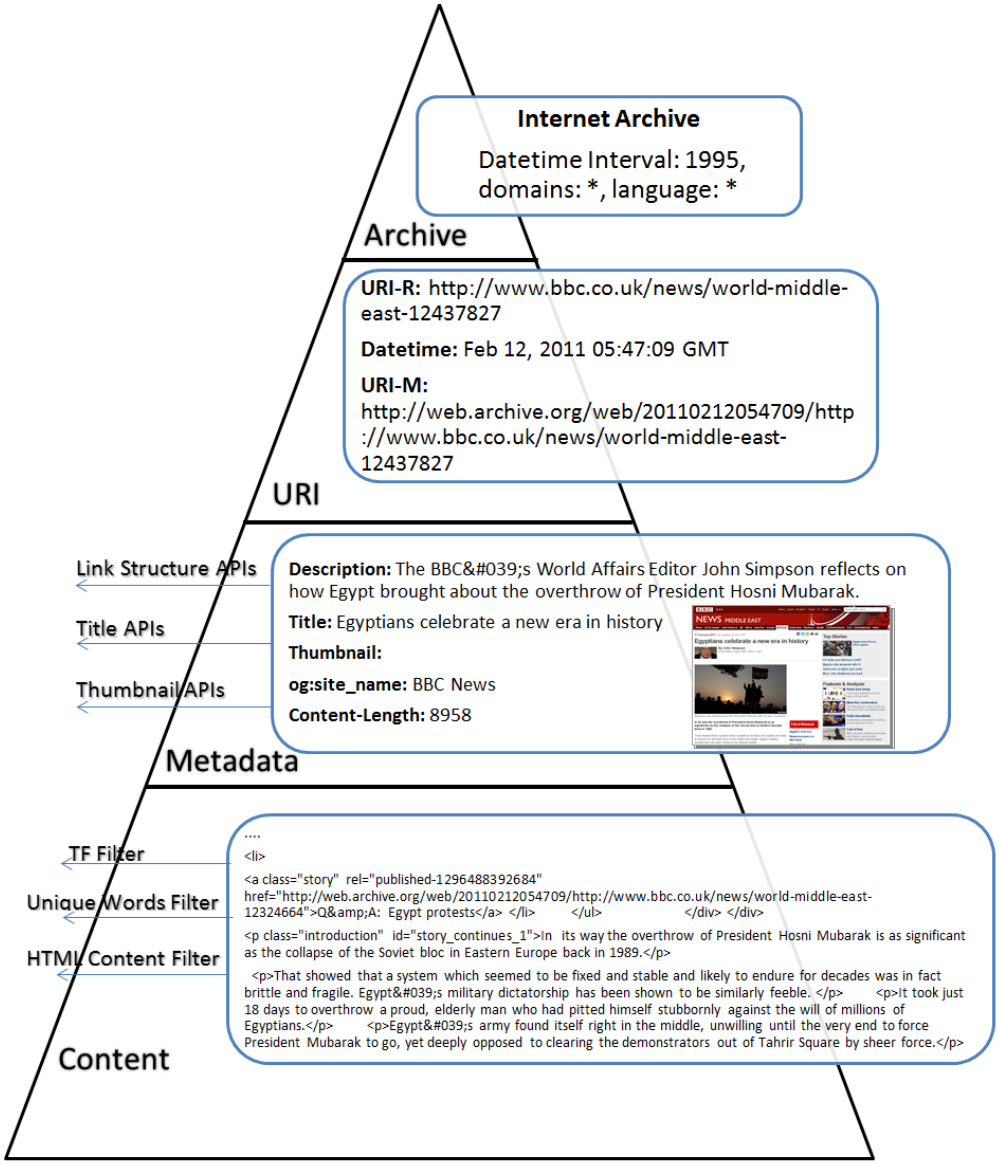


Fig. 24 Example services and metadata for each of the four levels.

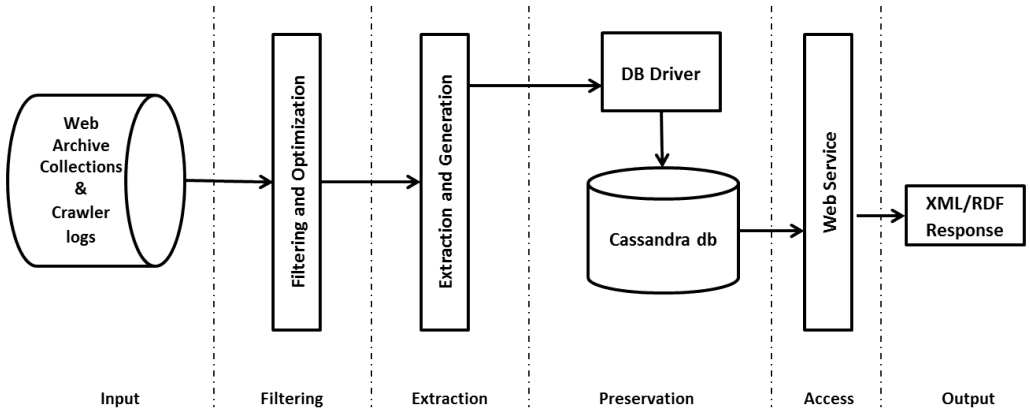


Fig. 25 ArcSys Architecture Diagram

expose APIs for the web archive collections. ArcSys is a set of integrated services that can be deployed separately. ArcSys depends on big data techniques so it can be extended for the large-scale web archive. In this research, we develop a set of subsystems for ArcSys:

- *ArcContent* works on the content layer and exposes APIs for web archive collection content.
- *ArcLink* works on the metadata layer and exposes APIs for the temporal web graph.
- *ArcThumb* works on the metadata layer and exposes APIs for generating thumbnail images for web archive collections.

We support ArcSys with two query optimization mechanisms:

- Using the URI HTTP redirection in the web archive.
- A profiling mechanism to describe and select the web archive based on the request characteristics.

Figure 25 shows the general architecture for the ArcSys system. The subsystem usually will follow the same stages. More details about each subsystem will be available in the next chapters. Generally, ArcSys accepts $URI - R$ and $URI - M$ as request parameters. ArcSys does not accept term or keyword queries. Table 2 lists a set of available request parameters accepted by ArcSys.

In the following sections, we will explain each service layer with a general discussion about the challenges for these services.

4.1.1 CONTENT ACCESS SERVICE

The content service is responsible for retrieving the actual content of the page as it appeared in the past. The content may be the whole representation of the resouce (e.g., HTML, video, audio) or it may be modified or extracted content of the resource.

In this research, we extend the content access service into three main response types: Raw, which is the raw representation as crawled; Modified, which is the raw representation after modifying it

Table 2 ArcSys request parameters.

Parameter	Description
URI-R	the original URI as it appeared (or used to appear) on the live web.
URI-M	a memento URI, a snapshot of URI-R as it used to appear in the past.
Datetime	a datetime in the past which the user is interested to view the web page as it appeared in this datetime.
DateInterval	an interval of time in the past. It could be defined by start-date and end-date, or datetime with boundaries (e.g., one month before and after Nov 3, 2001).
ResponseType	defines the format of the returned data from the web archive (e.g., JSON, XML, or CoRE).
Site	A website with all pages.
CollectionId	an ID for specific collection in the web archive.
Set of URIs	a list of distinct URIs.
Domains	a list of one or more domains (e.g., .com, or .fr).
Hostname	indicates a hostname for specific URI-R.

for better display experiences; and Extracted, which is extracted information from the raw content. We focus on the extracted response by applying filters that extract textual content from the HTML resources such as term frequency.

4.1.2 METADATA ACCESS SERVICE

Metadata services expose the metadata about the archived web pages through an API. This information may include title, HTTP response code, response headers, description, and keywords.

Currently, the Internet Archive is working on a new standard for metadata extraction, entitled Web Archive Metadata (WAT) file [123]. WAT utilities are used to extract metadata from WARC files. The structure of the extracted metadata is optimized for data analysis. WAT data can be used to efficiently create data analysis reports based on large datasets. Generally, the WAT specification is optimized for the mass computation of the web archive metadata but is not suitable for retrieving fine-grained information. We will discuss the limitations of WAT files in detail in Chapter 6.

The metadata service could enhance the web archive interface by adding extra information for the snapshot in addition to the capture datetime. For example, we can build a TitleMap that includes the title of the memento as it appears in Figure 26 or a Thumbnail view that includes the thumbnail of the memento as it appears in Figure 27.

The main challenge in extracting the metadata from the archived web page is the diverse techniques required in extracting each item. For example, extracting the title requires extracting title tag or checking the title field of the meta header. Where as taking a thumbnail of the page requires rendering the whole page with all the embedded resources to take a snapshot of the page.

We focus on two types of derived metadata, link structure and thumbnail creation. In link structure metadata, we develop ArcLink that can extract, preserve, and deliver the web graph for the web archived data. ArcLink focuses on the inlinks and outlinks of the archived web pages. In thumbnail creation, we develop ArcThumb that optimizes the creation of the thumbnails for the archived web pages by inspecting the HTML features such as SimHash fingerprint.

Fig. 26 TitleMap for vancouver2010.com.

Fig. 27 iTune cover prototype application powered by ArcThumb APIs for apple.com.

4.1.3 URI ACCESS SERVICE

In Section 2.1.1, we defined URI as a mechanism to identify the resources across the web. Gomes et al. [125] show that 89% of web archives provide the URI-lookup feature.

For Heritix-based crawler archives, CDX files are an index of the crawled URIs. CDX files consist of individual lines of text, with each line summarizing a single memento. The first line in the file is a legend for interpreting the data, and the following lines contain the data for referencing the corresponding mementos within the web archive. A record from a CDX file is shown below:

```
example.org/foo.html  20090312223142 http://www.example.org/foo.html
   text/html 200 E6T72C2R6BRRKSI3IZPMRJDXTFJIRC7P - - 111739
   TENN-000001.warc.gz
```

The URI access service is interested in the following fields from the CDX file:

- $URI-R$ (http://www.example.org/foo.html): the original URI as it appeared on the live web.
- Timestamp (20090312223142): the timestamp (YYYYMMDDHHMMSS) when the memento was captured.
- HTTP Status (200): The HTTP response received from the web server for the $URI-R$ at this timestamp.

Table 3 IIPC 2010 Winter Olympics Collection.

Size	700+GB
Number of WARC files	1950
Number of Seed URI-R	302
Number of Unique URI-R	6.4M
Number of URI-M	23.7M
Start date	Nov 3, 2009
End date	Mar 13, 2010

In this research, we focus on the URI-lookup mechanism in the presence of HTTP redirection. We propose two dereference techniques for the $URI - R$ with redirection on the live web and for the $URI - M$ with redirection on the archived web.

4.1.4 ARCHIVE ACCESS SERVICE

The global archived web corpus is distributed between various web archives around the world. Every archive has its own policy to crawl and preserve the web The archive access service is responsible for differentiating each web archive from the others based on the content holding characteristics. The characteristics include the age of the oldest memento and the distribution of top-level domains and languages in the web archive. These characteristics will be collected via quantitative analysis for the TimeMap responses to different sample sets of URIs. These characteristics will be used to describe each web archive and help the Memento Aggregator to rank and select the web archives that can best answer the URI request.

In this research, we perform a quantitative study to create profiles for 15 web archives around the world. To build these profiles, we use a dataset constructed from URIs for the live web, fulltext search of the archives themselves, and access logs of the archives. We evaluate the quality of the generated profiles in optimizing the query routing across various archives in the Memento Aggregator.

4.2 DATASETS

In this research, we use various datasets in the different experiments. In this section, we discuss the two main datasets.

4.2.1 IIPC 2010 WINTER OLYMPICS COLLECTION

We use the collaborative IIPC 2010 Winter Olympics collection[1], a collection of websites about the 2010 Winter Olympics, for experimental use. Twelve institutions contributed seeds for the project. Internet Archive performed the crawls and made all the harvested materials accessible including extracted metadata and crawl reports. The collection was crawled between 11/2009 and 03/2010 during the 2010 Winter Olympics. The corpus size is 700+GB, and the crawler started with a seed list of 302 websites. Table 3 lists more details about the collection.

[1] `http://olympics.us.archive.org/`

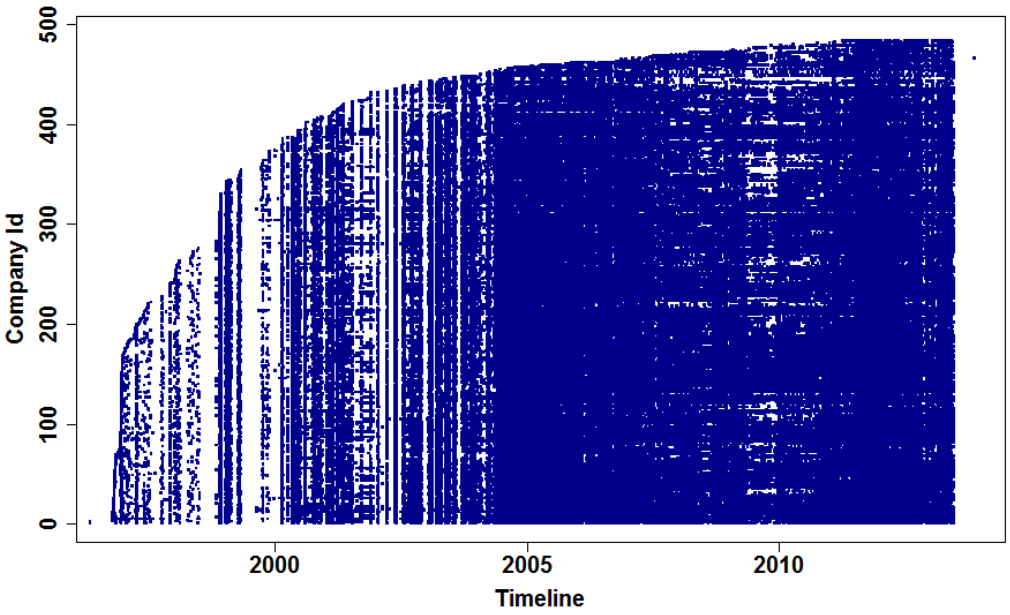


Fig. 28 TimeMap distribution through the time for F500 collection. The mean number of mementos per TimeMap is 1023 and the median is 685.

4.2.2 FORTUNE 500 COLLECTION

We build a collection of TimeMaps using the homepage URIs for the companies listed in the 2013 Fortune 500 list[2]. For each URI, we retrieved the TimeMap from the Internet Archive Wayback Machine. Of those 500 URIs, we have 488 TimeMaps since 12 are not archived due to robots.txt exclusion. The total number of mementos was 499,540. Figure 28 shows the distribution of the mementos through the time. For each memento, we downloaded the HTML text from IA and we used PhantomJS[3] to capture a screenshot for each memento.

4.2.3 OPEN DIRECTORY PROJECT (DMOZ)

The Open Directory Project (DMOZ)[4] is an open source web directory that is built by user submissions of URIs. DMOZ was the source of URIs in various research [90, 135, 142, 211, 221, 232]. Although it is an imperfect source for many reasons (e.g., its contents appear to be driven by commercial motives and are likely biased in favor of commercial sites), DMOZ was included because it is one of the oldest sources available. In particular, DMOZ archives dating back to 2000 are readily available, which makes DMOZ a reliable source for old URIs that may no longer exist. We depend on sampling from DMOZ in various experiments through our research. The sampling technique will differ based on the experiment requirements.

[2] http://money.cnn.com/magazines/fortune/fortune500/2013/full_list/
[3] http://phantomjs.org/
[4] http://www.dmoz.org

CHAPTER 5

CONTENT SERVICE

The content service is the basic service in the web archive services pyramid. The content service layer combines other core web archive applications such as replaying the memento and fulltext search services [261]. The content service is responsible for retrieving the raw content of the page as it appeared in the past. The content may be the whole representation (e.g., HTML, video, audio) or it may be a converted copy of the representation (e.g., textual content extracted from the HTML resource).

5.1 INTRODUCTION

The Wayback Machine is a popular tool to replay archived web pages. In the open source version architecture, it looks up the requested URI in the CDX file to determine the appropriate ARC/WARC files. Then it accesses the designated ARC/WARC file with the specific offset to read the web page content. In addition to these tasks, the Wayback Machine performs URI rewriting for the links and embedded resources to point into the archived embedded resource. URI rewriting means modifying the original version as it was captured to provide a consistent replay experience. The Wayback Machine modifies the URIs within the HTML page, both anchors (i.e., links) and embedded content (e.g., images, applets) to make these resources point to the Wayback application. It is implemented by client JavaScript or server-side code at the dissemination time without modifying the preserved materials on WARC files. Note that the rewriting mechanism is not effective with all types of embedded URIs. For instance, the Wayback Machine can not rewrite the URIs generated by dynamic JavaScript or embedded in Flash or PDF documents [22].

For example, if we investigate a memento for `http://www.google.com` on Nov. 11, 1998, the Wayback Machine provides two ways to replay the memento. First, the original resource as it was captured (`http://web.archive.org/web/19981111184551id_/http://google.com/`) as shown in Figures 29(a) and 29(b). In this case, the links point to the live web. Second, the rewritten memento (`http://web.archive.org/web/19981111184551/http://google.com/`) as shown in Figures 29(c) and 29(d). In this case, Wayback Machine inserts additional JavaScript that adds the TimeMap banner in the top and rewrite the links to point to the archived version from the Wayback Machine.

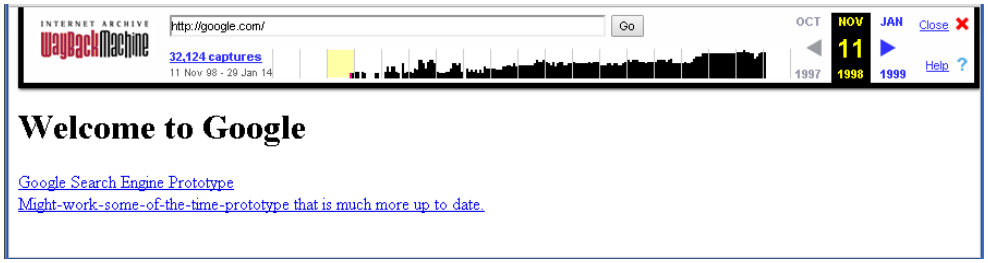
So the web archive should preserve the archived collections in its raw format in addition to building wrapper and filters to convert the web pages to different formats to enhance the browsing experience. We study the content layer as an access service with three main response types: *Raw*, *Modified*, and *Extracted*.

- Raw response: the replay of the bit-stream for the representation. It is available for all MIME types.
- Modified response: returns the raw response that could be rendered by web browser. It includes server-side rewriting of HTML, CSS, JavaScript, and SWF content.

(a) Screenshot for Google.com from 1998 without rewriting.

```
<h1>Welcome to Google</h1>
<a href="http://google.stanford.edu">Google Search Engine Prototype</a><br>
<a href="http://alpha.google.com/">
Might-work-some-of-the-time-prototype that is much more up to date.</a>
```

(b) Source code for Google.com from 1998 without rewriting



(c) Screenshot for Google.com from 1998 with Wayback Machine rewriting.

```
</div>
<script type="text/javascript">
 var wmDisclaimBanner = document.getElementById("wm-ipp");
 if(wmDisclaimBanner != null) {
   disclaimElement(wmDisclaimBanner);
 }
</script>
<!-- END WAYBACK TOOLBAR INSERT -->
<h1>Welcome to Google</h1>
<a href="/web/19981111184551/http://google.stanford.edu/">Google Search Engine Prototype</a><br>
<a href="/web/19981111184551/http://alpha.google.com/">
Might-work-some-of-the-time-prototype that is much more up to date.</a>




<!--
     FILE ARCHIVED ON 18:45:51 Nov 11, 1998 AND RETRIEVED FROM THE
     INTERNET ARCHIVE ON 15:33:24 Jan 29, 2014.
     JAVASCRIPT APPENDED BY WAYBACK MACHINE, COPYRIGHT INTERNET ARCHIVE.

     ALL OTHER CONTENT MAY ALSO BE PROTECTED BY COPYRIGHT (17 U.S.C.
     SECTION 108(a)(3)).
-->
```

(d) Source code for Google.com from 1998 with Wayback Machine rewriting

Fig. 29 Wayback Machine's response for google.com on Nov. 11, 1998.

- Extracted response: returns the textual content of the page excluding all the HTML tags and markup. Text service applies different filters that converts the text to different format. For example, main text filter, Term Frequency filter, and unique words filter.

The extracted responses are the main focus of this research. We develop ArcContent, the web archive content access service. The rest of this chapter is organized as follows. Section 5.2 shows the formal definitions for the content services filters. Section 5.3 shows the formal definitions for the content service operations. Section 5.4 discusses the ArcContent implementation details. Section 5.5 discusses the related work and background for the content service.

5.2 FORMALIZATION

In this section, we formalize the functionality for the content service. The content service starts with the content of the web page as it appeared in the past and applies a set of filter functions.

1. $RAW(M_i)$: the raw representation of the web resource representation as it was crawled at t_i. It may return text, image, or binary content based on M_i MIME type.

2. $HTMLContent(M_i)$: a sequence (ordered list with duplicates) of the textual representation content for the M_i including HTML tags, text, spaces, and special characters. This is for the M_i with an HTML MIME type[1]. The response can be modified to enhance the user browsing experience.

$$HTMLContent(M_i) = \begin{cases} (w_0, w_1, ..., w_n) & \text{if } M_i \text{ has an HTML MIME type} \\ () & \text{if } M_i \text{ does not have an HTML MIME type} \end{cases} \quad (1)$$

   $w_i \in L|L$ is a set of all the words, tags, punctuation, or spaces.

3. $TextContent(M_i)$: a sequence of textual content of the M_i excluding tags and punctuation.

$$TextContent(M_i) = \begin{cases} (w_0, w_1, ..., w_n) & \text{if } M_i \text{ has text/* mimetype} \\ () & \text{if } M_i \text{ does not have text/* mimetype} \end{cases} \quad (2)$$

   $w_i \in Content(M_i)|w_i$ is a word.

4. $TFContent(M_i)$: the term frequency for the words in the document excluding the tags and punctuation.

$$TFContent(M_i) =$$

$$\{< w, n >, w \in TextContent, n \in N, n \text{ is the number of occurrence of } w \text{ in } M_i\} \quad (3)$$

[1]There are multiple MIME type that signify HTML such as text/html and application/xhtml+xml.

5.3 OPERATIONS

1. $TFContent(R)$: the set of words and their frequency that appeared for this R in all the mementos.

   $$TFContent(R) = TFContent(TM) =$$

   $$TFContent(M_0) \bigcup TFContent(M_1) \bigcup \cdots \bigcup TFContent(M_n) \text{ where } M_i \in TM \quad (4)$$

2. $FeaturedContent(R)$: the set of the common words that appeared at least one time in each memento for this $URI - R$ TimeMap.

   $$FeaturedContent(R) = FeaturedContent(TM) =$$

   $$TFContent(M_0) \bigcap TFContent(M_1) \bigcap \cdots \bigcap TFContent(M_n) \text{ where } M_i \in TM \quad (5)$$

5.4 IMPLEMENTATION

For the content service, we build ArcContent, a complete system to extract, preserve, and access the textual content from the archived web pages. Figure 30 shows the main stages for the ArcContent system.

ArcContent differentiates between the *Modified* Filters and *Extracted* Filters. The extracted content will be moved to a new repository that will serve the future requests but modified content may be available directly from the WARC files or the new repository. ArcContent has three main phases: extraction, preservation, and retrieval.

5.4.1 EXTRACTION STAGE

In this stage, ArcContent extracts the textual content from the HTML mementos and sends them to the preservation stage. We use HTML parser[2] to extract the textual content from HTML, the system prototype is modular and can include other parsers.

ArcContent can extract the text from different web archive data sources, namely WARC files and a Wayback Machine. ArcContent is modular and can include new sources, such as text files or an API. Our prototype shows that extraction from WARC files directly saves around 40% of the processing time as opposed to using the Wayback Machine interface. We use Apache Hadoop to extract the textual content in a distributed environment.

5.4.2 PRESERVATION STAGE

ArcContent preserves the extracted content into an interim repository that will be the source of the access interface. In our reference implementation, we use Cassandra database [143], which is a

[2] `http://htmlparser.sourceforge.net/`

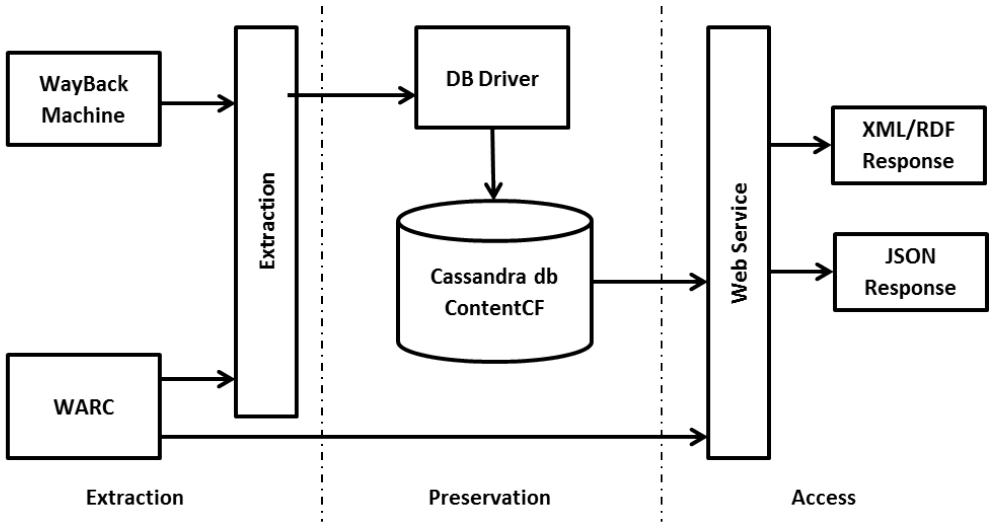


Fig. 30 ArcContent Architecture Diagram.

highly scalable, distributed, and structured key-value store, to save the extracted content. We use Super Column Family schema with checksum-SHA1 as key.

The *Raw* and *Modified* content have two preservation options: keeping the data in its original format (e.g., WARC files) and importing the data into the Cassandra db. The first option avoids the duplication of data between different places and reduces the space requirements. The second option is more suitable for page scraping cases where access to the raw content is through web interface.

5.4.3 ACCESS

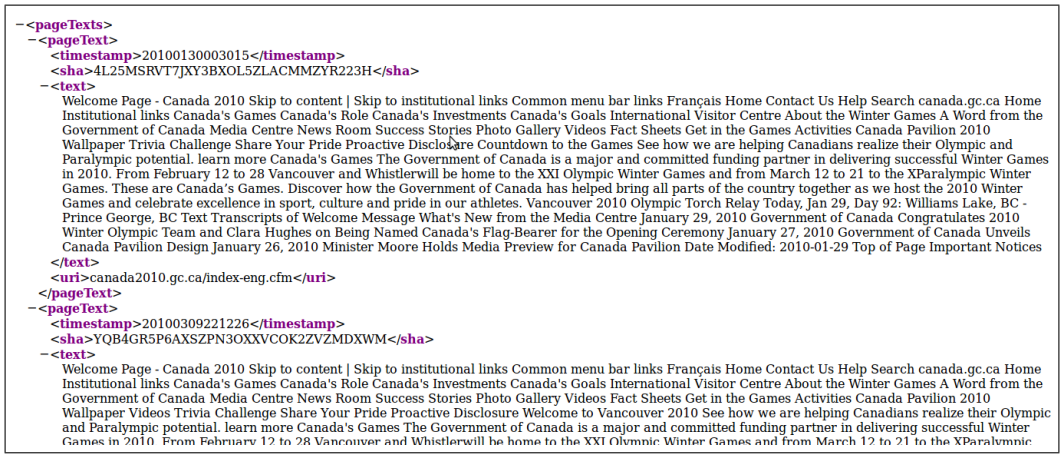
The access phase exposes different filters (representation) of the same content to give the users and third-party developers the ability to build various types of applications. ArcContent's access stage depends on a Java web service that receives two arguments for each request, the URI and the filter type. There is an optional datetime argument to determine a specific memento. ArcContent's access point returns the textual content in different formats such as XML and JSON.

We implement the content service for IIPC 2010 Winter Olympics collection (Section 4.2.1). Figure 31 shows screenshots from our ArcContnet service prototype on the IIPC 2010 Winter Olympics collection. Figure 31(a) is the fulltext filter in XML format, and Figure 31(b) is the fulltext filter in JSON format. Figure 32(a) shows the term frequency filter to count the appearance of each word in the document with a tag cloud application in Figure 32(b).

5.5 RELATED WORK AND BACKGROUND

Parsing the web page is an essential step for the search engine crawlers. Parsing includes extracting the web page content to feed the next step in search engine processes. Parsing may include URI canonicalization, removing stop words, and word stemming [223].

Web pages have various elements such as links, templates, JavaScript, and styles in addition to



(a) Text Service filter response in XML format.

(b) Text Service filter response in JSON format.

Fig. 31 ArcContent access filters response in different formats.

the main text. Various researchers have studied content extraction, which is extracting the main text from web page and excluding the template [136]. The next step is information extraction [103], which is extracting structured data from free or semi-structured documents. There are various techniques for content extraction. Kohlschütter et al. [177] used two shallow text features (i.e., number of words and link density) for classifying the individual text elements in a web page. Zheng et al. [295] used similarity between pages to detect the templates because the normal method to depend on URI only is not enough for dynamic web pages. Lin and Ho [186] developed InfoDiscoverer which partitioned a page into several content blocks, and then computed the occurrence and entropy value of each feature in the set of pages. Based on the entropy value, they considered the block as content block or not. Debnath et al. [106] proposed four algorithms to identify a content block from noncontent block. For example, *ContentExtractor* and *FeatureExtractor* identified noncontent blocks based on the appearance of the same block in multiple Web pages and looking for blocks with

(a) Term Frequency filter.

(b) TagClouds application.

Fig. 32 ArcContent access filter and application.

desired features such as text, images, applets, and JavaScript. Gupta et al. [136] used DOM tree representations instead of the HTML markup. Yi et al. [292] proposed "Style Tree" by converting the web page in DOM tree and defined the duplicate subtree as a template. Bar-Yossef et al. [58] used data mining techniques to extract template and text from web pages. Kovacevic et al. [179] and Baumgartner et al. [59] used visual appearance characteristics to define the content block. Laender et al. [181] performed a survey between web data extraction tools. They proposed a taxonomy for characterizing the web data extraction tools such as HTML-aware tools, NLP-based tools, and modeling-based tools.

Information extraction is the process of extracting structured data from free or semistrcutured documents [103, 200]. Chang et al. [91] surveyed the general techniques for transforming the web information into structure information such as relational database. They defined three dimensions. The first dimension is the cause for the failures of information extraction systems for some particular websites. The second dimension covers the techniques used, such as regular expression, deterministic

finite-state, or Markov models. The third dimension measures the degree of automation for IE systems such as categorization based on the programmer involvement. Liu et al. [187] proposed "Mining Data Records" algorithm to extract data record from the web pages in three steps: building HTML tag tree, then mining data regions that contain similar data, finally identifying data records for each region.

Gomes et al. [125] reported that 67% of the web archives supported full-text search. Costa et al. [100] surveyed the different architectures to build full-text search for the web archive. They discussed the concept of "Temporal Inverted Files" where the index is partitioned by time then by document or term; or partitioning by term or document first then by time. Berberich and Anand et al. [64, 44, 43] introduced time-travel text search for the temporal versioned documents such as Wikis and web archives. They presented time-travel query, e.g., `‘‘fifa world cup’’@[06/2006-07/2006]`. They discussed various inverted file index scheme to support the time dimension with focus on indexing, maintenance, and compressing techniques. Kanhabua et al. [163, 164] discussed two types of temporal query: query with explicit temporal criteria (e.g., "US Presidential Election 2008", and 2) query with implicit temporal criteria (e.g., "Germany FIFA Worldcup"). They focused on the implicit query and how determine the query time and how to re-rank search results [164]. Then, they extended the concept of time to publication time and content time [163]. He et al. [137, 138] proposed different techniques to compress and query temporal versioned documents. They partitioned the index into different time ranges, and access only the requested part. This technique showed better performance than they keyword-only queries.

5.6 SUMMARY

In this section, we discuss ArcContent, a complete system that can extract, preserve, and expose web archive content. ArcContent focus on the textual content extracted from web archive collections. ArcContent exposes the textual content using API interface available in different formats.

CHAPTER 6

METADATA SERVICE

Occasionally, the user may need information about the archived web page that is more than the URI and capture datetime and simpler than the actual content of the archived web page. In this chapter, we extend the concept of web page metadata to include information about the archived web page. Section 6.1 introduces the concept of metadata for the web pages and mementos. Section 6.2 discusses ArcLink, which extracts, preserves, and exposes the temporal web graph. Section 6.3 discusses ArcThumb, which optimizes and summarizes the number of thumbnails that required to represent TimeMap.

6.1 INTRODUCTION

Metadata is commonly defined as data about data [183]. Metadata could be categorized based on the type of data into various categories. *Descriptive* metadata describes the resource content. *Technical* metadata describes the object creation, format, and storage. *Structural* metadata describes the encoding and organization of an object within the digital library. *Preservation* metadata describes how the object could be stored. *Rights* metadata describes the access rights and license to access and use the object [14]. Metadata could be categorized based on the extraction method into two categories. *Extracted* and *Harvested* metadata are extracted from the resource content and *Derived* metadata is developed after a set of extraction and analysis steps. [131].

Metadata fields may include title, HTTP response code, response header, description, and keywords. Table 4 lists the available metadata fields that can be extracted from the archived web page with examples from Figure 24.

As introduced in Chapter 3, the Internet Archive released a new file format standard entitled Web Archive Metadata (WAT) file [123] to store the metadata. WAT utilities are used to extract metadata from WARC files. The structure of the WAT files is optimized for data analysis. WAT files can be used to create reports about the web archived materials based on large data sets. An example of WAT file record for `vancouver2010.com` is available in Listing 5. Even though a WAT file contains a set of metadata fields, it is not suitable for the metadata retrieval services for the following reasons:

1. The final specification of the WAT file is defined as a metadata for the web archived data (i.e., WARC files). So its focus is the WARC file more than the individual mementos themselves. For example, WARC files have three records for each crawled snapshot (request, metadata, and response). The WAT file also has three records which corresponds with WARC records. So it includes extra information that increases the space requirements.

2. WAT files are used for data exchange about the crawls between the web archive institutions and the other partners. So the included information in the WAT file is subject to the same

Table 4 Metadata fields with examples fromFigure 24.

Field	Type	Description	Example
Title	Extracted	The title of the page.	Egypt rejoices at Mubarak departure
HTTP Response Code	Technical	The HTTP response code when this URI was captured (e.g., 200, 301)	200 OK
Outgoing Links	Derived	A list of all the outlinks that the page pointed to	
Content-type	Technical	It is the entity mimetype	text/html
Content-length	Technical	The size of the entity-body	Content-Length: 90883
Thumbnail	Derived	Thumbnail of the rendered web page	
Description	Extracted	Description about the content of the entity-body	The BBC World Affairs Editor John Simpson reflects on how Egypt brought about the overthrow of President Hosni Mubarak

intellectual property limitations as the pages themselves. For example, the WAT file must not contain any of the page content such as the page thumbnail. Copyrighted content should be accessed through the general interface (i.e., Wayback Machine) because it is configured to include all the filtration techniques (e.g., robots.txt exclusion).

3. WAT file size is 30% to 50% of the WARC file, which requires significant storage.

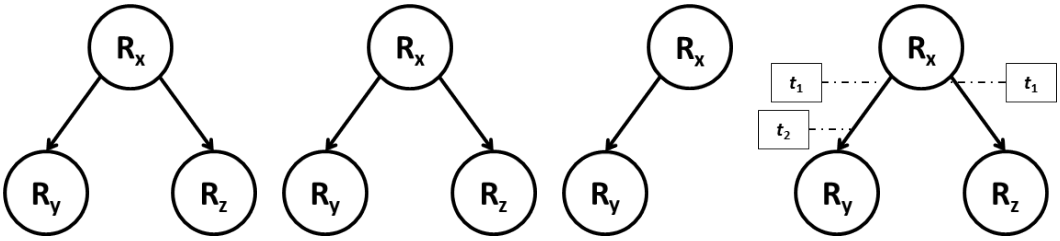
WAT files are required for the mass computation of the web archive metadata but are not suitable for retrieving fine-grained information on the memento level. The metadata service can enhance the web archive interface by displaying extra information to the memento in addition to the capture datetime.

The main challenge in extracting the metadata from the archived web page is that there are separate technique to extract each item. For example, the title can be extracted by parsing the title tag or checking the title field of the HTML META element, whereas taking a thumbnail of the page requires rendering the whole page with all of the embedded resources.

6.2 ARCLINK

The web graph is a directed graph where each vertex represents a URI and the edge represents a hyperlink between them. The web graph is the foundation of various applications: ranking such as PageRank [78] and Kleinberg HITS [175], web spam detection [115, 60], finding related pages [105], and building web graphs based on top-level domains and not individual pages [69].

ArcLink is a proof-of-concept system that optimizes the construction, storage, and access to the temporal web graph. We divide the web graph construction into four stages (filtering, extraction, storage, and access) and implement the optimization in each stage. ArcLink extends the current



(a) Regular WG. (b) WG at t1. (c) WG at t2. (d) Temporal WG.

Fig. 33 Regular and Temporal Web Graph.

web archive interfaces to return content and structural metadata for each URI. We show how this API can be applied to such applications as retrieving inlinks, outlinks, anchor text, and PageRank.

In this service, we propose new optimization techniques for the creation, preservation, and retrieval for the web graph considering the time dimension. The contribution started with decreasing the size of the data by applying filtering techniques based on $URI - M$ characteristics, applying a hashing technique for the URI to enable a native distributed processing with linear and equal insertion and update overhead, developing efficient schema to represent the time dimension, and implementing API interface for accessing the web graph.

Web archives preserve web pages before they change or disappear forever. Each URI may have one or more mementos with different timestamps from which we construct a Temporal Web Graph. Figure 33(a) illustrates the regular Web graph where each URI is connected with another URI by only one link because there is only one snapshot for the source URI. In the temporal web graph (Figure 33(d)), some of the outlinks changed through the time. At t_1, R_x has two outlinks to R_y and R_z (Figure 33(b)), then at t_2, R_x has only one outlink to R_y (Figure 33(c)). So the graph structure is changing through the time.

Web archives do not cover everything in the past. Archival coverage differs based on many factors, including the popularity of the URI [35]. The missing pages will affect the completeness of the Temporal Web Graph because it will decrease the number of available inlinks in the web archive. For example, if $URI - R_x$ has n web pages that pointed to it (n inlinks), the web archive may miss part of n inlinks. The number of incoming links for $URI - R_x$ is determined by the available mementos that point to this URI which may be different from the live web inlinks. So even though $URI - R_x$ may be a popular URI and is captured successfully, the web archive may not archive all the related $URI - R$ that pointed to it and may underestimate the importance of $URI - R_x$.

Motivation

The IIPC Access working group's use cases report [149] highlighted the importance of the linking information (the outgoing and the incoming links) of the archived web for analysis purposes. The report encouraged web archives to develop an API interface to query the link structure for specific URI while recognizing that providing such an API is time-consuming.

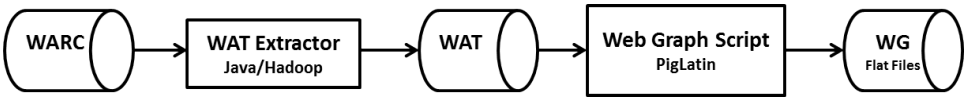


Fig. 34 Internet Archive Web Graph extraction architecture.

The time dimension in the web graph extends the traditional web graph applications. For example, the temporal web graph could be used in ranking the fulltext search for the web archives related to the time. Also, the results should differ based on the time filter[1], for example, the query string "Email" on 2004 might bring "Yahoo Mail" as the first result, today, the same term may return "Gmail". The temporal web graph could be used by the web crawler to discover new URIs, and the increasing number of inlinks could prioritize the crawling schedule. Researchers can use the temporal web graph to study the evolution of the web.

Current Approaches

Due to the lack of APIs, public users and researchers can only scrape the archived pages and extract the links and other metadata. Brügger [83] developed IssueCrawler[2] to crawl the Internet Archive and national Danish web archive Netarkivet[3] to analyze the hyperlink network in web archives; Brügger showed the inadequacy of the crawler to work on the web archive. Weber [286] built a custom web crawler, called HistoryCrawl, to crawl the Internet Archive interface to extract the hyperlinks through the time. Klein and Nelson [173] had to download and page scrape every memento in the TimeMap and extract the changing titles in HTML pages. Also, Padia et al. [222] downloaded pages from Archive-It collections to extract terms for new visualization techniques. CommonCrawl[4] provide an access to a set of metadata (including outlinks) using Amazon EC2 in flat file format. In this research, we show the poor performance of this technique on both the web archive resources and the extraction performance.

The Internet Archive builds web graphs for some collections for internal use where regular users do not have access to it. IA has a two-phase extractor to construct the web graph. Figure 34 illustrates the IA web graph extraction architecture. First, IA builds WAT files for the available WARC files. The second phase uses the WAT files to construct the web graph. IA uses PigLatin scripts to extract the complete web graph from the outlink field in the WAT file. The limitations of this technique are the following:

1. Generally, it is a batch process and can not be implemented on a fine-grain level (e.g., site or domain).

2. IA uses the extracted web graph internally. There is no public access point to read the graph.

3. The extracted web graph can not be aggregated with other web graphs on the same web archive or other web archives.

[1] `http://www.alexa.com/help/traffic-learn-more`
[2] `https://www.issuecrawler.net/`
[3] `http://netarkivet.dk/`
[4] `http://commoncrawl.org`

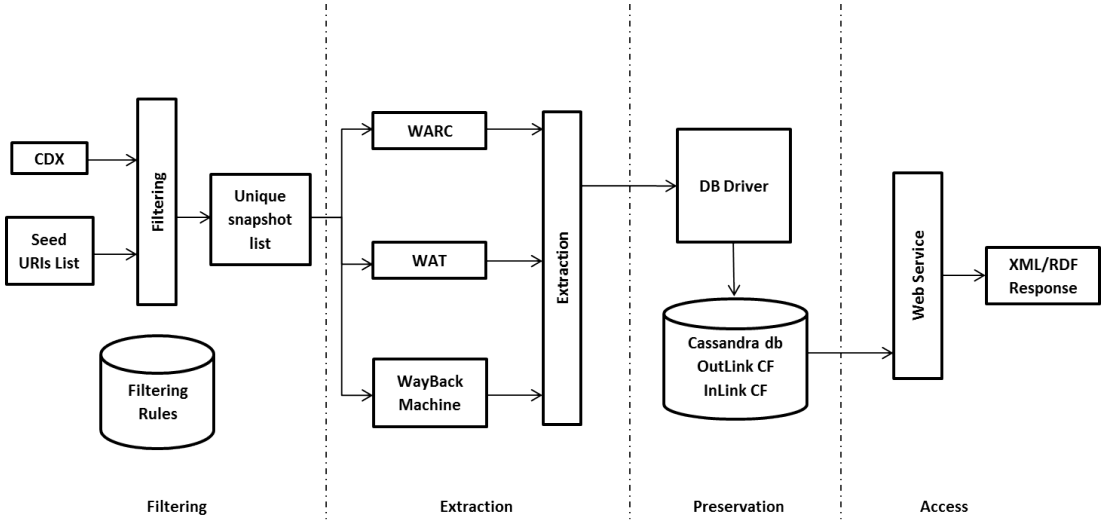


Fig. 35 ArcLink Architecture

4. This technique does not support incremental update.

6.2.1 ARCLINK STAGES

ArcLink is a distributed processing system to pre-process the link structure information from the archived web collections and build a temporal web graph. ArcLink has four main stages: filtering, extraction, storage, and access.

Figure 35 illustrates the different stages and the relation between them. *Filtering* is using the crawler log to omit the mementos that will not contribute to the final temporal web graph. The goal is to reduce the size of the input corpus to optimize the following stages. *Extraction* is responsible for extracting the outgoing links from the mementos. Extraction is available from various input sources. *Storage* is preserving the web graph into a database for further usage. *Access* provides APIs that have the outgoing and incoming links for the requested URI. In this research, we study the characteristics of each stage and propose suitable optimization approaches.

6.2.2 FILTERING OPTIMIZATION

The goal of the filtering optimization is to reduce the size of the input corpus to focus on the snapshots that carry link structure information. Lee et al. [184] defined a page that has no outlinks as a dangling page and proposed how it should be incorporated in the PageRank computation. ArcLink avoids dangling pages in the filtering phase, however they may be included later if ArcLink detects a link to this dangling node. The input for the ArcLink's filtering stage is a list of the URIs within a collection. For Heritrix-based web crawlers, we used the CDX file.

The ArcLink filtering stage uses this information to create a unique list of mementos that will contribute to the web graph. Reducing the input size will reduce the extraction time and the storage space.

Table 5 Reduction Efficiency Experiment Results.

Rule type	Rule parameter	#Remaining records (out of 23.8M)	Time
INCLUDE	HTTP Status 200	17.908M (75%)	936 sec
EXCLUDE	Images, JS, and CSS	15.922M (67%)	807 sec
INCLUDE	text/* only	12.656M (53%)	744 sec
EXCLUDE	Resources with image extension	16.338M (69%)	845 sec
EXCLUDE	Duplicate checksum	9.191M (39%)	886 sec
	All the rules	6.789M (29%)	2098 sec

Filtering rules

Building the web graph started with extracting the outlinks of the web page and creating the inlinks model from that. Based on this procedure, the filtering stage will exclude any memento that does not have outlinks (e.g., images). The following set of filtering rules could be used:

- **HTTP Status:** Include memento with a successful HTTP status (i.e., 200).
- **Content Mimetype:** Exclude mementos with MIME type that do not carry textual content (e.g., images, javascript, stylesheets).
- **Resource extension:** Exclude mementos with file extension that suggest non-textual content even if the MIME type is text/html.
- **Content checksum:** CDX files include the page content checksum calculated with SHA-1 algorithm for each memento. This value could be used to exclude duplicate mementos.

Implementation

Apache Pig is an Apache open source tool that is customized for parallel processing of large-scale data. Filtering rules are written with a PigLatin script that is capable of loading the CDX file fields, then applying INCLUDE/EXCLUDE filters. PigLatin is customized to work with big data, using Hadoop clusters for parallel processing. The output of the filtering steps is a list of the unique mementos after applying the filtering rules.

Experiment

To quantify the efficiency of the filtering optimization techniques, we run various filtering rules on the 2010 Winter Olympics collection's CDX files. The input file has 23.7M URI-Ms. The success criteria is how to include only the mementos that will contribute to the temporal web graph and how to exclude the dangling and duplicate nodes that will not add any value to the web graph. The experiment calculates the reduction in the number of items by applying each rule and what is the total gain of applying all the filters. The efficiency is the percentage of mementos in the output snapshot list as compared to the original number of snapshots.

Results and Analysis

Table 5 shows the experiment results. First, only 75% of the mementos have HTTP status 200; this means 25% of the collection does not have available representations or carry HTTP redirection status. Second, 33% of the collection was embedded resources (e.g., images, stylesheets). Third, the percentage of duplication in the collection is approximately 61%. Finally, applying all rules shows that the number of output records are 29% of the original size. This reduction in the number of records reduces the the required computation time in the upcoming stages.

The filtering rules approach is flexible enough to manage several kinds of computation. For example, it could be used to extract images only or videos only by adapting rules to filter by MIME type.

Generally, the crawling log is not available for public users. The screen scraping technique (as discussed in [83, 173, 222, 286]) could not benefit from this information. Absence of this information adds extra load to the web archive and the client during screen scraping. We recommend that web archives to publish this information to public users to help the third-party developer/researcher. IIPC funded the IIPC Memento Aggregator Experiment [240] to aggregate all the CDX files of the distributed archives of IIPC members to provide Memento based access to the holdings of the open, restricted, and closed archives.

6.2.3 EXTRACTION OPTIMIZATION

The first step of constructing the web graph is extracting the outgoing links from the web pages. Optimizing the extraction stage focuses on two things: the creation of the URI-ID, and extraction mechanism (data source and tools).

URI-ID Generation

The creation of a unique ID for each URI is a common technique in web graph creation. ID creation approaches may depend on ordering, by lexicographical ordering [233, 213, 50, 74, 134], inlinks degree [34] then applying Huffman codes, or storing in an array then using the array index as ID [70]. These approaches made a tightly coupled relation between the URIs themselves, so the parallel and distributed processing were impossible. Also, it affected the update and re-indexing process. To avoid these problems, ArcLink generates a unique ID for each URI in the following manner:

1. Canonicalize
   the URI into SURT format[5] converting URI `<scheme://domain.tld/path?query>` to SURT form `<scheme://(tld,domain,)/path?query>`.

$$\left.\begin{array}{l} www.example.org/foo.html \\ example.org/foo.html \\ www1.example.org/foo.html \end{array}\right\} \rightarrow example.org/foo.html$$

$$\left.\begin{array}{l} example.org/foo.html \end{array}\right\} \rightarrow org,example/foo.html$$

[5] `http://crawler.archive.org/articles/user_manual/glossary.html#surt`

2. Encode this canonicalized SURT string using SimHash [92] with 64 bit length.

$$org,example/foo.html \rightarrow FC9B122400794808$$

Using this 1-1 mapping between the URI and the ID enables ArcLink to increment and distributed processing for the same URI on different cycles or different machines because the ID generation depends on the URI only.

Implementation

Apache Hadoop is a framework that allows the distributed processing of large data sets across clusters of computers using a simple programming model. We build MapReduce jobs using Java. The mapper part starts with one memento at a time and extracts the outlinks based on the available source. For extracting the link structure from the WARC and web interface, we use HTML Parser, which is a Java library used to parse HTML. ArcLink focuses on the following fields: the outlink (anchor link or embedded resource link), the type (href or image), and the associated text (anchor text for the href or the alternate text for the image). The reducer is responsible for ID creation for each of the extracted links and creating one record that contains (Document Checksum, Outlink URI, OutLink ID, type, text). ArcLink could extract the link structure from three sources: WARC, WAT, or Web Archive UI.

Partitioning

Hadoop provides a transparent partitioning technique to divide the data on the cluster machines, but the input file for the extraction does not carry the input data, it only has pointers to the actual data (e.g., WARC files). Each line in the input file has the URI and WARC file name with an offset to the content. Processing each record requires an access to the file on the harddisk. The simultaneous multi-access to the same file affects the performance of the extraction stage [213]. ArcLink uses Unix commands (sort and split) to partition the input files to ensure a single access to the WARC file per task. Each WARC file name will appear in only one split file, the input split file could contain one or more WARC file. Each split file will be assigned to one Map task.

Table 6 Extraction Experiment Results.

Input		**Map**	**Reduce**	**Total time**
2 Tasks (Partition)	*WARC*	13,327	2,770	16,098
	Web Archive UI	21,422	4,194	25,616
2 Tasks (Normal)	*WARC*	15,324	2,940	18,265
	Web Archive UI	17,447	4997	22,444
5 Tasks (Partition)	*WARC*	8,304	1,746	10,051
	Web Archive UI	13,721	2,257	15,978

Extraction from WAT files

The evaluation does not cover the extraction from the WAT files. Theoretically, the extraction from the WAT file is faster than the extraction from the actual content (WARC files) because the WAT

file has been pre-processed earlier. WAT creation is an expensive task because it extracts a lot of information from the actual text. So the quantitative comparison should consider the time to create the WAT file. Also, WAT files are not available for all the collections.

Experiment

In this section, we perform a quantitative study to compare the various extraction techniques. We extract the outlinks from two sources (WARC and web archive UI) for the same collection. We repeat the experiment on different numbers of MapReduce jobs two tasks and five tasks. For the two tasks experiment, we sample 800k records from the filtering phase output, and we feed the extractor on two modes: *Normal* using the default Hadoop partition, and *Partition* in which ArcLink splits the data before the submission. For the five tasks experiment, we sample 500k records. The task is repeated to extract from the WARC file and web archive UI. Accessing the web archive UI was done without any politeness period between the requests.

Table 6 shows the results for *Map phase* (total time for all mappers), *Reduce phase*, and the total time for both phases in seconds. The results show that using WARC as data source is 61% more efficient than performing page scraping from the web archive UI interface. Also, the partitioning technique has better performance with two tasks (85%).

The extraction from the IIPC 2010 Winter Olympics dedicated Wayback Machine server failed in the first round because of the server was not prepared to receive this high load. After updating the memory configuration, we are able to access the server with different tasks. This problem does not happen with WARC extraction because we depend on Hadoop DFS [76] (part of Apache Hadoop project) which is designed to receive high load of requests. It shows the negative impact of the page scraping on the archive resources.

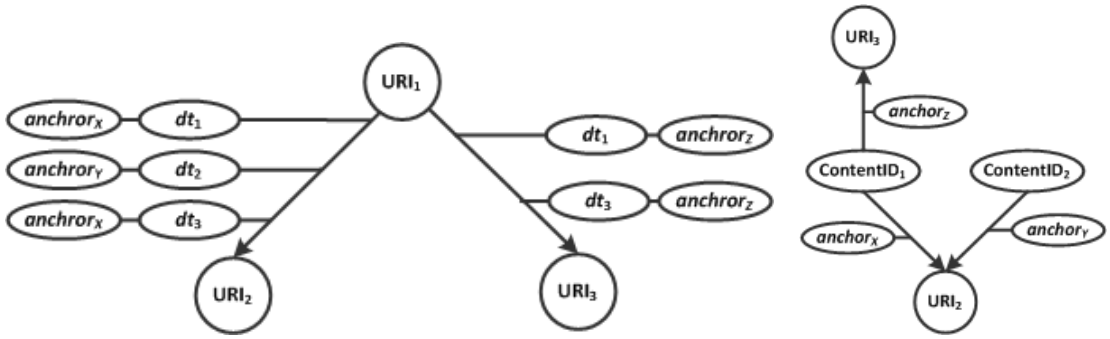
6.2.4 PRESERVATION OPTIMIZATION

In the previous stages, the optimization techniques focused on time and computation power. In the preservation optimization stage, ArcLink will optimize the required space to store the web graph. ArcLink preserves the extracted links structure for future access using a database which will be the repository of the link structure information.

Schema

The outlinks and inlinks with temporal dimension are many-to-many relations with various properties. For example, the $URI - R$ may have different mementos, each memento may have the same or different outlinks with different anchor text.

The research question is how to represent the web graph with the temporal information. Usually, the web graph is represented as a directed graph: the vertex represents the URI and the edges represents hyperlinks between the URIs. With the temporal dimension, $URI - R_x$ could point to $URI - R_y$ at different timestamps with different properties (i.e., anchor text). Adding this information to the graph needs a more sophisticated method to make the graph with the minimum size. We build two schemas to represent the temporal web graph.



(a) Web graph with temporal properties. (b) Content-centric.

Fig. 36 Temporal Web Graph schema.

- *Web Graph with temporal properties* (Figure 36(a)): In this schema, we expand the regular web graph to include attributes for the edges. For each edge, which represents a link from $URI - R_x$ to $URI - R_y$, we add two fields, the datetime for this memento and the anchor text for this reference. We used this schema for access (Section 6.2.5), because it is more readable for users and applications.
- *Content-centric temporal web graph* (Figure 36(b)): In this schema, we replace the $URI - R$ and datetime attribute with the checksum for the content for this memento. The duplicate mementos will have the same content (with the same checksum) which evolve to the same vertex. This schema is used for preservation because focusing on the content will remove the duplicate information. The rest of this section will explain the schema implementation in Cassandra db.

Implementation

The ArcLink reference implementation uses the Apache Cassandra database [143]. We used the "*Super Column Family*" structure to build the schema for saving the link structure information. The advantage of this technique is providing the same analogy of the temporal relation between the datetime and the list of mementos with different attributes for each one. The Cassandra db handles the update/insert operations, so even if we insert the same record twice it will detect it was previously inserted and update the content with the new information.

Experiment and Results

In this section, we evaluate the efficiency of the new schema in both space (reduction in storage) and time (insertion/update).

The database driver is the program that is responsible for inserting the extracted links into the database, creating the outlinks and inlinks tables. The Cassandra database driver calculates the required time for the insertion for evaluation purposes. Then, we repeated the process again to

calculate the time for the update. Figure 37 shows that there is a linear relationship between the number of links and required time for insertion. Also, the same linear relationship appeared for the update which means there is no extra overhead for the update process.

The content-centric schema focuses on the content more than the URI. The same checksum may belong to one URI in one timestamp (unique memento), one URI in different timestamps (duplicate mementos), or different URIs in different timestamps (duplicate content). Our experiment found duplicate content is common. For example, the Olympics collection has 23K mementos that have a custom a "Page not found" that returns HTTP status 200 instead of 404 (known as a "soft 404" [56]). In the regular web graph, each one of these snapshots will have one vertex with the same outlinks.

6.2.5 ACCESS

ArcLink provides APIs to enable applications to access the link structure information. ArcLink delivers the link structure information to other systems instead of processing/analyzing the information by itself. It makes ArcLink a source of knowledge which could be expanded by incrementally processing more archived web material or by aggregating the ArcLink interface with other archives. The ArcLink delivery method depends on REST web services. The ArcLink response is annotating the URI-R with the outlinks and inlinks predicates through time. Listing 4 shows a complete request/response session for ArcLink. Line 1 shows the curl request. The service take a `uri` parameter that denotes the requested URI. Lines 3-32 show the response in RDF/XML. Line 5 shows the URI as subject. Lines 6-18 show the *hasOutlinks* to list the outlink set with the observed datetime and anchor text. Lines 19-31 shows *hasInlinks* to list the inlink set.

The ArcLink interface is built on RDF that could be aggregated with other ArcLink instances that carry information about the requested URI-R. The aggregation could be done on the response level to give each ArcLink implementer the freedom to adjust the implementation details based on the requirements and capabilities.

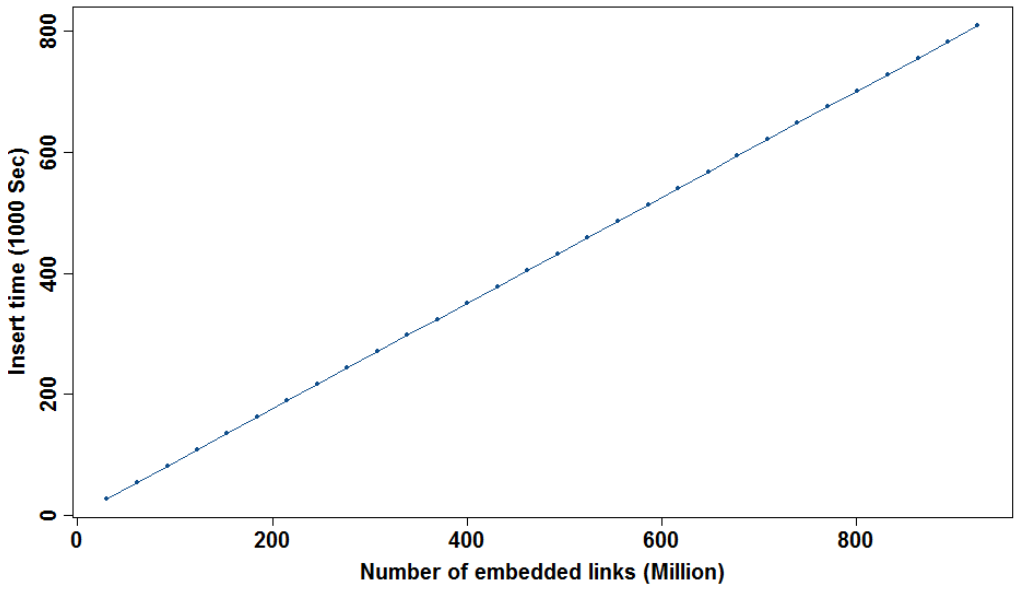
6.2.6 SCALABILITY AND COST MODEL

ArcLink has been designed to deal with large-scale web archives. The evaluation for the different stages shows a linear relation in every stage. The Filtering stage applies different rules to the CDX input file; each record is visited once. The total complexity of this stage is $O(n)$, where n is the number of records. Extraction is also linear; each file is visited only once. Section 6.2.4 discusses the linear complexity for the storage stage. In addition to the construction process, the access stage could be aggregated with other web graphs based on the RDF aggregation schema. Based on our empirical study, we present the following cost model:

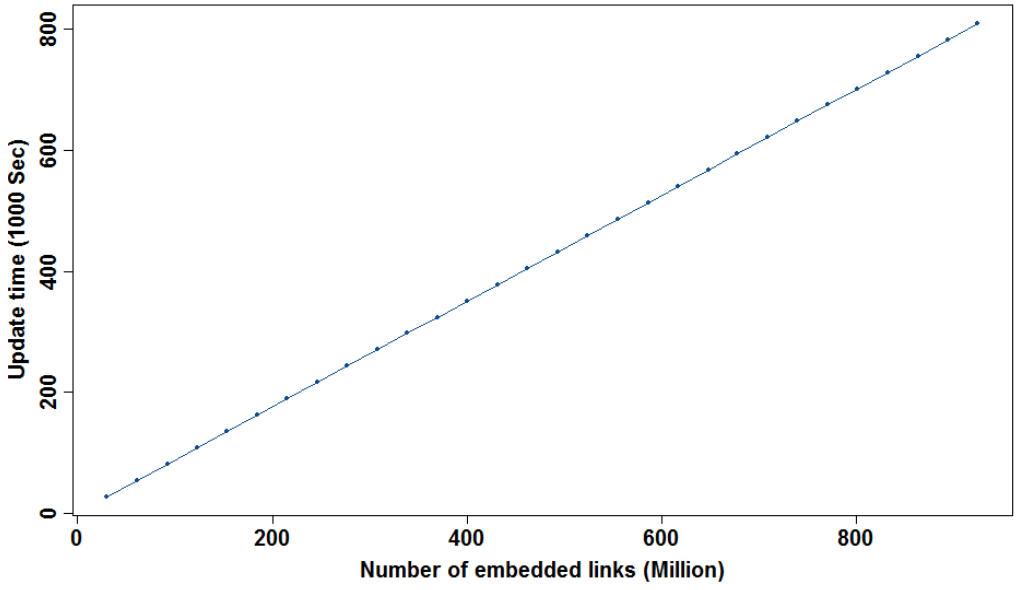
- **Filtering stage**
  Assume you have CDX input file with size n *mementos* using m machines in a Hadoop cluster.

$$FilteringTime = n/10^6 * 88/m \ (sec) \tag{6}$$

$$FilteringReduction = n * 0.30 \ (mementos) \tag{7}$$



(a) Insert Time.



(b) Update Time.

Fig. 37 Insert and Update times.

- **Extraction stage**
  Assume you have a web archive corpus with n *mementos* using m machines in a Hadoop cluster.
  $$ExtractionTime = n/10^6 * 5.5/m\ (hrs) \tag{8}$$
  So you can extract the link structure for 1 billion mementos within 2 days using a cluster of 100 nodes.

- **Storage stage**
  In the previous stages, all the output data is temporary and consumed by the next phase. The storage stage will preserve the link structure for further access. The required space for the Cassandra database is 35GB for outlinks and almost the same for inlinks. Also, it consumes 2GB for the total URI ID information. So the total required space is almost 10% of the original size of the collection. The Extraction stage produces the web graph in a flat file with size 120GB.
  $$StorageSize = Outlinks(n) + Inlinks(n) + Link(n) \tag{9}$$
  $$Outlinks(n) = n * 0.05;$$
  $$Inlinks(n) = n * 0.05;$$
  $$Link(n) = n * 0.002$$
  $$and\ n\ \text{ is the size of the collection.}$$

If we are to scale this up to the announced WayBack Machine with 360B mementos and total size of 5PB [237], the results would be:
$FilteringTime = 360 * 10^9/10^6 * 88/100 = 88\ hrs$
$FilteringReduction = 360 * 10^9 * 0.30 = 108 * 10^9\ mementos$
$ExtractionTime = 108,000 * 10^9/10^6 * 5.5/100 = 247\ days$
$StorageSize = 5\ PB * 0.05 + 5 * 0.05 + 5 * 0.0.2 = 500\ TB$

6.2.7 APPLICATIONS

In this section, we explore various applications that can benefit from the temporal web graph.

Temporal Web Graph Properties

The temporal web graph for 2010 Winter Olympics collection contributes 19.8M nodes (each node represents a $URI - R$) with 792+M edges (each edge represents a hyperlink). We analyzed the $URI - R$ set and found that 18.57M URI-Rs do not have any mementos in the collection. So for this bounded collection, 92% of the outlinks were not crawled which means less ability for the user to browse the collection as it appeared in the past.

Temporal PageRank

PageRank [78] is a ranking technique used to rank a set of web pages based on the link structure. In this section, we explore the ranking information through time. First, we calculate the page rank

for the whole collection without the time dimension. We create an edge between $URI - R_x$ and $URI - R_y$ if $URI - R_x$ pointed to $URI - R_y$ in any point of the time. Then, we repeat the algorithm with the time dimension. We divide the graph per month. For example, for the January 2010 graph, we create an edge between $URI - R_x$ and $URI - R_y$ if $URI - M(URI - R_x)$ pointed to $URI - R_y$ during January 2010.

Table 7 shows the PageRank for the collection per month. Table 8 shows the rank for the whole collection. The rank for each month is affected by the crawling activity during this month. November 2009 focused on the preparation of the Olympics, and the `vancouver2010.com` domain has the highest rank during this month. By March 2010, the news about the Olympics became more important. Newspaper sites like `lefigaro.fr` and `lemonde.fr` receive higher ranks. We could consider the whole collection rank as the intersection of the rank for the various months. We calculate the overlap (the intersection between two sets) and correlation (using Kendall's τ) for the top 50 results in each month [205]. We compare the page rank between each subsequent months, then with the whole collection results. Table 9 shows a weak positive relationship between the ranks in different months, which means we can not depend on specific time rank to estimate another time period.

The temporal PageRank is affected by the available mementos on this date. We were not able to get a page rank for December 2009 because there were so few mementos captured that month.

Time-Indexed Inlinks

The Memento protocol provides a TimeMap for each $URI - R$. For each memento, the user could extract the list of outlinks, but this is an expensive operation as noticed by Weber [286]. Thelwall and Vaughan [271] used AltaVista to retrieve the inlinks information from the current web instead of Internet Archive in order to examine the country balance in Internet Archive. With the current WayBack Machine configuration, it is not possible to extract the incoming links for a specific $URI - R$ through the time, but ArcLink makes such an application possible.

Listing 4 shows a sample from the access response for `vancouver2010.com`. The response has a complete list of the available incoming links to this URI. From this response, we can produce various information. For example, Table 10 shows a sample list of the anchor text for different incoming links to `vancouver2010.com` ordered by time. This is not feasible with the regular WayBack Machine. If the $URI - R$ is not in the archive, this summary of the anchor text could provide a good description of what is missing [174].

6.2.8 RELATED WORK AND BACKGROUND

Link Database [233], a part of the Connectivity Server [70] (used by AltaVista), provides fast access storage to the web graph. Link Database compressed the URI Id to 6 bits per link, so the graph could be loaded into main memory. Scalable Hyperlink Store (SHS; used by Microsoft) [213] is a distributed in-memory "database" to store the web graph. SHS provides fast access by keeping the web graph information in main memory. SHS provides a kind of API to facilitate the interactions with SHS servers. Suel and Yuan [265] created a hostname list and URI list. They used Huffman codes to compress each one, then divided the links into global links between pages on different hosts,

Table 7 2010 Winter Olympics collection's ranking through time.

	Nov-2009	**Jan-2010**
1	monlibe.liberation.fr	monlibe.liberation.fr
2	topsport.com/sportch/liveticker/	laprovence.com/la-provence-le-faq-de-la-moderation
3	lefigaro.fr	get.adobe.com/flashplayer
4	laprovence.com/la-provence-le-faq-de-la-moderation	vancouver2010.teamgb.com /teamgb/team-behind-team-gb/filenotfound.aspx
5	lefigaro.fr/sport	ledauphine.com
6	get.adobe.com/flashplayer	lefigaro.fr/economie
7	lefigaro.fr/meteo	lefigaro.fr/sport
8	lefigaro.fr/le-talk	lefigaro.fr/actualites-a-la-une
9	dosb.de/de/vancouver-2010/vancouver-ticker/detail/printer.html	lemonde.fr/cgv
10	ledauphine.com	ffs.fr/index.php

	Feb-2010	**Mar-2010**
1	monlibe.liberation.fr	monlibe.liberation.fr
2	topsport.com/sportch/liveticker/	laprovence.com/la-provence-le-faq-de-la-moderation
3	lefigaro.fr	get.adobe.com/flashplayer
4	laprovence.com/la-provence-le-faq-de-la-moderation	vancouver2010.teamgb.com /teamgb/team-behind-team-gb/filenotfound.aspx
5	lefigaro.fr/sport	ledauphine.com
6	get.adobe.com/flashplayer	lefigaro.fr/economie
7	lefigaro.fr/meteo	lefigaro.fr/sport
8	lefigaro.fr/le-talk	lefigaro.fr/actualites-a-la-une
9	dosb.de/de/vancouver-2010/vancouver-ticker/detail/printer.html	lemonde.fr/cgv
10	ledauphine.com	ffs.fr/index.php

Table 8 2010 Winter Olympics collection's general ranking.

	Collection (*Nov-09 to Mar-10*)
1	monlibe.liberation.fr
2	vancouver2010.com/code
3	lefigaro.fr
4	laprovence.com/la-provence-le-faq-de-la-moderation
5	lefigaro.fr/sport
6	get.adobe.com/flashplayer
7	lefigaro.fr/meteo
8	lefigaro.fr/le-talk
9	topsport.com/sportch/liveticker/
10	vancouver2010.com/en/langpolicy

Table 9 Overlapping and (Correlation) between the top 50 results.

	Jan	**Feb**	**Mar**	**Collection**
Nov	29 (-0.042)			27 (0.078)
Jan		5 (0.089)		27 (0.169)
Feb			16 (0.053)	22 (0.185)
Mar				14 (0.062)

Table 10 vancouver2010.com Inlinks anchor text.

Date	**Text**
04-Nov-09	vancouver2010.com
11-Nov-09	vancouver2010.com
18-Nov-09	vancouver2010.com
16-Jan-10	Vancouver 2010 Olympic Games
16-Jan-10	Vancouver 2010 Olympic Games
23-Jan-10	vancouver2010.com
23-Jan-10	2010 Vancouver Olympic Games Medals Results Schedule Sports
30-Jan-10	2010 Vancouver Olympic Games Medals Results Schedule Sports
30-Jan-10	vancouver2010.com
30-Jan-10	Vancouver 2010 Olympic Games
13-Feb-10	Vancouver 2010 Olympic Winter Games
15-Feb-10	Vancouver 2010 Olympic Games
18-Feb-10	Official Vancouver Games site
19-Feb-10	vancouver2010.com
20-Feb-10	Official Vancouver Games site
21-Feb-10	VANOC 2010

and local links between pages on the same host. These systems were built to run on a single machine, and mainly to work on the live web; they did not take the temporal dimension of the web archive in consideration.

Avcular and Suel [50] discussed distributed manipulation of archival web graph using Hadoop. Pregel [191] is a distributed system for efficient processing the large scale graph. PeGaSus [162] is a petascale graph mining library to process the large web graph. Donato et al. [108] studied the properties of web graph based on Stanford WebBase collection [95].

Bordino et al. [75] provided statistical analysis to the temporal characteristics of 100M mementos covering 12 months of the .uk domain captured between June 2006 to May 2007. They built a web graph for each month using Web Graph [74].

Anand et al. [45] proposed EverLast, a distributed framework to address the challenge of capturing, preserving, and querying the web archive. EverLast proposed a Time-Travel Index (TTI) to support queries of the web archive. Song and JaJa [256] proposed the EdgeRank technique to partition the web graph in order to merge the web pages in container. The new algorithm decreased

the number of containers that were needed to browse the web archive that enabled faster browsing of archived web contents. Also, Song and JaJa [255] proposed a new schema for efficient preservation and retrieval for the archived web.

6.3 ARCTHUMB

A thumbnail is a small image that represents a web page as it is rendered in a web browser such as Firefox or Chrome. Representing web pages with thumbnails has been used in various research such as the visualization of the web search results [289, 180, 275], the visualization of recommended and similar pages such as `SimilarWeb.com`, and helping in revisitation and remembrance of the web page [161, 268]. Using thumbnails in web archives and temporal data has been studied [222, 33], but the creation cost of the thumbnails in the web archive has not yet been studied.

There are currently a few web archives that create thumbnails for mementos. For example, the UK Web Archive and Archive.is provide a partial list of thumbnails per URI. Archive-It enables the partners to generate thumbnails for quality assurance purposes.

Figure 38 shows two examples of a TimeMap for `VisitWales.com`. The HTML bubble interface in the Internet Archive[6] is in the back with the red border while in the front is the thumbnail view of the same URI in the UK Web Archive[7]. The thumbnail view gives the user an idea about the website layout and its evolution through time. Also, it helps the user in searching and revisitation for specific mementos of this URI. For example, thumbnail view for `VisitWales.com` could show that it did not change from Jul 2012 to Sep 2012; the website layout was the same from Feb 2011 to Jun 2013 while it had a slight change for the rest of the timeline. This visual representation of the website through the time helps the user to select the desired mementos in the TimeMap. The thumbnail with the text "Sorry, no thumbnail yet" could have been generated for several reasons, but a likely explanation is that the HTML was not successfully processed by the thumbnail generator.

Creating thumbnails in the web archive has a number of challenges. First challenge is the scalability in time, the computation of the thumbnail requires rendering of the web page including retrieving all the images, style sheets, and running any javascript. Second challenge is the scalability in space, the thumbnail (as metadata information about the web page) requires storage and our experiment shows that the average size for thumbnail with 64x64 pixels is 3KB and with 600x600 pixels is 133KB while the average size of HTML text is 39KB. Based on the Internet Archive estimated size of 360 billion mementos [237], they would need an additional 502 TB to store a 64x64 thumbnail for each memento. It would reach 21.75 PB to store a 600x600 thumbnail for each memento. Third challenge is the page quality, the construction of the archived web page may not be successful at the time of building the thumbnail due to the missing embedded resources.

Using thumbnails also has some limitations. Some TimeMaps have too many mementos for browsing. For example, the Internet Archive has over 10,500 mementos for `apple.com` over an 18 year span. So even if the Internet Archive has thumbnails for all the mementos (as shown in Figure 39(a)), some form of sampling or partitioning will be necessary because the cognitive load of processing all the mementos will be beyond what the user can handle [198]. Figure 39(b) shows

[6] `http://web.archive.org/web/*/http://visitwales.com/`
[7] `http://www.webarchive.org.uk/ukwa/target/56197146/source/subject`

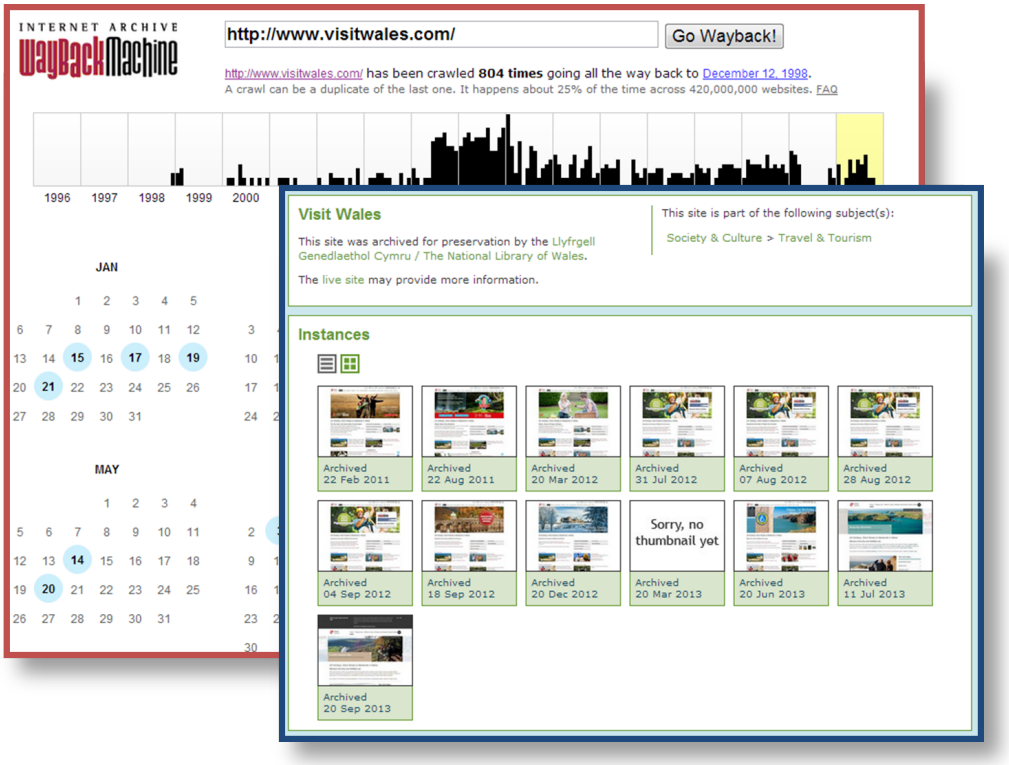


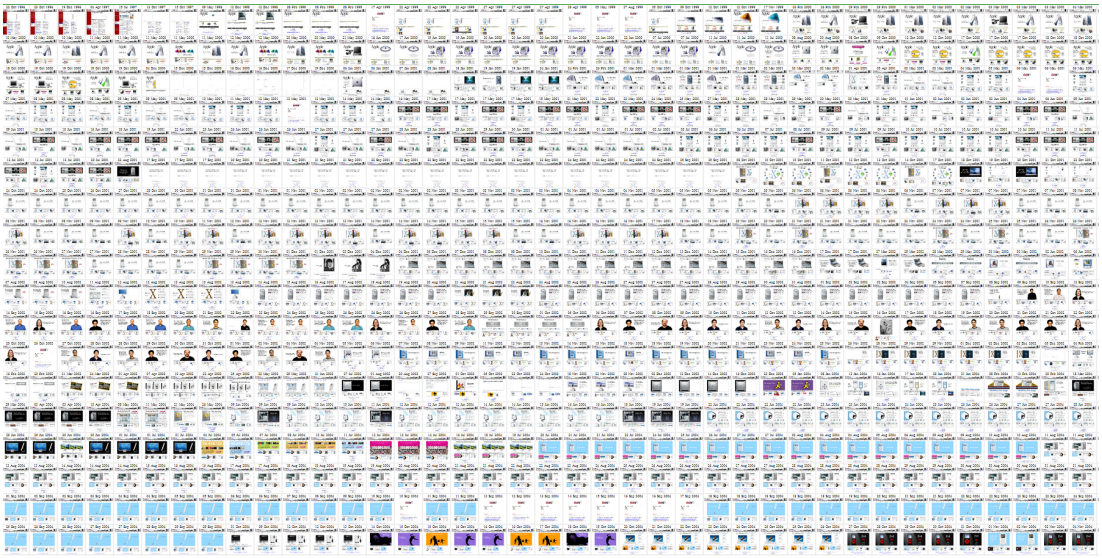
Fig. 38 Different representations for VisitWales.com TimeMap.

a summarized version of the TimeMap, which is easier to visualize and understand. In a personal digital library, Graham et al. [128] suggested 25 images per view panel because displaying all images did not help in conveying an overview of the album.

In this research, we propose ArcThumb for thumbnail creation in the web archive. ArcThumb focuses on selecting the most representative thumbnails to summarize a TimeMap. We use the HTML text of the mementos and the crawler log information to predict the visual change in the URI through the time in order to select the significant set of mementos that can summarize the TimeMap. We explore various features that could be extracted from the HTML pages and select the most relevant to the visual effect, and then we propose different techniques for online and offline thumbnail creation.

6.3.1 EXPLORATION OF FEATURES

In this subsection, we explore various features that can be used to predict the change in the visual representation of the web page. The features could be obtained from the crawler log (e.g., CDX file for Heritrix crawler) or from the HTML text of the memento.



(a) Partial view of all thumbnails.

(b) Summarized thumbnails.

Fig. 39 TimeMap for apple.com represented using thumbnails..

Web page similarity features

There have been several methods proposed for calculating similarity between web pages [140, 79, 92, 192]. In this subsection, we explore various features and techniques to estimate the resulting difference in thumbnail images based on the HTML source of the mementos.

- **SimHash Similarity**

  SimHash [92] is a fingerprint technique to calculate the near-duplicates between web pages. We used 64-bit SimHash fingerprints with $k = 4$. We calculate the SimHash for different parts of the web page. First, we calculate SimHash for the full HTML text. Then, we use boilerpipe library[8] [177] to extract different parts of the web page such as the main content from the web page, all the text (even the text within the template), templates including the text, and the template excluding the text (just HTML structure). For each subsequent memento in the TimeMap, we computed the Hamming Distance between their SimHash fingerprints.

- **Levenshtein Distance between HTML DOM Tree**

  Each web page can be expressed in a DOM tree. We can compare two web pages by calculating the difference between their DOM trees. In this feature, we transform each HTML page into a DOM tree (using jsoup[9]), then we calculate the Levenshtein distance between both trees by calculating the number of operations (insert, delete, and replace) to turn one tree into the other [226].

- **Embedded Resources**

  The change in the embedded resources that construct the web page affects the visual appearance of the rendered web page. We extract the embedded resources for each memento and calculate the difference between each pair of pages. We calculate the total number of new resources that have been added and the resources that have been removed. Then, we divide the total number based on the embedded resource type (e.g., image, style sheet, javascript), so we have the specific number of additions and removals for each category. For example, the difference between M_{t1} and M_{t2} could be addition of 5 resources (2 javascript files and 3 images) and removal of 2 resources (1 javascript and 1 image).

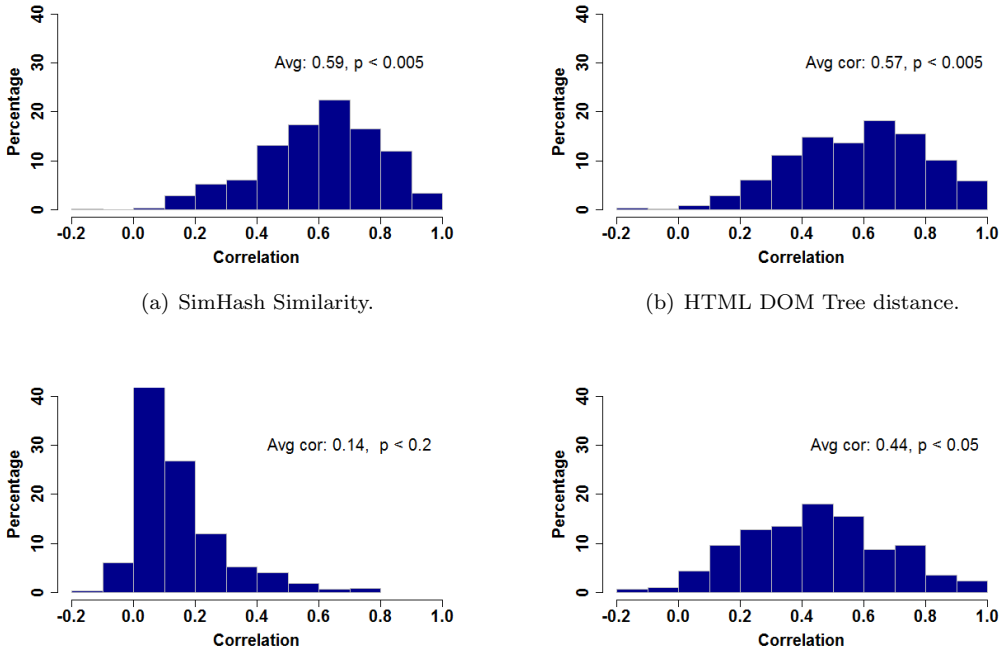
- **Memento Datetime**

  Memento Datetime is the datetime that the memento was crawled (or observed) by the web archive. This information is already available in the CDX file. The similarity is calculated based on the difference in seconds, where a low number indicates high similarity.

Experiment

The success criteria for each feature is how well it can predict the visual difference between two mementos. First, we generate a thumbnail for each memento in each TimeMap. Then, we calculate the difference by comparing the number of different pixels between each pair of thumbnails using

[8] `https://code.google.com/p/boilerpipe/`
[9] `http://jsoup.org/`



(a) SimHash Similarity.

(b) HTML DOM Tree distance.

(c) Memento Datetime

(d) Embedded Resources.

Fig. 40 Histogram for the correlation between Thumbnail difference and various features.

SciPy[10]. In order to compare two thumbnails, we resize them into different dimensions to 64x64, 128x128, 256x256, and 600x600. We calculated the Manhattan distance and Zero distance between each pair. Finally, we calculate the correlation between the similarity of the web pages (based on features in Section 6.3.1) and the difference between the thumbnails.

6.3.2 RESULTS

Figure 40 shows the histogram of the correlation (using Pearson's ρ) (on x-axis) between some features and the image difference calculated by the Manhattan distance using thumbnail size 600x600 and the number of TimeMaps that achieved this level (on y-axis). Generally, all the features showed a positive relationship that range from weak to strong. The best correlation has been found between SimHash fingerprints for the original HTML text, which showed a positive correlation. 70% of the TimeMap has a correlation $\rho \geq 0.5$ (on average $\rho = 0.59, p < 0.005$). The Levenshtein distance between the HTML DOM tree has a high correlation on average ($\rho = 0.57, p < 0.005$). We use the SimHash similarity because the computation is faster than the HTML DOM tree distance.

Our results show interesting features. First, the calculation of the visual difference between the images is not affected by the thumbnail size. The results are consistent between the different thumbnails sizes. Second, we repeat the SimHash calculation on different parts of the web page. The

[10] `http://docs.scipy.org/doc/scipy/reference/index.html`

SimHash similarity between the full HTML text shows the highest correlation over the rest of web page parts. Third, the datetime and embedded resources show a weak positive correlation, however it can be used with other strong features to get a better prediction.

6.3.3 SELECTION ALGORITHMS

In this section, we discuss three algorithms to select a list of representative thumbnails for individual TimeMaps.

Threshold Grouping

In this algorithm, we use feature f for TimeMap $TM = \{M(t_1), M(t_2), ..M(t_n)\}$. The initial step is to divide the TM into groups G, where each group has only two subsequent mementos $M(t_i)$ and $M(t_{i-1})$. For each G, we compute the $diff(f)$ between $M(t_i)$ and $M(t_{i-1})$. If $diff(f, M(t_i), M(t_{i-1})) < \alpha$, we will eliminate one of the mementos M in the group. Then, the new list is sorted by time and we repeat the grouping again. We will continue the process until we reach that for every pair of mementos $diff(f, M(t_i), M(t_{i-1})) > \alpha$. Figure 41 shows the first step of the threshold grouping.

The value of α can be configured and may depend on other factors. Figure 42 shows the relationship between the change in the SimHash threshold (on x-axis) and the reduction of the TimeMap and the loss of image difference (on y-axis). The image loss is calculated as the Manhattan distance between the selected thumbnail and the eliminated thumbnail. We defined the optimum point as the smallest TimeMap size with the least image difference loss. Our empirical study shows that for SimHash similarity, $\alpha = 0.05$ is the optimum value where it decreases the TimeMap to 27% of its original size with the loss of 27% of the image differences.

We notice here that with SimHash threshold at 0, which means removing all the duplicate snapshots only, we still have loss in image differences. It happens because even if the HTML text may be the same, the rendered image may not be visually identical. The reason for that on the live web may be different advertisements, different results from JavaScript (e.g., picking a random image each time), etc. The web archive environment adds more causes such as the embedded resources may not be archived, change of embedded resources (e.g., update in style sheet without changing the URI), or the web archive server can not render the page at that time (cf. "Sorry, no thumbnail yet" in Figure 38).

K Clustering

In this algorithm, the web archive will select K thumbnails for each TimeMap. K could be absolute, relative, or an expression based on other parameters (e.g., rate of change, crawling frequency). The algorithm will depend on a set of features F. To get the most representative K, we apply the "*K-Medoids*" [224] on each TimeMap to cluster the mementos into K clusters. For each cluster, we select a random thumbnail to represent the cluster. We apply the algorithm on two sets of features, $F_1 = \{SimHash\}$ and $F_2 = \{SimHash, Memento - Datetime\}$, we repeat the algorithm on $K \in \{5 : 200\}$. Figure 43 shows the average sum square error and the average image loss in both

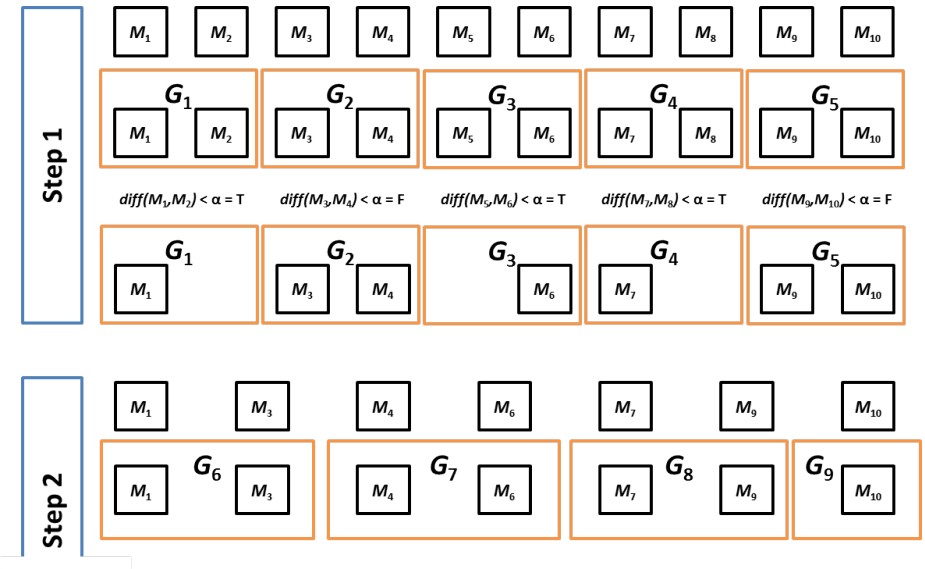


Fig. 41 Threshold Grouping algorithm.

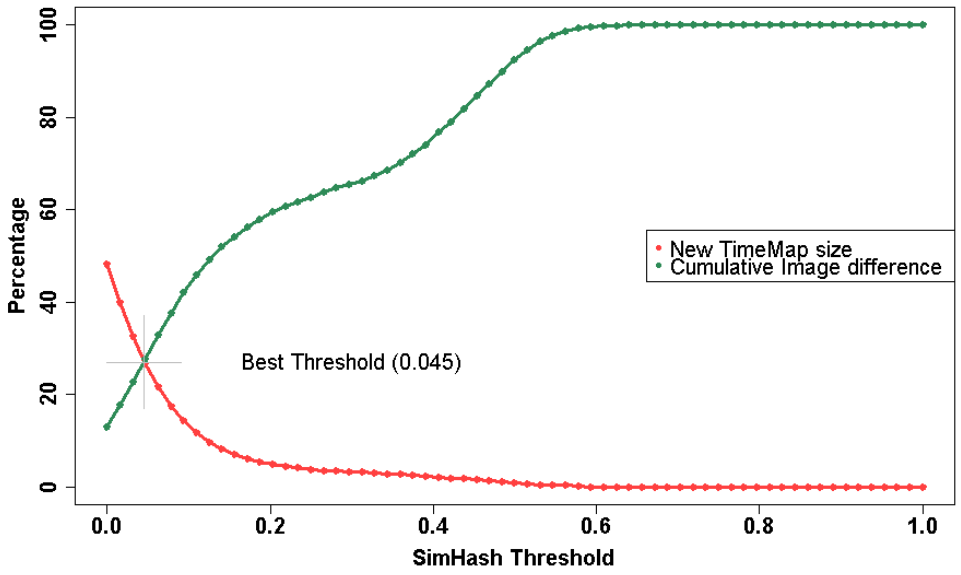


Fig. 42 Optimum SimHash threshold point for the Threshold Grouping algorithm

Table 11 Comparison between the selection algorithms.

	Threshold Grouping	K clustering	Time Normalization
TimeMap Reduction	27%	9% to 12%	23%
Image Loss	28	78 - 101	109
# Features	1 feature	1 or more	1 feature
Preprocessing required	Yes	Yes	No
Efficient processing	Medium	Extensive	Light
Incremental	Yes	No	Yes
Online/offline	Both	Both	Both

cases. The red dots describe the average sum square error for each cluster, the red line is the power regression for the sum square error. The blue dots describe the average image loss, and the blue line is its power regression. The black line is a linear cost function based on the time taken to generate the K thumbnails. Notice that the increasing number of clusters decreases the error value but also it increases the cost of creating the thumbnails. Also, the sum square error and image loss error trends are the same which align with our results at Subsection 6.3.2. Using two features (Figure 43(b)) gives lower error rate in both sum square error and image loss which means a better representation of the TimeMap.

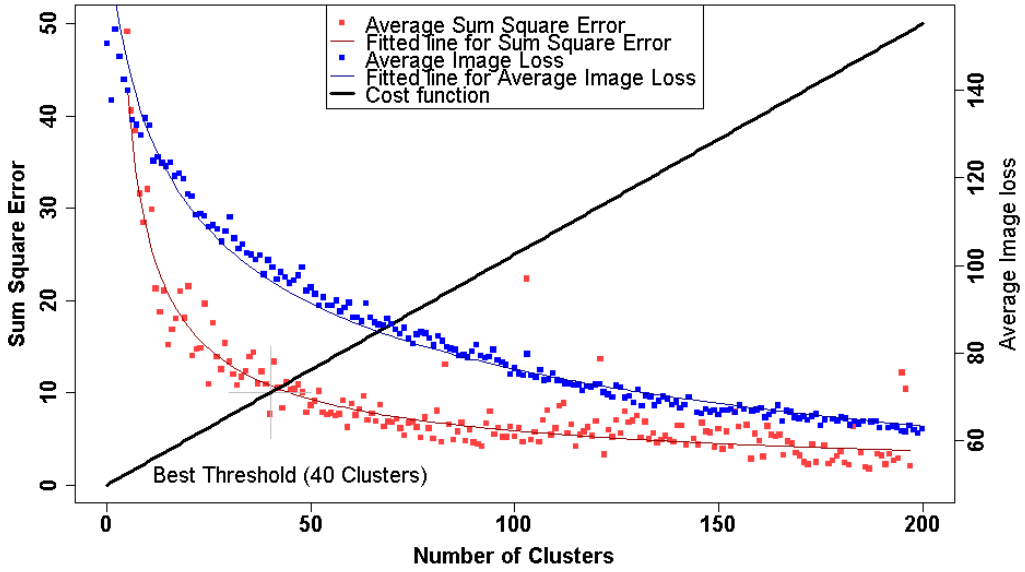
Based on our empirical study, we find the optimal values for $K = 25..40$, which decreases the size of the average TimeMap by 9% to 12% of its original size.

Time Normalization

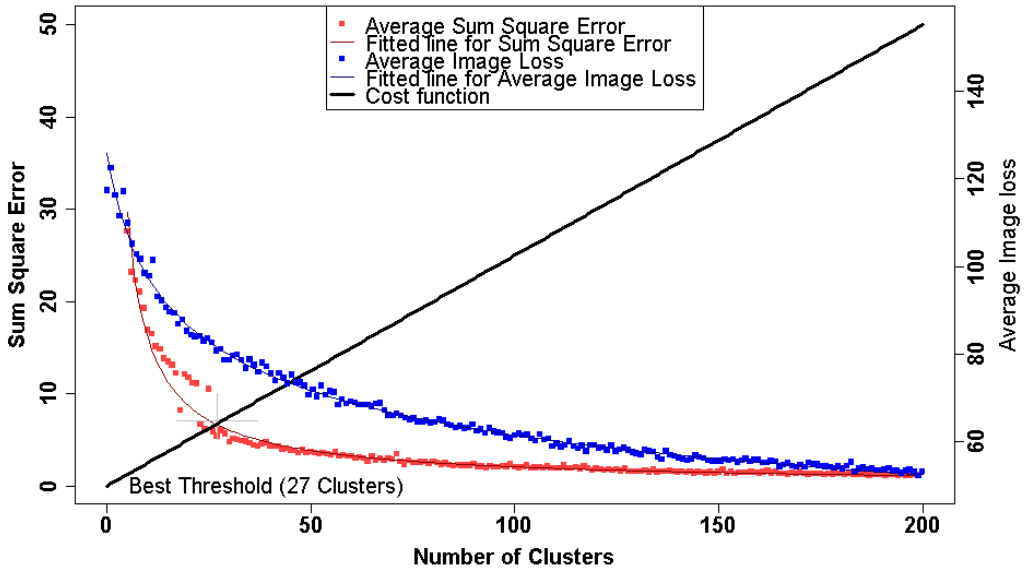
The drawback of the previous algorithms is that they did not take the time dimension into the consideration. In this algorithm, we will apply normalization per time for the TimeMap. We will divide the TimeMap into fixed time-slots T and select random k thumbnails from each slot. The advantage of this algorithm is that it is easy to implement, because it depends only on the CDX files independent from the web pages itself. Figure 44 shows the reduction of each TimeMap after selecting $k = 1$ and $T = 1$ *month*. It reduced the TimeMap size on average to 23% of its original size.

Analysis

Table 11 compares the three algorithms. Even though the three algorithms could be used for online and offline processing, we suggest using Threshold grouping for offline processing because it generates a list of thumbnails that represents the TimeMap that can fit in various applications. K clustering could be suitable for the online processing because it generates the requested K thumbnails that are required by the application, even the extensive nature of clustering algorithms may affect the application performance. Comparing the average image loss of the selected thumbnails in the Time Normalization technique shows higher error rate than the other algorithms which means it gives a poor representation of the TimeMap, however it requires less computation.



(a) Best K using SimHash feature only.



(b) Best K using SimHash and Memento-Datetime features.

Fig. 43 Best K value for K Clustering algorithm.

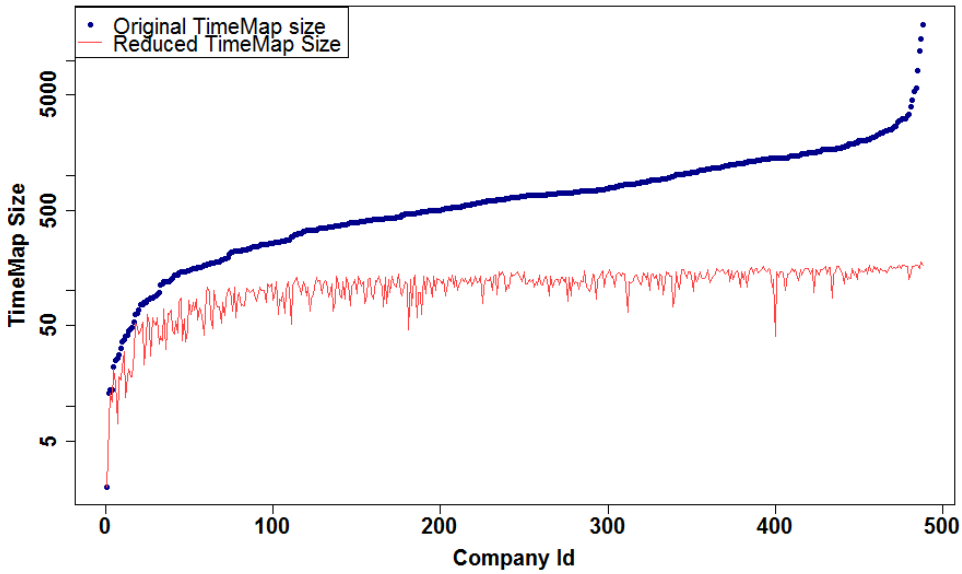


Fig. 44 Reduction of the TimeMap size after applying k thumbnail per time-slot algorithm

6.3.4 RELATED WORK AND BACKGROUND

A few web archives provide a thumbnail view. The UK Web Archive uses thumbnails in timeline and 3D wall visual browsing [144]. Reed archives[11], National Taiwan University Web Archiving System (NTUWAS) [93], and Archive.is provide either a partial or full list of thumbnails for each memento. Archive-It generates on-demand thumbnails as part of the quality assurance process but it is only accessible for their partners. Web Archive Service at California Digital Library[12] uses thumbnails to represent the collections. The usage of thumbnails in the web archives has been studied in a few applications. Soman et al. [254] developed ArcSpread which took a query from the user, extracted related information from the Stanford WebBase [95] using a Hadoop cluster, and displayed the results in spreadsheet style. ArcSpread used both images and web page thumbnails to express the matched web pages. Jatowt et al. [155] proposed *Past Web Browser* to present TimeMap page snapshots as a slideshow of memento thumbnail. It has been extended to *Page History Explorer* [159] for visualizing and comparing the history of web pages. Zoetrope [33] provided a timeline visualization to show the duration and frequency of web pages on specific website. Padia et al. [222] implemented new visualization techniques for Archive-It collections entitled *image plot with histogram* in which they represented each page with its thumbnail.

Thumbnails help with the presentation of temporal data. Tsang et al. [276] proposed the concept of *Temporal Thumbnail* where they included the time dimension into the space of 3D model. They showed that the Temporal Thumbnails were effective for quickly analyzing large amounts of

[11] `http://www.reedarchives.com/`
[12] `http://webarchives.cdlib.org/`

viewing data. Stoev and Straßer [264] studied the visualization of historical data in existence of time dimension. They studied two models; navigation with 4D, and navigation with one fixed dimension.

Thumbnails have applications on the web. `SimilarWeb.com` and `StumbleUpon.com` have recommendation pages function that uses the page thumbnails to represent the recommended pages to help the user to determine the page relevancy. Janssen [153] used fixed thumbnail per document (called an Icon Document) to visualize the search results from UpLib digital library system[13]. Janssen studied the computation of the icon size and the decoration of the icon with labels (e.g., creation date).

Thumbnails can enhance users' ability to find information. Lam and Baudisch [182] proposed a *Summary Thumbnail* that generated a customized thumbnail for web pages to increase the readability of the text on embedded/small screen devices. They discovered that the preservation of the page layout allowed the users to better detect useful information. Woodruff et al. [289] showed that using a mixture of textual and image representations of the web page increased the user prediction of the page effectiveness.

Aula et al. [49] compared thumbnail and textual summaries by surveying the users among different types of page previews. The study showed that the combination of textual information with the thumbnails increased the user estimation of the page content and usefulness. Also, they found that the recognition improved when the thumbnail size was 200x250 pixels. Kaasten et al. [161] and Teevan et al. [268] discovered that thumbnails of size 208x208 pixels or above are effective to remember an exact page.

The selection of the representative image from a set of images depends on comparison techniques between the images itself. These techniques are different from our proposed techniques as we depend on the HTML text to select the representative thumbnail image. AutoAlbum [229] clustered personal photographs into album using time-based and content-based clustering. Coelho and Ribeiro [99] used image filter to get an abstracted version of the images to reduce the dataset to 10% of the original size. Chu et al. [97] studied the near duplicate detection technique for images to cluster a set of images and select a representative image from them. Kherfi and Ziou [169] used the image clustering to organize a large set of images and to provide the user with an overview of the collection's content. They used probabilistic models based on predefined keywords or visual features. Graham et al. [128] studied a set of images that had an attached timestamp (e.g., images from a digital camera). They provided various clustering and summarization techniques such as summarization by time and clustering based on the time and location. They developed a calendar browser with specific 25 photos per panel. They selected the images based on the number of images each month.

6.4 SUMMARY

In this chapter, we gave an overview about the metadata for web archives content. We explored two derived metadata fields. First, we presented ArcLink, a distributed system to construct, preserve and deliver the temporal web graph for large-scale web archives. ArcLink extended the current Wayback Machine with APIs interface to access the link structure metadata on fine-grained level. ArcLink optimization techniques reduced the input corpus to 29%, extracted efficiently from the

[13] `http://uplib.parc.com/`

WARC files with 61% of the regular page scraping, and deliver the link structure in RDF/XML format. ArcLink provided an adequate platform for new applications. Temporal PageRank had weak relation between the rank at each month, and Time-Indexed Inlinks gave information about URI through the time. Second, we studied effective methods to generate thumbnails for the web archive corpus. We explored various similarity features, we found that SimHash and Levenshtein distance between HTML DOM trees had $\rho = 0.59$ and $\rho = 0.57$ correlation with the visual difference between two mementos.‘ We suggest using SimHash as it is efficiently computed from the HTML text. We proposed three algorithms to select K representative thumbnails from the TimeMap. The algorithms decreased the TimeMap size from 9% to 27% of its original size with the minimum loss in thumbnails differences. The techniques could be used with online and offline processing.

CHAPTER 7

URI SERVICE

7.1 INTRODUCTION

HTTP [117] supports redirection using 3xx status codes, which indicate further action needs to be taken by the user-agent in order to fulfill the request. The resource has been moved temporarily (302, 307) or permanently (301) to another URI provided in the `Location` response header.

In web archiving, the user-agent must decide if the URI before or after the redirection should be used to access the web archive. For example, URI `http://bit.ly/r9kIfC` provides a redirection to `http://www.cs.odu.edu` via a 301 status code and a `Location` header.

```
curl -I http://bit.ly/r9kIfC

HTTP/1.1 301 Moved
...
Location: http://www.cs.odu.edu/
...
```

Querying the ODU Memento Aggregator with the shortened URI returns a 404 response because this URI has never been archived, while using `www.cs.odu.edu` as the lookup key returns 700+ mementos.

Figure 45 illustrates another example, $URI-R$ `www.draculathemusical.co.uk` has a redirection on the live web to $URI - \overline{R}$ `http://www.dracula-uk.com/index.html`. If we use $URI - R$ as a lookup key, we can find a memento with HTTP redirection (i.e., `http://web.archive.org/web/20020212194020/http://www.draculathemusical.co.uk` redirects to `http://web.archive.org/web/20020212194020/http://www.geocities.com/draculathemusical`). Now, we end up with three original $URI - R$s:

1. www.draculathemusical.co.uk
2. www.dracula-uk.com/index.html
3. www.geocities.com/draculathemusical

In these examples, the client's awareness with the HTTP redirection status code provides a new approach to reach a nearest memento for the requested datetime. On the other hand, using $URI-\overline{R}$ directly could be misleading. For example, the Architecture Department at Oxford Brookes University's $URI - R$ (`http://www.brookes.ac.uk/schools/be/architecture/`) redirects to $\overline{R}$ (`http://www.brookes.ac.uk/about/faculties/tde`). Using $URI - R$ as a lookup key in this example, we reach 30+ mementos where $URI - \overline{R}$ has only one memento. It is difficult to know a

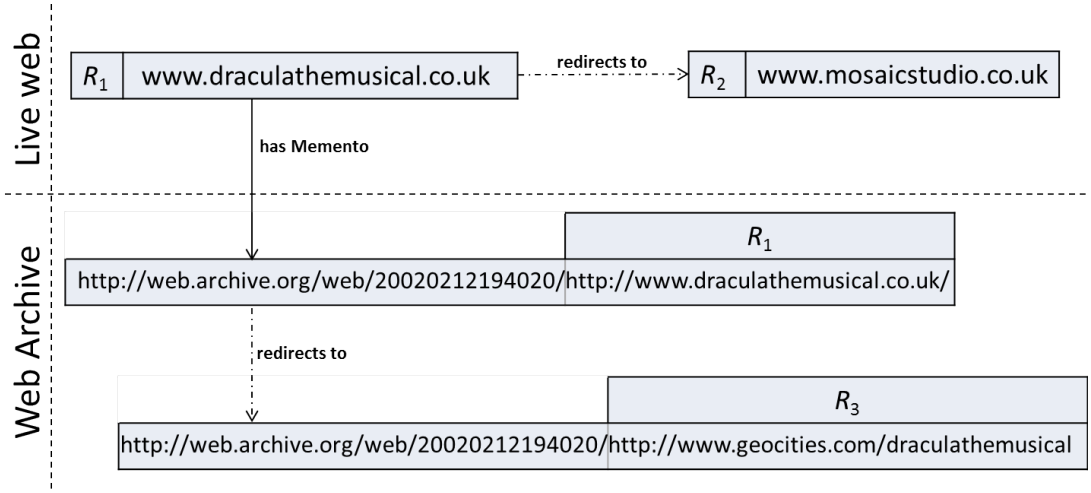


Fig. 45 Redirection in the live web and archived web.

priori which of these two URIs should be used to discover archived copies of the resource. In this research, we study the stability of redirecting $URI-R$s across time. We present new policies that will help the client to use HTTP redirection and obtain a closer Memento to the requested datetime. The first proposed policy (Section 7.5.2) will discuss the different cases that enable the user to use the redirected URIs on the live web instead of the original URIs (i.e., select between 1 and 2 in the previous list). The second policy (Section 7.5.3) will discuss the cases when the user-agent should use the redirected URIs on the archived web instead of the original URIs (i.e., select between 2 and 3 in the previous list).

7.2 ABSTRACT MODEL

Although a lot of research has been done on estimating the frequency of change of a web page [94, 116, 219], no one has focused on the change of the HTTP status code of the URI. In this section, we will discuss the change of HTTP status code through time and the relationship between the live web HTTP status code and the memento HTTP status code.

In this section, $URI-R$ and R denote the original resource; $URI-\overline{R}$ and $\overline{R}$ denote the redirected resources interchangeably. Memento defines the TimeMap TM as a list of the available mementos for $URI-R$ (Equation 10). We extend the Memento TimeMap definition to include $|TM|$ as the size of TM (Equation 11) and $[TM(R)]$ as the timespan for the $TM(R)$ (Equation 12). We add a set of new operations: $Status(M_i(R))$ returns the HTTP status code for $M_i(R)$, and $Location(M_i(R))$ returns the URI in the `Location` header for $M_i(R)$ with HTTP redirection status code.

$$TM(R) = \{M_1, M_2, \ldots M_n\}, \quad where\ M_i = M(R)\ at\ t_i \tag{10}$$

$$|TM| = n, \quad where\ n = \text{ is the number of mementos in } TM. \tag{11}$$

$$[TM(R)] = [datetime(M_1), datetime(M_n)] \tag{12}$$

7.2.1 URI STABILITY

We can determine a URI's stability by examining the HTTP responses across time, and then count the number of changes to the status code (200, 3xx, or 4xx) and the number of different URIs in the `Location` for 3xx status code as appeared in Equation 13.

For example, if $URI - R$ has a TimeMap and all the mementos have HTTP status code 200, we consider $URI - R$ stable through time ($Stability = 1.0$). Also, we can consider $URI - R$ stable if its TimeMap has mementos with status code 3xx to the same "`Location`", in other words, it always redirects to the same $URI - \overline{R}$ through time with $Stability = 1.0$. On the other hand, if $URI - R$ has a TimeMap and each memento has a redirection to a different "`Location`", in this case we consider this $URI - R$ as unstable ($Stability \simeq 0$ for large TimeMap), because $URI - R$ redirects to different $URI - \overline{R}$ through time.

$$Stability(R) = 1 - \frac{\sum_{M \in TM} Change(M_i, M_{i-1})}{|TM|} \tag{13}$$

$$Change(M_i, M_{i-1}) = \begin{cases} 1 & Status(M_i) \neq Status(M_{i-1}) \\ & or\ Location(M_i) \neq Location(M_{i-1}) \\ 0 & \text{otherwise} \end{cases}$$

$$where\ |TM| > 0$$

We present four categories of TimeMaps and discuss the stability for each one. These categories focus only on the HTTP status codes of the mementos excluding the HTTP status code of the original URI on the current web.

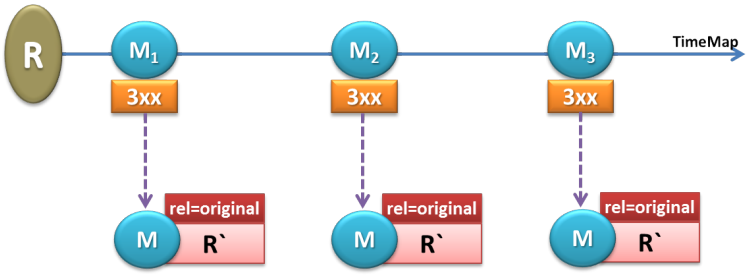
Figure 46 illustrates the different categories. The horizontal line represents the TimeMap for the resource R in the left-hand oval. The circle represents a memento; the attached small rectangle represents the HTTP status code of this memento. If the status code is 3xx, a dashed arrow will represent the redirection to another memento. The big rectangle carries the original resource that belongs to this memento.

All Mementos have 200 HTTP status code

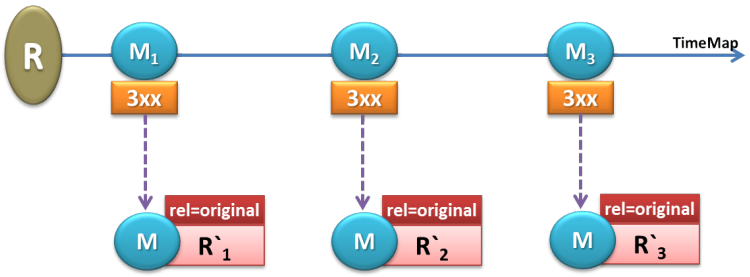
The TimeMap $TM_1(R)$ is a list of available mementos M_i for the resource R where each memento carries HTTP response code 200 as illustrated in Figure 46(a).

$$TM_1(R) = \{M_1, M_2, \ldots M_n\} \quad where\ Status(M_i) = 200$$

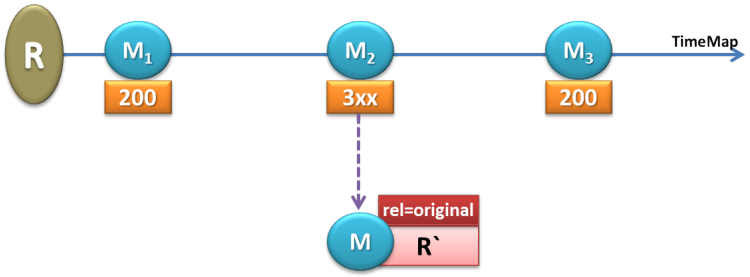
(a) All Mementos have 200 HTTP status code ($Stability = 1$).



(b) All Mementos have redirection to the same URI ($Stability = 1$).



(c) All Mementos have redirection to different URIs ($Stability \approx 0$).



(d) Mementos have different HTTP status code($Stability$ =?).

Fig. 46 Timemap Redirection Categories.

For this TimeMap category, we calculate the stability as 1.0 because $URI - R$ did not change through time.

All Mementos have redirection to the same URI

TimeMap $TM_2(R)$ is a list of available mementos M_i for the resource R where each memento has HTTP redirection status code. Each $M(R)$ redirects to $M(\overline{R})$ for all the mementos in TimeMap as illustrated in Figure 46(b).

$$TM_2(R) = \{M_1, M_2, \dots M_n\} \quad where\ Status(M_i) = 3xx$$
$$\forall\ M_i(R)\ \exists M_j(\overline{R})\ where\ M_i(R) \rightarrow M_j(\overline{R})$$

This category describes this set of URIs that have a redirection status code that have not changed over time. For example, bit.ly/xxx URIs do not change over time. The stability for such $URI - R$ is 1.0 because it redirects to one $\overline{R}$ through time. Stability is a function of redirection so it is possible to have a stable TimeMap that never returns a 200 response code.

All Mementos have redirection to different URIs

TimeMap $TM_3(R)$ is a list of available mementos M_i for the resource R where each memento has a redirection status code to mementos that belong to the same or different $\overline{R}$ as illustrated in Figure 46(c).

$$TM_3(R) = \{M_1, M_2, \dots M_n\} \quad where\ Status(M_i) = 3xx$$
$$\forall\ M_i(R)\ [status : 3xx]\ \exists M_j(\overline{R})\ where\ \overline{R} \in \left\{\overline{R}_1, \overline{R}_2, \dots \overline{R}_n\right\}$$

In this case, $URI - R$ was not stable over time, as $URI - R$ redirects to various $URI - \overline{R}$ through time. Here, stability will asymptotically approach 0.

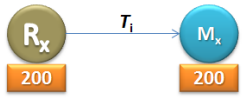
Mementos have different HTTP status codes

TimeMap $TM_4(R)$ is a list of available mementos M_i for the resource R where each memento may or may not have a redirection status code. In the existence of the HTTP redirection status code, it could be to the same or different $URI - \overline{R}$ as illustrated in Figure 46(d).

$$TM_4(R) = \{M_1, M_2, \dots M_n\} \quad where\ Status(M_i) = xxx$$
$$and\ xxx \text{ is a valid HTTP response code}$$

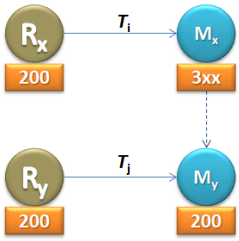
7.2.2 URI RELIABILITY

Even though the stability gives us a good indication about the status code change of the $URI - R$ through time, it does not necessarily indicate the ability to retrieve the mementos successfully. We can categorize the $M_i(R)$ into two categories: successful retrieval, where the memento has HTTP status code 200 or a redirection chain ends with 200; and unsuccessful retrieval, where the memento has 4xx/5xx or a redirection chain that ends with 4xx/5xx. We define URI reliability as the ratio between the number of successful mementos to the total number of mementos per TimeMap.



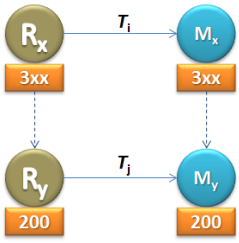
R_x	www.cnn.com
R_y	20110729013512/http://www.cnn.com

(a) Case 1 (80.83%)



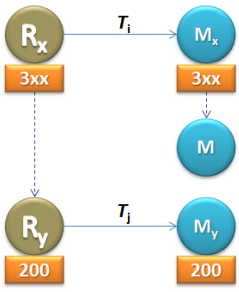
R_x	www.abcsystems.com/
M_x	[ARCBASE]/20040612004302/http://www.abcsystems.com/
R_y	http://www.abcsystems.com/content/abc/
M_y	[ARCBASE]/20040612004302/http://www.abcsystems.com/ content/abc/

(c) Case 2 (2.74%)



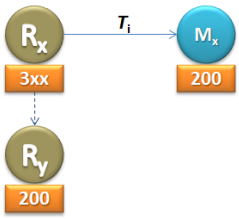
R_x	bit.ly/2EEjBl
M_x	[ARCBASE]/20101109032705/http://bit.ly/2EEjBl
R_y	www.cnn.com
M_y	[ARCBASE]/20101109032705/http://www.cnn.com

(e) Case 3 (1.34%)



R_x	draculathemusical.co.uk
M_x	[ARCBASE]/20020212194020/http://www. draculathemusical.co.uk
M	[ARCBASE]/20020212194020/http://www.geocities.com/ draculathemusical
R_y	www.dracula-uk.com/index.html
M_y	[ARCBASE]/20060615010730/http://www.dracula-uk.com/ index.html

(g) Case 4 (1.33%)



R_x	www.emsetal.com/
M_x	[ARCBASE]/20080703195602/http://www.emsetal.com/
R_y	www.emsetal.com/de/index.php

(i) Case 5 (13.73%)

Fig. 47 *URI – R & URI – M* HTTP redirection relationship cases (ARCBASE=http://web.archive.org/web).

Table 12 $URI - R$ & $URI - M$ Relationship.

		Live Web ($URI - R$)	
		OK	Redirection
Web Archive ($URI - M$)	OK	Case 1	Case 5
	Redirection	Case 2	Case 3, 4

$$Reliability(R) = \frac{\#Mementos\ end\ 200}{|TM|} \tag{14}$$

$$where\ |TM| > 0$$

7.2.3 HTTP REDIRECTION RELATIONSHIP BETWEEN $URI - R$ & $URI - M$

In this section, we study the relationship between the HTTP status code for the original resource ($URI - R$) and the memento ($URI - M$) which we classify into five cases, shown in Table 12. The column represents the status code on the live web for $URI - R$ and the row represents the status code on the web archive for $URI - M$. Both of cases three and four have redirection for $URI - R$ and $URI - M$. If both of Original and Memento redirect to the same $URI - \overline{R}$, it will be case 3, otherwise it is case 4.

Figure 47 illustrates these cases. The circle on the left side represents the $URI-R$ in the live web, and the circle on the right side represents its memento at time t_i. The small rectangle represents the HTTP status code. The dashed arrow represents the redirection between two $URI - R$s or two mementos.

7.3 EXPERIMENT

To quantify our abstract model (Section 7.2), we randomly sampled 10,000 URIs from Open Directory Project (DMOZ) in January 2012. Table 13 shows the distribution of HTTP status code on the current web for our 10,000 sampled URIs. The sample set does not include any shortened [46] nor DOI URIs [225], which we consider as a special case and we will include them in future research.

First, we used the Memento Aggregator to retrieve $TM(R)$ for each $URI - R$. For each $R \rightarrow \overline{R}$, we also retrieved $TM(\overline{R})$ for $\overline{R}$. Second, for each M in $TM(R)$, we retrieved its HTTP status code ($Status(M)$). For the mementos with redirection (i.e., $M(R_x) \rightarrow M(R_y)$)), we followed the redirection location and recorded the destination $M(R_y)$, then extracted its original resource R_y. In order to compare the URIs, we performed a canonicalization routine to ensure consistency.

7.4 RESULTS

From 10,000 URIs sampled from DMOZ, we found 8,903 URIs returned TimeMaps for a total of 894,717 mementos. The distribution of HTTP status codes for the mementos list is shown in Table

Table 13 Sample URI Current HTTP status code

HTTP Status/Code	**Percentage** (10,000 URI-R)
OK (200)	82.83%
Redirection (3xx)	14.71%
Redirection (301)	8.4%
Redirection (302)	6.1%
Redirection (others 3xx)	0.2%
Not Found (4xx)	1.18
Others	1.28

Table 14 Mementos HTTP status code

HTTP Status/Code	**Percentage** (894,717 URI-M)
OK (200)	93.46%
Redirection (3xx)	5.69%
Not Found (4xx)	0.26%
Others	0.59%

14. The table shows that nearly 6% of the mementos have archived redirects (i.e., $URI - M$ with 3xx HTTP status code), where $URI - R$ had a redirection status code at crawling time.

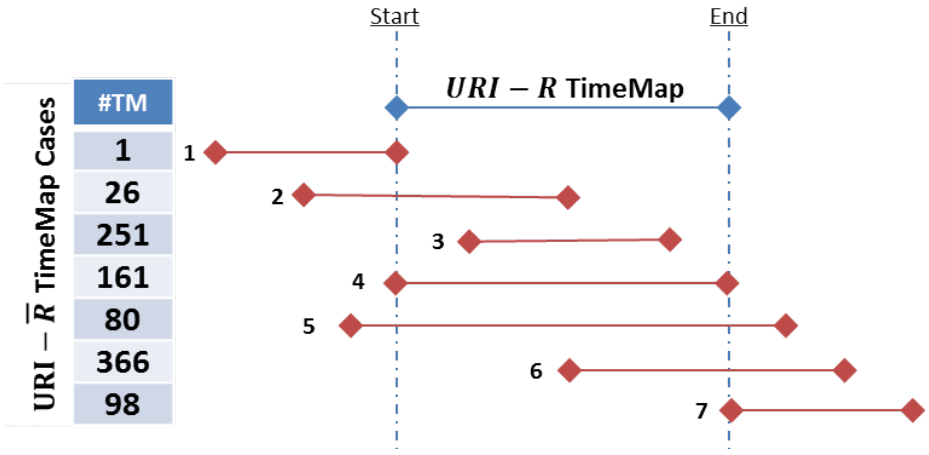
7.4.1 RELATIONSHIP BETWEEN $TM(R)$ AND $TM(\overline{R})$

In this section, we assume $URI - R$ redirects to $URI - \overline{R}$ on the live web, so we will compare the TimeMaps for $URI - R$ and $URI - \overline{R}$.

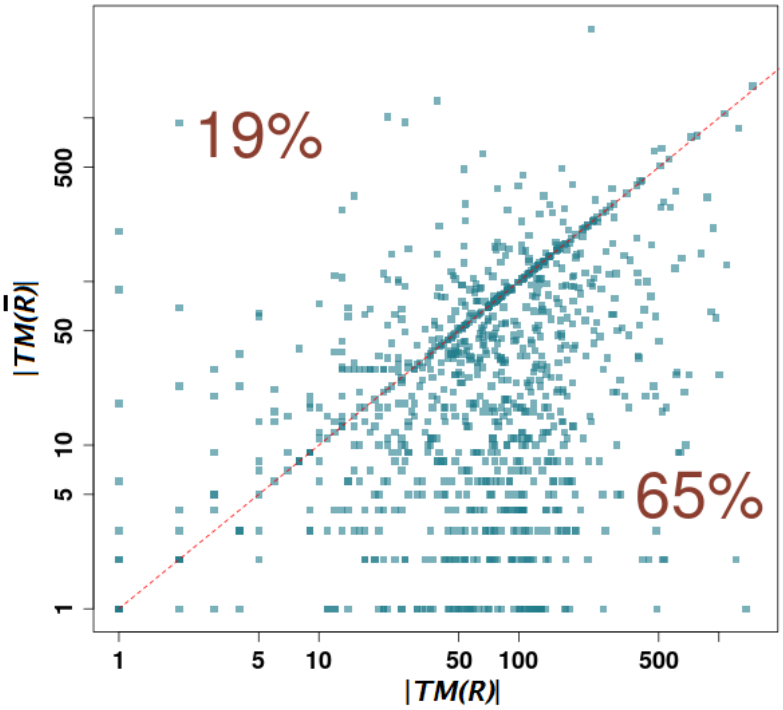
Figure 48(a) illustrates the relationship between $[TM(R)]$ and $\left[TM(\overline{R})\right]$. Each case is defined based on the first and the last Memento-Datetime in the TimeMap for R and $\overline{R}$, it may indicate the lifetime of the URI. For example, in case 1, $TM(\overline{R})$ starts and ends before the beginning of $TM(R)$. TheFigure lists the number of TimeMaps that occurred in each of the seven cases. In case 4, both of $TM(R)$ and $TM(\overline{R})$ are the same. This case occurs when the redirection does not affect the canonicalized form of R (e.g., `http://example.org` redirects to `http://www.example.org`), the web crawler considers both of them as one URI. Cases 1, 2 and 5 have low numbers, which means that the existence of $\overline{R}$ was related to the existence of R first. Cases 5, 6, and 7 shows the continuous existence of the $\overline{R}$ on the web even after the disappear of the R.

Figure 48(b) shows the relationship between the number of mementos for the original resource $TM(R)$ (x-axis) and the number of mementos for the redirected resource $TM(\overline{R})$ (y-axis). The red dashed line shows the cases where $|TM(R)| = |TM(\overline{R})|$ 16% of the time. In 65% of the cases, $|TM(\overline{R})| < |TM(R)|$.

7.4.2 URI STABILITY



(a) Timespan of TimeMap.



(b) Number of Mementos (16% on the diagonal).

Fig. 48 The relationship between TimeMap for the Original ($URI - R$) and the Redirected ($URI - \overline{R}$).

Table 15 Temporal TimeMap Redirection categories.

Timemap Category	**%**	**Stability**
All Mementos have OK	52%	1.0
Mementos have mix status code	36%	0.91
All Mementos have Redirection	0.92%	0.85
Redirection to the same URI	0.62%	
Redirection to different URI	0.30%	
URI has no Mementos at all	10.97%	0.0

Figure 49 shows the relationship between the stability of $URI - R$ (x-axis) with the number of mementos in its TimeMap (y-axis). The results show that 48% of the URIs are not perfectly stable across time. TheFigure shows that a large number of mementos have high $Stability \simeq 1$.

By grouping the memento's status code per TimeMap, we can quantify the different categories and calculate the average stability for each category. Table 15 shows that 52% had 200 status code for all mementos with $Stability = 1.0$. Also, 0.62% of the URIs have redirection to the same original URI with $Stability = 1.0$.

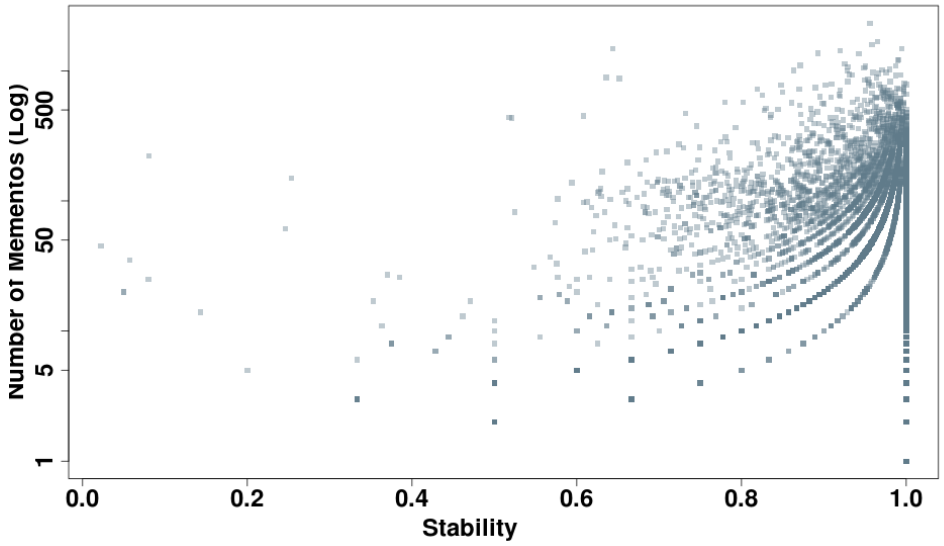
7.4.3 URI RELIABILITY

Figure 50 shows the relationship between the $URI - R$ reliability (x-axis) and the number of mementos for each TimeMap (y-axis) that contain at least one memento that has a redirection status code (2890 URIs out of 10,000 URIs). TheFigure shows the distribution of the reliability, we could not conclude a relationship between the reliability and the number of mementos. We calculate the redirection chain (the number of URI that should be followed before reaching 200 status code) and find that 23% do not lead to a successful memento at the end. A few mementos (0.63%) have infinite redirection chains (50+ redirections).

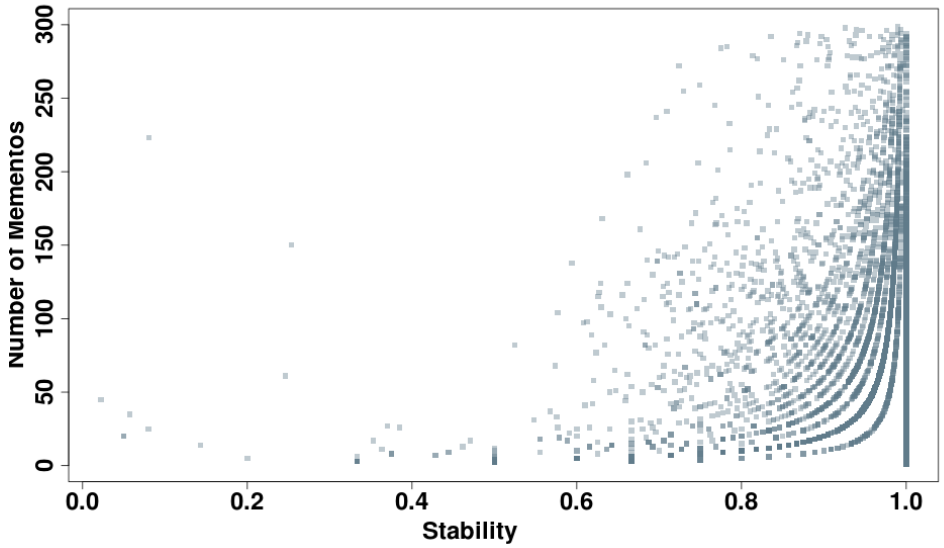
7.4.4 HTTP REDIRECTION RELATIONSHIP BETWEEN $URI - R$ & $URI - M$

In this section, we compare between the live web ($URI - R$ status code) and the archived web (TimeMap and $URI - M$ status codes). Table 16 shows the distribution of the cases of the relationship between the $URI - R$ and its mementos as illustrated in Figure 47. In 19% of the mementos, the client faces HTTP redirection that requires an advanced mechanism to deal with the existence of HTTP redirection in both the live and archived web.

Table 17 shows the relationship between the status code on the live web and the status code of the TimeMap. Even though 1,471 URIs have HTTP redirection in the current web, only 83 TimeMaps have HTTP redirection status codes for all the mementos, while there are 425 TimeMaps with 200 HTTP status code in all the mementos. We can conclude that the HTTP status code on the live web does not give an indication about the status code of the TimeMap because the URI's HTTP status code may change through time. During the experiment, we were not able to deduce a pattern for the URI's HTTP status code change. This quantitative analysis shows the importance

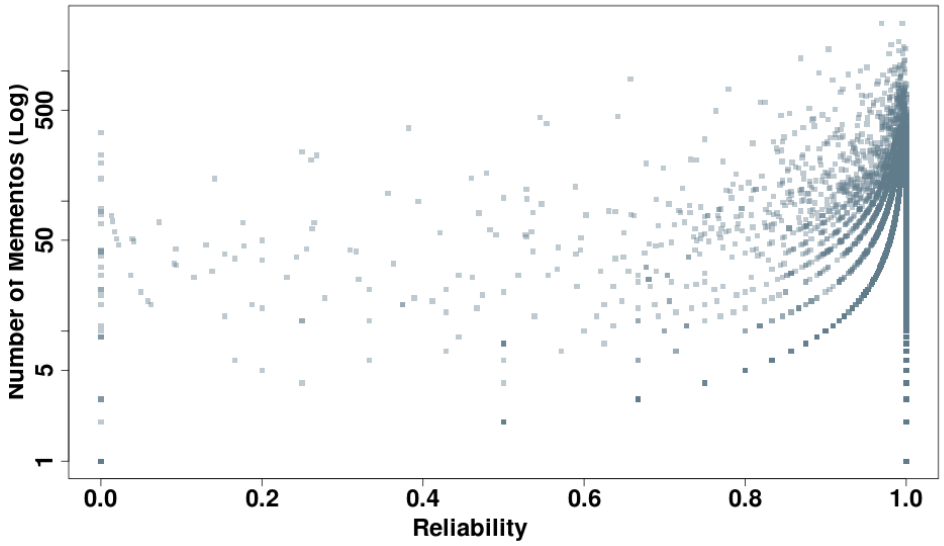


(a) Stability in semi-log scale.

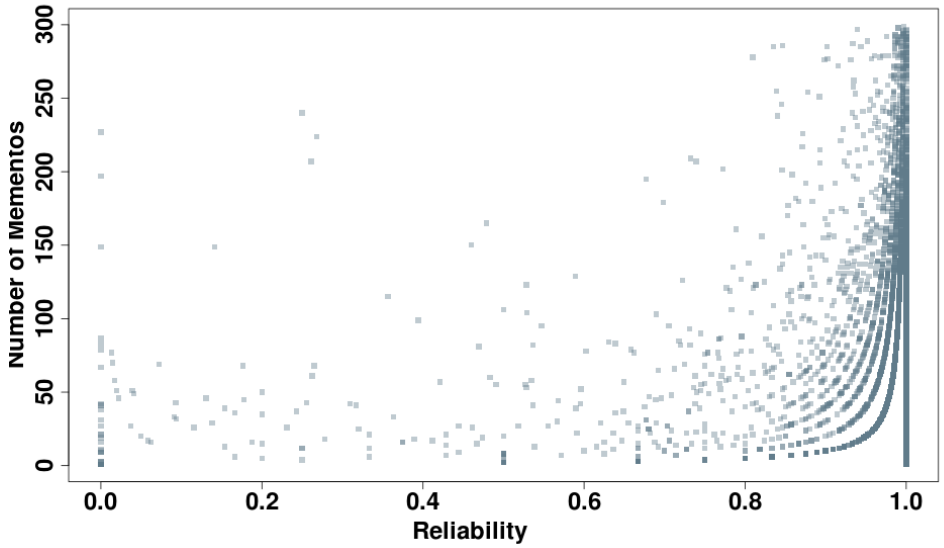


(b) Stability for $|TM(R)| < 300$.

Fig. 49 URI Stability



(a) Reliability in semi-log scale.



(b) Reliability for $|TM(R)| < 300$.

Fig. 50 URI Reliability

of finding new policies, instead of the straightforward $URI - R$ lookup.

Table 16 URI-R - Memento HTTP Redirection relationship cases.

Case	URI-R status code	URI-M status code	Percentage
1	Non-Redirection	Non-Redirection	80.83%
2	Non-Redirection	Redirection	2.74%
3	Redirection to R_x	Redirection to R_x Mementos	1.34%
4	Redirection to R_x	Redirection to R_y Mementos	1.33%
5	Redirection	Non-Redirection	13.73%

Table 17 Timemap status compared to the $URI - R$ status on the live web.

URI Status	Count	Timemap Status	Count
OK	8283	Mixed status	1849
		All 200 status	5886
		All Redirect status	14
		No Mementos	534
Redirection	1471	Mixed status	880
		All 200 status	425
		All Redirect status	83
		No Mementos	83
Not found	118	Mixed status	32
		All 200 status	75
		All Redirect status	2
		No Mementos	9
Others	128	All 200 status	79
		Mix status	30
		No Mementos	19

7.5 ARCHIVED HTTP REDIRECTION RETRIEVAL POLICIES

In this section, we develop new policies to query the web archive with a URI that has a HTTP redirection status code. First, we will describe the current Internet Archive Wayback Machine retrieval policy. Then, we will give two policies: $URI - R$ with an HTTP redirection status code; and $URI - M$ with an HTTP redirection status code.

7.5.1 CURRENT WAYBACK MACHINE BEHAVIOR

The Wayback Machine requires a $URI - R$ to begin exploration of the past web, returing what is essentally an HTML TimeMap, without to possible redirection for $URI - R$. Also, the Wayback Machine replays the resource as it appeared in the past. If the memento has HTTP redirection status code, Wayback Machine displays a message with "Got HTTP 3xx response code at crawl time", then it redirects to another URI-M. Figure 51 is a screenshot for the Wayback machine while retrieving an archived bit.ly URI. When Wayback Machine receives a query of URI that has a HTTP status redirection, it ignores the status of the URI and search for the mementos for this URI.

7.5.2 POLICY ONE: $URI - R$ WITH HTTP REDIRECTION

In our sample data, 1,471/10,000 $URI - R$ redirected to $URI - \overline{R}$, which covers cases three, four and five from Table 16. The proposed policy is as following:

Required: R and "`Accept-Datetime`" header.

1. Retrieve the memento for R.
2. (a) If the retrieved memento has 200 OK HTTP status code, then return this memento. (Stop)
   (b) Else if the retrieved memento has 3xx Redirection HTTP status code, go to policy two.
   (c) If the retrieved memento is unavailable (4xx/5xx HTTP status code) and R has a redirection to $\overline{R}$, use $\overline{R}$ instead of R then go to step 1.

7.5.3 POLICY TWO: $URI - M$ WITH HTTP REDIRECTION

Here, we address case two from Table 16. Assume memento $M(R_x)$ redirects to another memento with a different original $\overline{M}(R_y)$. For example, redirection from a memento for `http://bit.ly/2EEjBl` on 2010-11-09 to another memento for `http://www.cnn.com` on the same date, as shown inFigure 51.

```
curl -I http://web.archive.org/web/20101109032705/http:
   //bit.ly/2EEjBl
HTTP/1.1 301 Moved Permanently
Memento-Datetime: Tue, 09 Nov 2010 03:27:05 GMT
...
Location: http://web.archive.org/web/20101109032705/http:
```

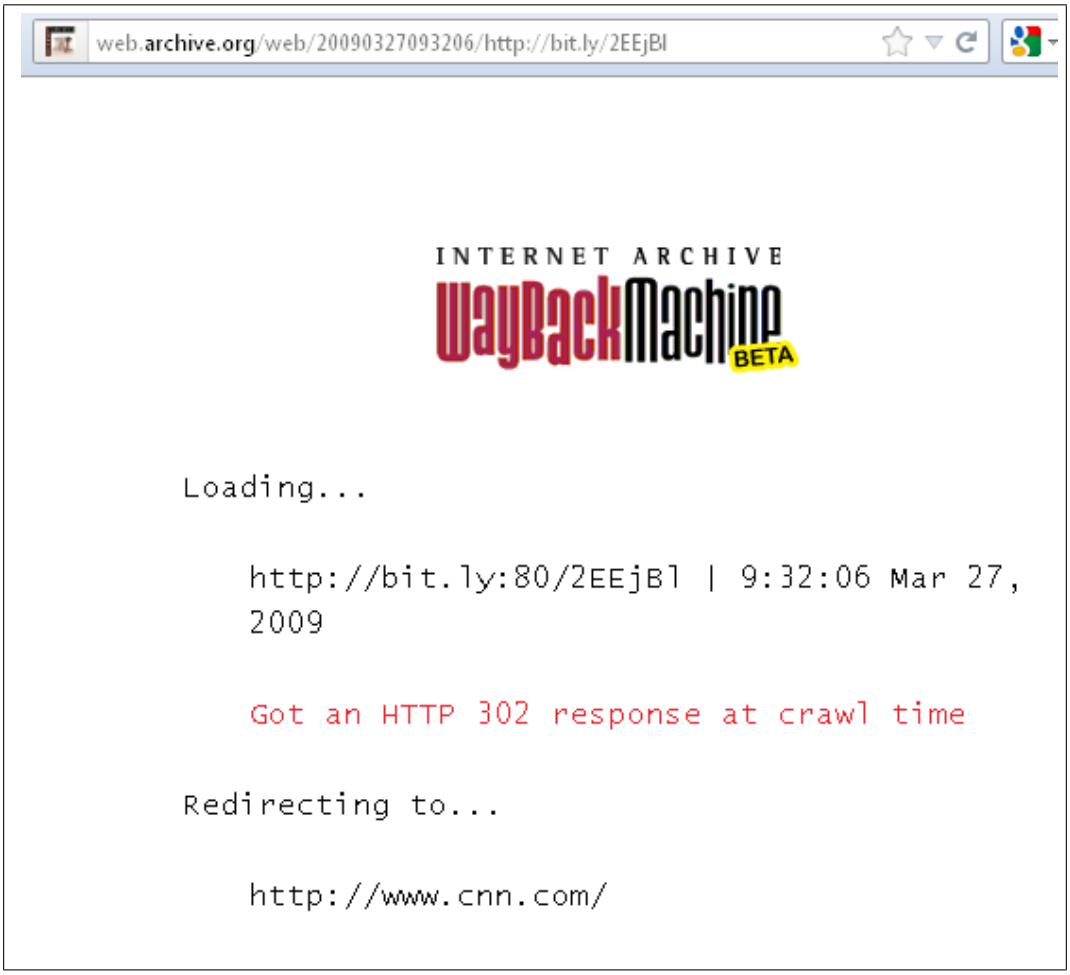


Fig. 51 Wayback Machine is replaying a memento with HTTP redirection status code

```
   //www.cnn.com/
...
```

In this case, the client will repeat the content negotiation in the datetime dimension for the "`rel=original`" R_y extracted from the "`Link`" header for $\overline{M}(R_y)$. For the previous example, the client should repeat the content negotiation with `www.cnn.com` with the requested datetime on 2010-11-09.

Some web archives do not rewrite the memento "`Location`" header, so the memento could redirect to another original resource on the live web. In this case, policy two will redo the content negotiation using the new original resource instead of redirecting to the live web.

The new policy extends the default Wayback Machine behavior by retrieving the nearest memento to the redirected $URI-\overline{R}$ which may not be available from $URI-R$. Also, applying the new policy on the Memento Aggregator will benefit from the multi-archive environment which may find a better copy in another archive.

7.5.4 EVALUATION

Policy one: $URI - R$ with HTTP Redirection

This policy focuses on the 1,471 URIs from our sample that have HTTP redirection on the live web. We find 77 URIs that have no mementos at all ($|TM(R)| = 0$). Based on this policy, we can retrieve mementos for 17 URIs out of that 77 URIs where $|TM(\overline{R})| > 0|$.

Policy two: $URI - M$ with HTTP Redirection

We have 2980 TimeMaps that showed HTTP redirection status code in at least one memento. For these TimeMaps, we follow the memento redirection and extract the original URIs. We extract 7115 URIs. The evaluation criteria for this policy is defined by the number of the cases that the policy will contribute to the TimeMap.

Assume that $TM(R) = \{M_1(R), M_2(R), \ldots M_n(R)\}$. For each $M(R)$ with an HTTP redirection status code, we have $M(\overline{R})$ where $M(R) \rightarrow M(\overline{R})$. In this case, the policy will contribute to the $TM(R)$ if the $TM(\overline{R})$ covers a larger time frame (i.e., $\left[TM(\overline{R})\right] > [TM(R)]$).

From our sample, the policy contributes more mementos to the original TimeMap in 58% of the cases. In the rest of the cases $|TM(\overline{R})| < TM(R)$.

Discussion

The existence of the HTTP redirection supports the retrieval process with the required information to reach a better estimation of the presentation of this URI in the past. The policies' evaluation show the ability to deliver new mementos that were unreachable using the regular methods. Both policies redo the content negotiation for the redirected URIs (on live or archived web). Policy one uses the live redirect if there are no mementos for the original resource. If there are mementos, policy two will give the priority to the archived redirected because this is what has been recorded by the web archive in the past. Policy one succeeded in 17/77 of the cases. The second policy extends the TimeMap time span to include mementos from the archived redirected URI. So using the preserved redirection information helps the client to find the nearest memento to the requested datetime. These policies could be implemented in the client side. The client should give the user the ability to optionally select between the different policies.

7.6 SUMMARY

In this chapter, we study the change of HTTP status codes through the time with focus on the HTTP redirection. Two novel measurements have been proposed, the stability of the URI and the reliability of the URI. Our experiments showed that URIs are not stable through time. We studied the different categories of the TimeMaps with a focus on HTTP status code. We found that in 36% of the cases the TimeMaps are not fully stable through time. Based on this quantitative study, we introduce two retrieval policies to handle HTTP redirect. The first policy focused on a resource that redirects on the live web; it was successful with 22% of the applicable cases. The second policy

focused on the mementos with HTTP redirection status code; it extended the original TimeMap in 58% of the applicable cases.

CHAPTER 8

ARCHIVE SERVICE

In this chapter, we will study the services that can be applied on the level of the web archive itself. These services include how to differentiate between the web archives and how to use these differences to build better discovery mechanisms for the Memento Aggregator. We will answer two research questions. First, how much of the web is archived? Second, what is the distribution of the web archived materials related to the top-level domains, content languages, and the growth rate? We develop "Archive profiles" for various web archives that optimize the distributed queries performance in the Memento Aggregator.

8.1 THE PERCENTAGE OF THE ARCHIVED WEB

In the area of digital preservation, the question "how much of the Web is archived?" is usually raised. This research question considers the percentage of the live web that has been archived by the available web archives.

In this research, we do not attempt to measure the absolute size of various public web archives because just few web archives report their size (e.g., Internet Archive reported its size 360B $URI-M$ [237]). Instead, we sample URIs from a variety of sources (DMOZ, Delicious, bitly, and search engine indexes) and report on the number of URIs that are archived, and the number and the frequency of their mementos. From this, we extrapolate the percentage of the surface web that is archived. We ran the experiment in late 2010 and repeated it in late 2013. We report the differences in the experiment environment and the results between both runs.

8.1.1 URI DATASET SAMPLES

Discovering every URI that exists is impossible; therefore, representative sampling is required. We prepare samples from four different resources: Open Directory Project (DMOZ), the Delicious Recent bookmark list, bitly shortner service, and from search engines using the code and procedures provided by Bar-Yossef [57]. The reasoning behind these sources and the methods used for URI selection are detailed below. For practical reasons (e.g., search engine query limits and execution time), we selected a sample size of 1,000 URIs for each of the four sample sets. We collected these samples between November 2010 and December 2010.

Open Directory Project (DMOZ) Sampling

Our URI selection from DMOZ differs from previous methods such as Gulli and Signorini [135] in that we use the entire available DMOZ history instead of a single snapshot in time. In particular, we extract URIs from every DMOZ archive available[1], which includes 100 snapshots of DMOZ made

[1] `http://rdf.dmoz.org/rdf/archive`

from July 20, 2000 through October 3, 2010. First, a combined list of all unique URIs is produced by merging the 100 archives. From this combined list, 3,806 invalid URIs (URIs not in compliance with RFC 3986) are excluded, 1,807 non-HTTP URIs were excluded, and 17,681 URIs with character set encoding errors are excluded. This results in 9,415,486 unique valid URIs from which to sample. DMOZ sample is highly biased to the Internet Archive because Internet Archive's crawler uses the DMOZ directory as a seed for site-crawling [19].

Delicious Sampling

The next source for URIs is the social bookmarking site Delicious[2]. Delicious's service started in 2003; it allows users to tag, save, manage, and share web pages from a centralized source. Delicious provides two main types of bookmarks. Recent bookmarks are the URIs that are recently bookmarked by users, and popular bookmarks are the bookmarks that are saved by the most users. In our experiment, we retrieve 1,000 URIs from the Delicious Recent random URI generator[3] on Nov. 22, 2010, however the APIs are different from the web interface [170].

bitly Sampling

The bitly[4] project is a web-based service for URI shortening [46]. Its popularity grew as a result of being the default URI shortening service on the microblogging service Twitter (from 2009-2010), and now enjoys a significant user base of its own. Any link posted on Twitter is automatically shortened and reposted. bitly creates a "short" URI that when dereferenced issues an HTTP 301 redirect to a target URI. The shortened URI consists of a hash value of up to six alphanumeric characters appended to `http://bit.ly/`, for example the hash value `A` produces:

```
% curl -I http://bit.ly/A
HTTP/1.1 301 Moved
Date: Sun, 30 Jan 2011 16:00:48 GMT
Server: nginx
Location: http://www.wieistmeineip.de/ip-address
...
```

The shortened URI consumes fewer characters in short messages (e.g., a tweet), protects long URIs with arguments and prevents from being mangled in emails and other contexts. The shortened URI provides an entry point for tracking clicks by appending a "+" to the URI: `http://bit.ly/A+`. This tracking page reveals when the short URI was created, as well as the dereferences and associated contexts for the dereferences.

For our bitly sample, we created a series of random, candidate hash values for bitly and recorded the target URIs (i.e., the URI in the *Location:* response header) of the first 1000 bitlys that successfully returned HTTP 301 responses. We also recorded the creation time of the bitlys via their associated "`+`" pages.

[2] `http://www.delicious.com`

[3] In 2010, the recent service was `http://www.delicious.com/recent/?random`. In 2013, it was updated to `http://feeds.delicious.com/v2/json/recent`

[4] `http://bit.ly`

Similar to the URIs obtained from Delicious, we assumed the URIs returned by sampling from bitly are biased toward URIs that users have interacted with directly, typically from sharing the URI in a tweet or a similar social media context. It is also possible, considering the nature of a shortening service, that the URIs in this sample are longer in terms of characters in the URI.

Search Engine Sampling

Search engines play an important role in web page discovery for most casual users of the web. We include a sample from search engine as a representative of the indexed web. Removing the bias to page rank and popularity is one of the main challenges in sampling from search engines. Bar-Yossef and Gurevich [57] developed two methods to implement this unbiased random URL sampler from search engine's index. The first method was utilizing a pool of phrases to assemble queries that would be later fed to the search engine. The other approach was based on random walks and did not need a preparation step. The first method was utilized in this research with a small modification to the first phase of pool preparation.

We prepared the query pool by sampling from the Google N-grams query list [119], we chose a size of 5-grams. A total query pool of 1,176,470,663 queries was collected. A random sampling of the queries was provided to the URI sampler as the second phase. A huge number of URIs were produced, and 1,000 were randomly selected.

8.1.2 EXPERIMENT AND RESULTS

The four samples of URIs were used to estimate the percentage of the archived web. We ran the experiment twice in late 2010 and late 2013. The purpose of this experiment is to estimate the percentage of all publicly-visible URIs that have archive copies available in public archives. The list of the available public archives has changed between 2010 and 2013. In 2010, the caches for the major three search engines (Google, Bing, and Yahoo) were accessible. While technically not web archives, search engines caches can sometimes be used to retrieved a preserved copy of the URI [194]. In 2013, various web archives became accessible to through the Memento Aggregator. Table 18 lists the web archives queried in 2010 and 2013. This experiment was accomplished in three parts:

1. Selecting a sample set of URIs that are representative of the web as a whole,
2. Determining the current state of the sample URIs (URI HTTP status, SE index, and estimated age), and
3. Discovering mementos for the sample URIs.

Current URI State Determination

After selecting the URIs, we determined the current HTTP status for each URI on the live web. In 2010, we were able to determine if the URI was indexed in the search engines.

First, the current status of each URI was tested using the `curl` command. The status was classified into one of six categories based on the HTTP response code. We divided the 3xx responses

Table 18 Archives queried

Archive name	2010	2013
Internet Archive	x	x
Archive-It	x	x
UK National Archives	x	x
UK Web Archive by British Library	x	x
Web Cite	x	x
Google	x	
Bing	x	
Yahoo	x	
Diigo	x	
ArchiefWeb	x	
CDLIB	x	
NARA	x	
Singapore Web Archiv		x
Web Archive of Catalonia		x
Croatian Web Archive		x
Archive of the Czech Web		x
National Taiwan University		x
Icelandic Web Archive		x
Library & Archives Canada		x
Slovenia Web Archive		x
Archive.is		x
Library of Congress		x

into two categories because a search engine could carry the redirect information and report the original URI as indexed and display the new URI. Table 19 lists the results of the current status of the URIs on the web. 95% of Delicious and search engine sample URIs sets are live URIs (status code 200), but only 50% of DMOZ and bitly sample URIs sets are live. The reason is that the Delicious sample came from the "Recent" added bookmarks which means these URIs are still alive and some users are still tagging them. Search engines purge their caches soon after they discover URIs that return a 404 response [206].

In 2010, we tested the status of each sample URI to see if it was indexed by each search engine. We used Bing APIs[5], Yahoo BOSS APIs[6], Google Research Search Program APIs[7]. to find the URI in these search engine indexes. It is well-known that the APIs return different results than those from the web interfaces [204]. A significant difference in the indexed results between Google web interface and Google Research Search Program APIs was found. Table 21 lists the number of the URIs indexed by the different search engines. Google indexed status is shown on two rows; the first one is the number discovered by the API, and the second one is the union between the URIs discovered by the APIs and the URIs found by the Memento proxy for the Google cache.

[5] `http://www.bing.com/developers`
[6] `http://developer.yahoo.com/search/boss/`
[7] `http://www.google.com/research/university/search/`

Table 19 Sample URIs status on the live Web.

Status	DMOZ	Delicious	bitly	SE
200	507	958	488	943
3xx⇒200	192	27	243	17
3xx⇒Other	50	1	36	3
4xx	135	8	197	16
5xx	4	3	6	0
Timeout	112	3	30	21

Table 20 Number of mementos per URI.

		2010		
Mementos per URI	**DMOZ**	**Delicious**	**bitly**	**SE**
0 (Not archived)	93	25	648	225
1	46	79	100	336
2 to 5	142	491	171	320
6 to 10	85	35	17	35
More than 10	634	370	64	84
		2013		
	DMOZ	**Delicious**	**bitly**	**SE**
0 (Not archived)	95	45	482	667
1	15	6	18	44
2 to 5	68	12	178	102
6 to 10	65	29	45	60
More than 10	757	908	277	127

Table 21 Sample URIs indexed by the search engines in 2010.

	DMOZ	Delicious	bitly	SE
Bing	495	953	218	552
Yahoo	410	862	225	979
Google$_{(\text{APIs Only})}$	307	883	243	702
Google$_{(\text{APIs+Cache})}$	545	951	305	732

General Coverage

Table 20 shows the distribution of the number of mementos per URI. DMOZ and Delicious samples have the same coverage level in 2010 and 2013. They have many URIs with more than 10 mementos. DMOZ is considered as the main input for the Internet Archive crawler, also the DMOZ sample is retrieved from historical DMOZ archives which means these URIs may have a long age. The Delicious sample had good coverage in 2010, and it greatly increased in 2013. The bitly sample has many URIs that are not covered at all. This means that bitly URIs are not discovered by the different crawlers. This matches the observation in Table 21 of poor coverage of the bitly URIs by the search engine indexes. Figures 52-55 show the histogram for the number of mementos per URI. The search engine sample had good coverage in 2010 based on mementos contributed by search engines. The number of not archived URIs increased in 2013 with the absence of search engines' caches.

Table 22 shows the retrieved mementos and the original resources from each archive. For each archive and sample, we report the number of retrieved mementos and the coverage of the URIs, which is the number of unique URIs that have mementos.

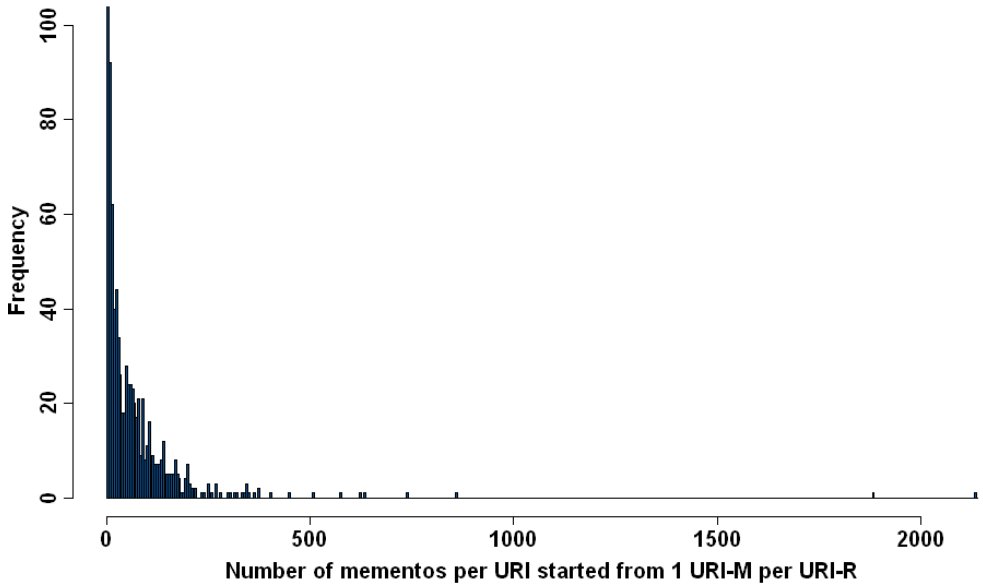
Coverage Distribution through time

Figures 56-59 shows the histogram of the number of retrieved mementos. These figures show the distribution of the mementos through time. These figures also show that there is low coverage for the period before 2000 as could be expected. The end of 2010 has good coverage led by the search engine caches. In 2010, both DMOZ and Delicious samples have a similar distribution. However, the coverage for the Delicious and bit.ly samples has increased on the last three years. The Search Engine sample has a good coverage in late 2010 powered by search engine caches.

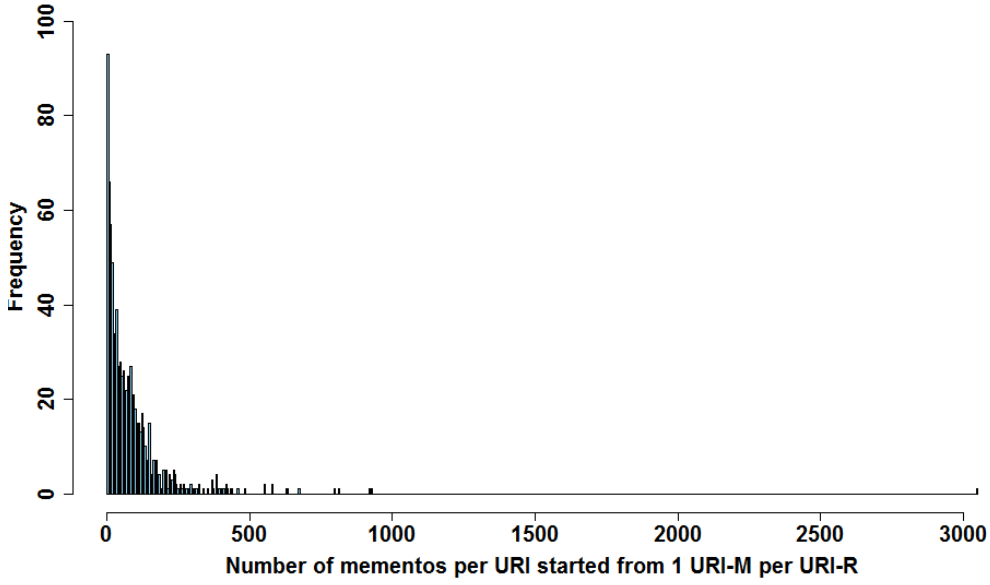
Figures 60-63 are detailed graphs of the distribution of the mementos through the time and is divided into three categories: Internet Archive, search engine caches, and the other archives. The x-axis presents the timeline (1996-2013), and the y-axis presents the URIs ordered by the date of first observation. A dot means that this URI has a memento at this time, and the color shows the source of this observation. Figures show that most of the mementos before 2008 are provided by the Internet Archive. Search engine caches provide very recent copies of the URI. In 2010, the Internet Archive had a quarantine period from 6 months to 1 year [4]; in 2013, the mementos became available on the Internet Archive within hours of the crawling time [237].

8.1.3 ANALYSIS

When we began the research to answer the question "how much of the Web is archived?" we did not anticipate a quick, easy answer. Suspecting that different URI sample sources would yield different results, we chose sources we believed would be independent from each other and provide enough insight to allow estimation of a reasonable answer. Instead we have found that the URI source is a significant driver of its archival status. Table 23 lists the archive percentage for the four samples in 2010 and 2013. In 2010, we divide the results to include and exclude the search engine caches in order to compare with 2013 results. The archive percentage ranged from 16% to 79% in 2010 which increased to 33% to 95% in 2013. The archive percentage is related to the URI

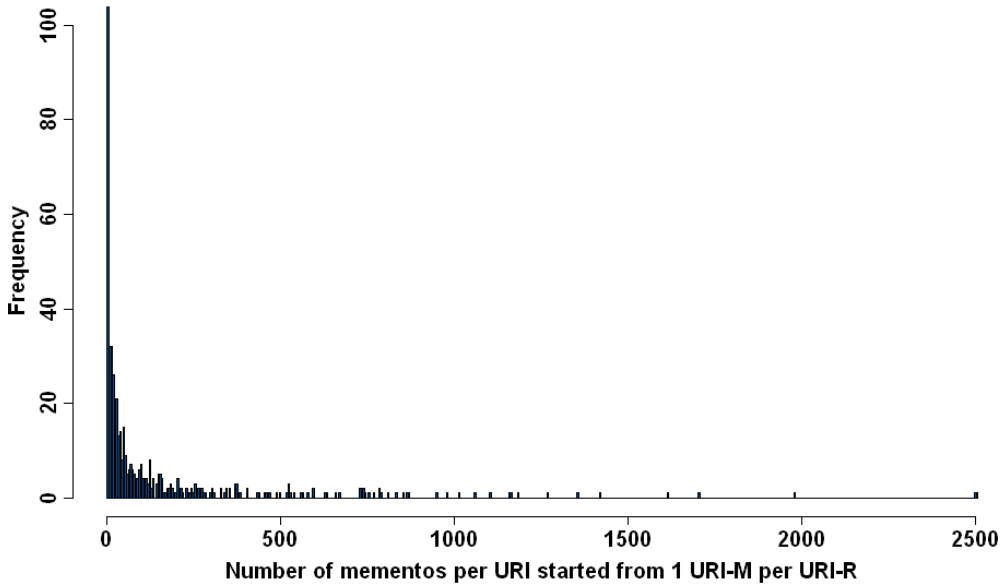


(a) DMOZ - 2010

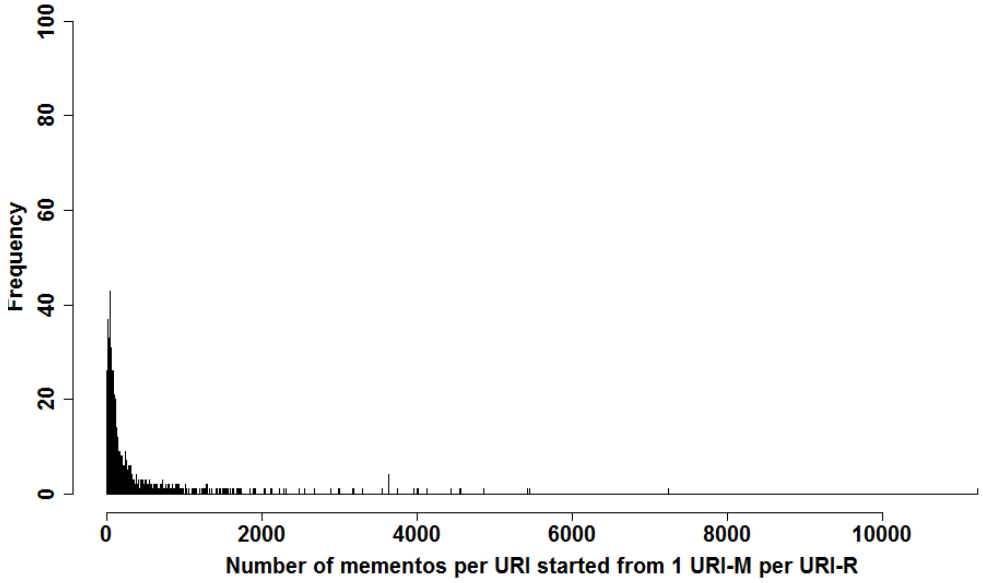


(b) DMOZ - 2013

Fig. 52 Histogram for URI-Ms per $URI-R$ from each sample collection - DMOZ sample.

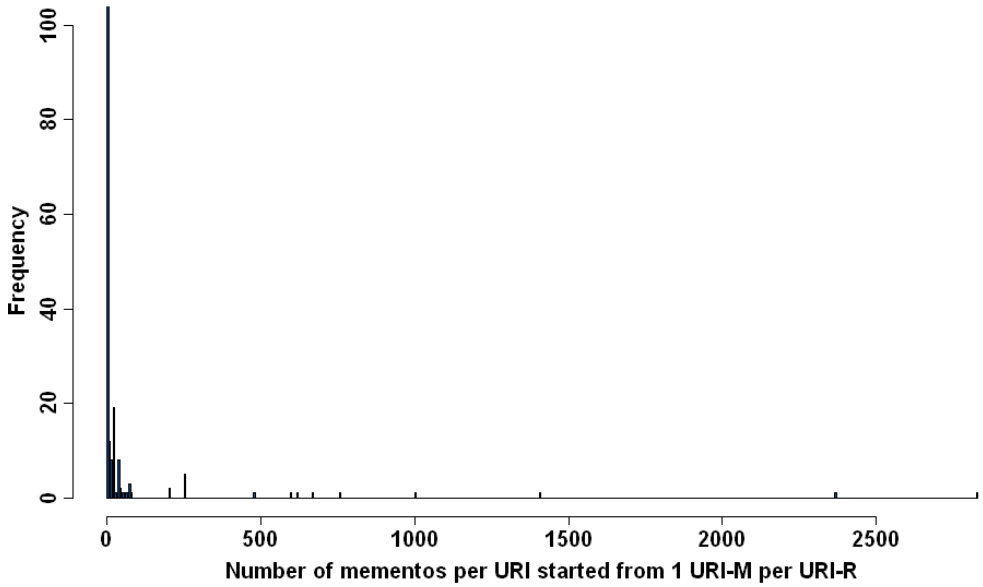


(a) Delicious - 2010

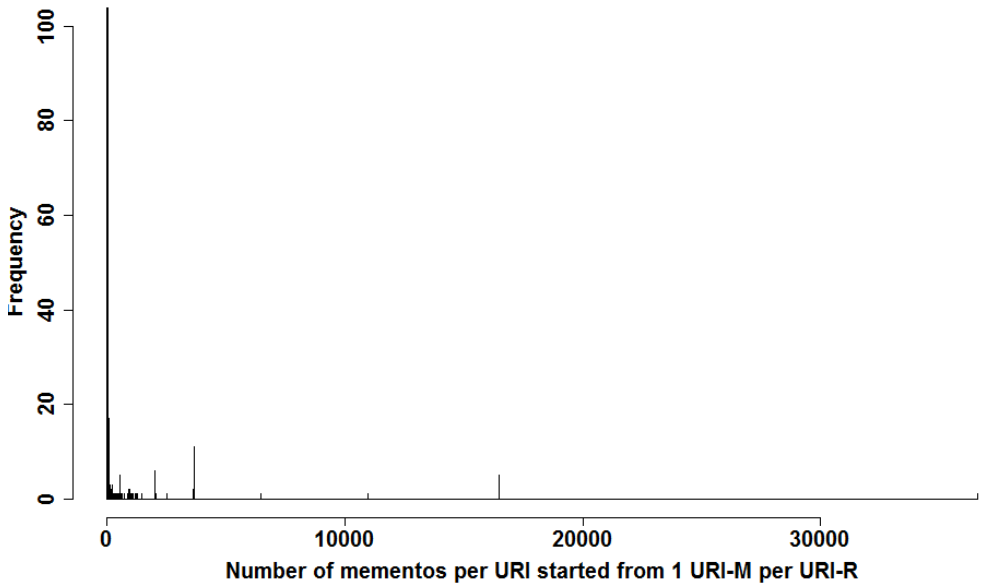


(b) Delicious - 2013

Fig. 53 Histogram for URI-Ms per $URI - R$ from each sample collection - Delicious sample.

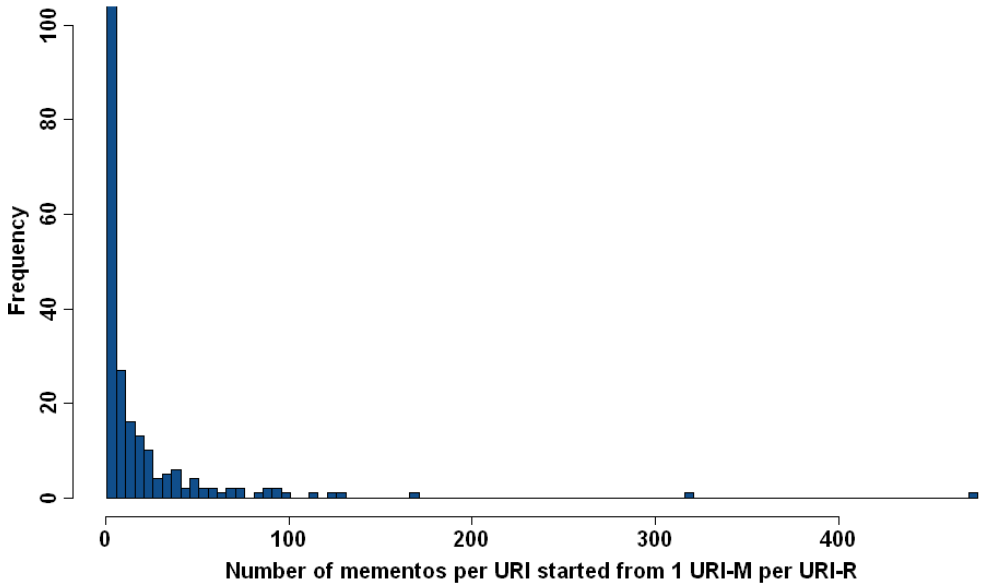


(a) bitly - 2010

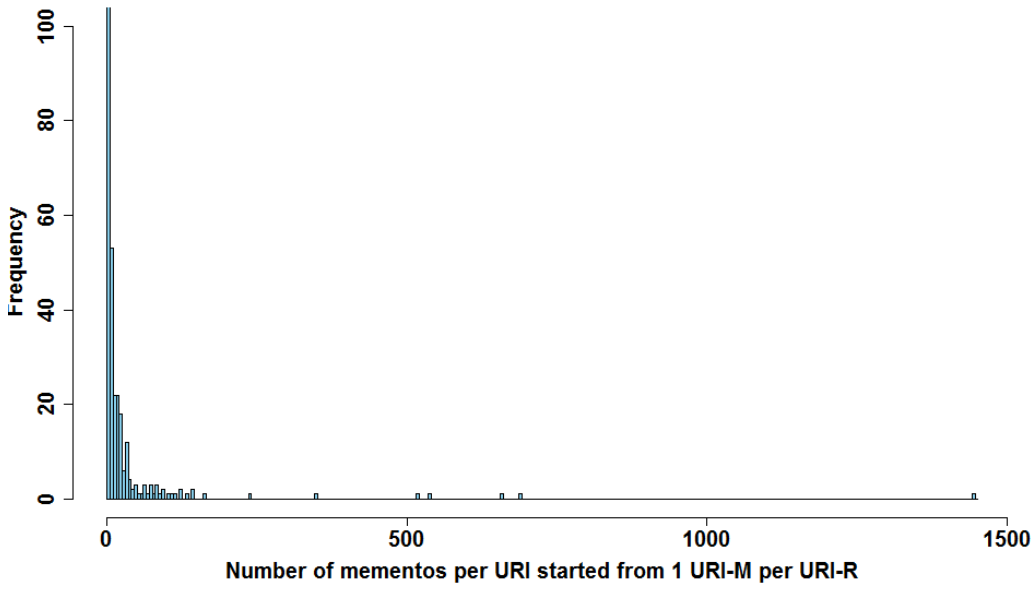


(b) bitly - 2013

Fig. 54 Histogram for URI-Ms per $URI - R$ from each sample collection - bitly sample.



(a) Search Engines - 2010

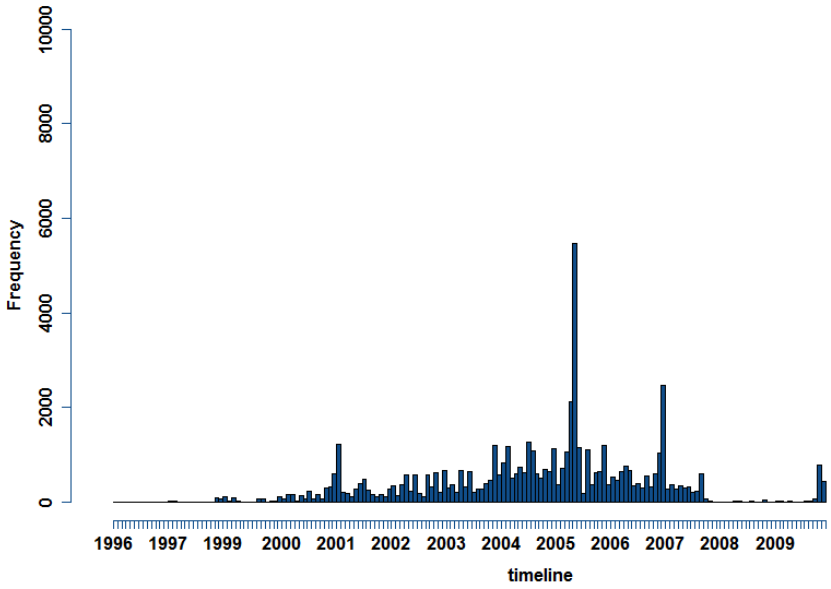


(b) Search Engines - 2013

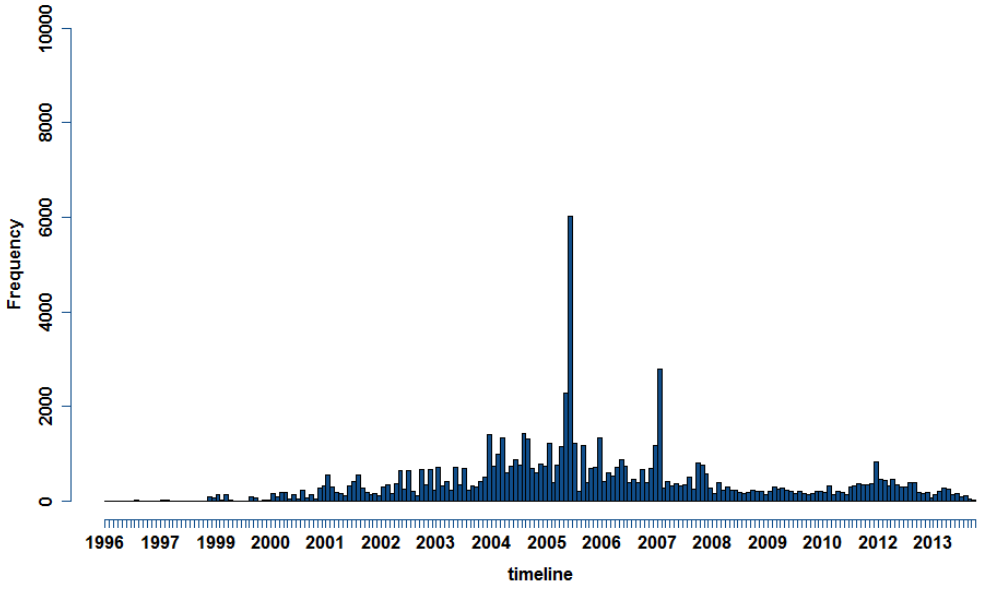
Fig. 55 Histogram for URI-Ms per $URI-R$ from each sample collection - Search Engines sample.

Table 22 Mementos coverage per type. #R denotes the number of URI-Rs and #M denotes the number of URI-Ms.

	DMOZ		**Delicious**		**bitly**		**SE**	
2010	#M	#R	#M	#R	#M	#R	#M	#R
Internet Archive	55293	783	74809	408	8947	70	4067	170
Google	523	523	897	897	253	253	486	486
Bing	427	427	786	786	204	204	515	515
Yahoo	418	418	479	479	87	87	229	229
Diigo	36	36	354	354	61	61	10	10
Archive-It	92	4	500	38	75	13	49	12
UK Nat. Archives	25	8	521	102	531	12	1	1
NARA	5	5	31	19	10	2	4	2
UK Web Archive	8	5	391	38	2892	32	9	3
Web Cite	26	5	594	57	989	58	-	-
ArchiefWeb	-	-	22	3	609	1	-	-
CDLIB	-	-	20	5	-	-	-	-
2013								
Internet Archive	288487	939	76593	904	208715	497	6536	330
Archive-It	7192	320	71	11	33538	79	228	24
Archive.is	6690	397	91	80	509	27	5	4
UK National Archive	3429	201	75	10	43	10	1	1
Web Archive of Catalonia	4855	64	18	11	90	8	2	1
Web Cite	1779	76	23	6	1140	52	-	
Slovenia Web Archive	322	12	-	-	20	1	-	-
Icelandic Web Archive	1474	122	58	8	316	14	27	4
Czech Web Archive	755	129	61	17	219	10	2	1
British Library	278	36	6	3	1509	4	10	4
Singapore Web Archive	104	18	23	3	128	1	210	5
Library of Congress	31	4	6	2	10	2	12	2
Canadian Archive	3	1	2	1	-	-	1	1
Taiwan Web Archive	-	-	-	-	11	1	-	-

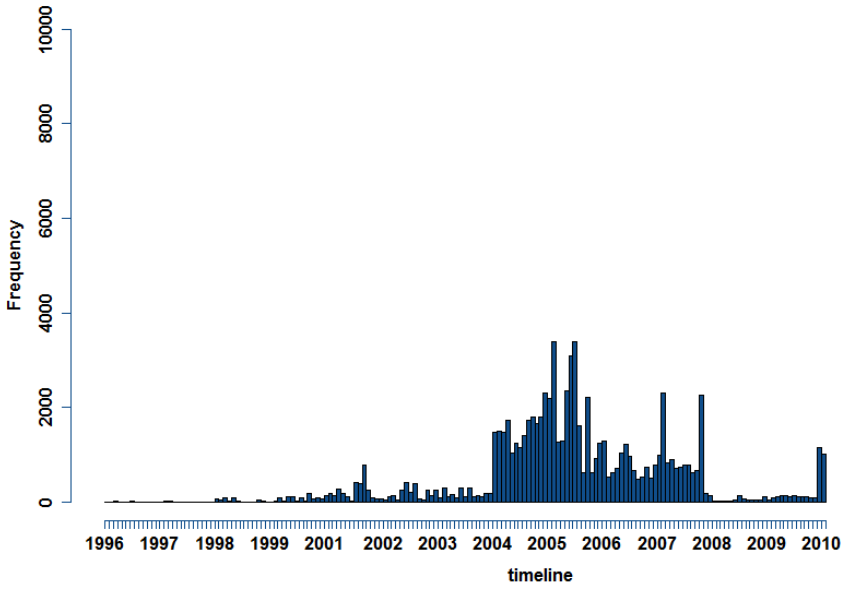


(a) DMOZ - 2010

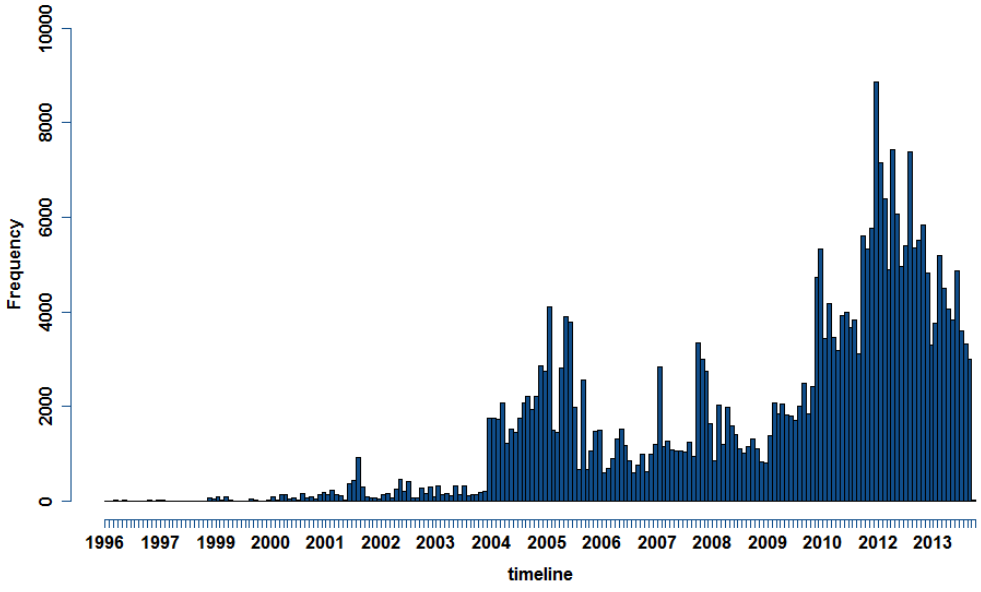


(b) DMOZ - 2013

Fig. 56 Histogram for URI-Ms by month from each sample collection - DMOZ sample.

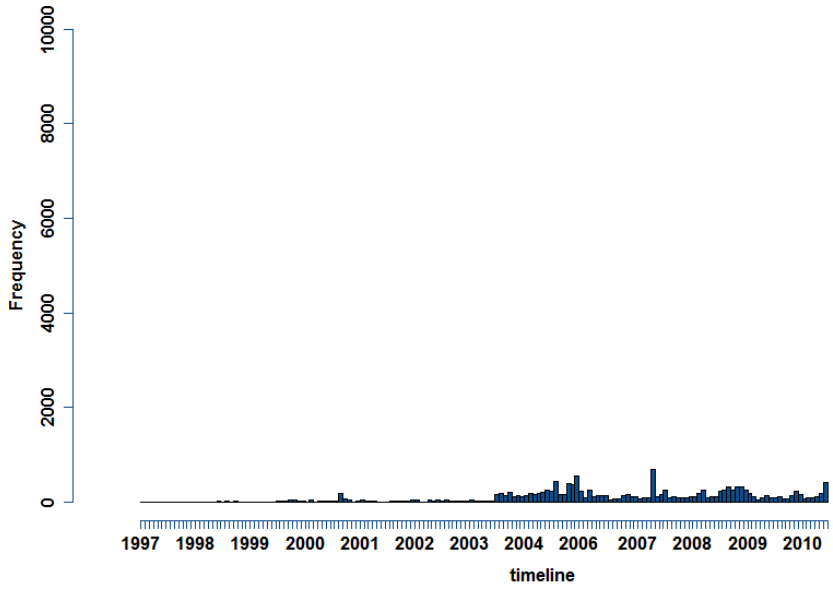


(a) Delicious - 2010

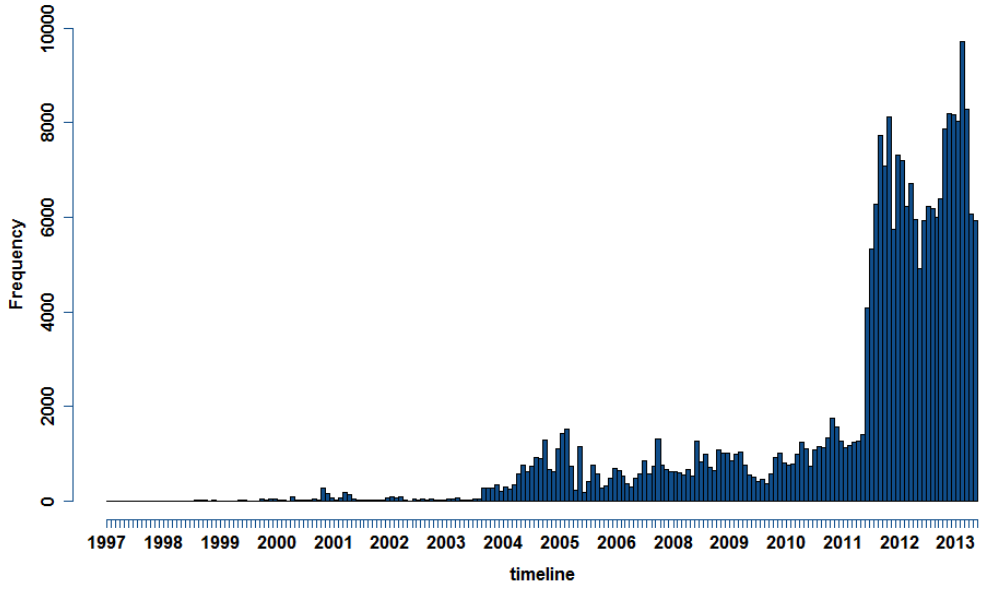


(b) Delicious - 2013

Fig. 57 Histogram for URI-Ms by month from each sample collection - Delicious sample.

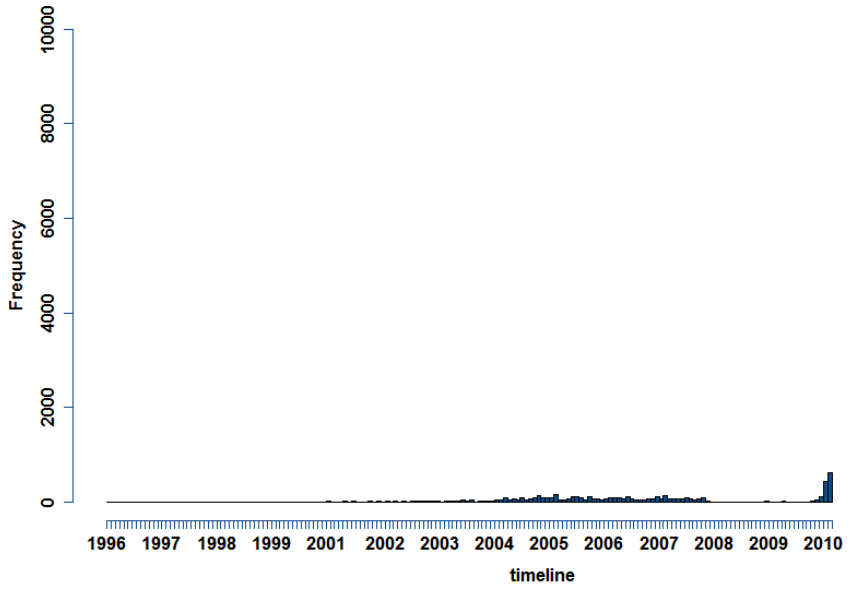


(a) bitly - 2010

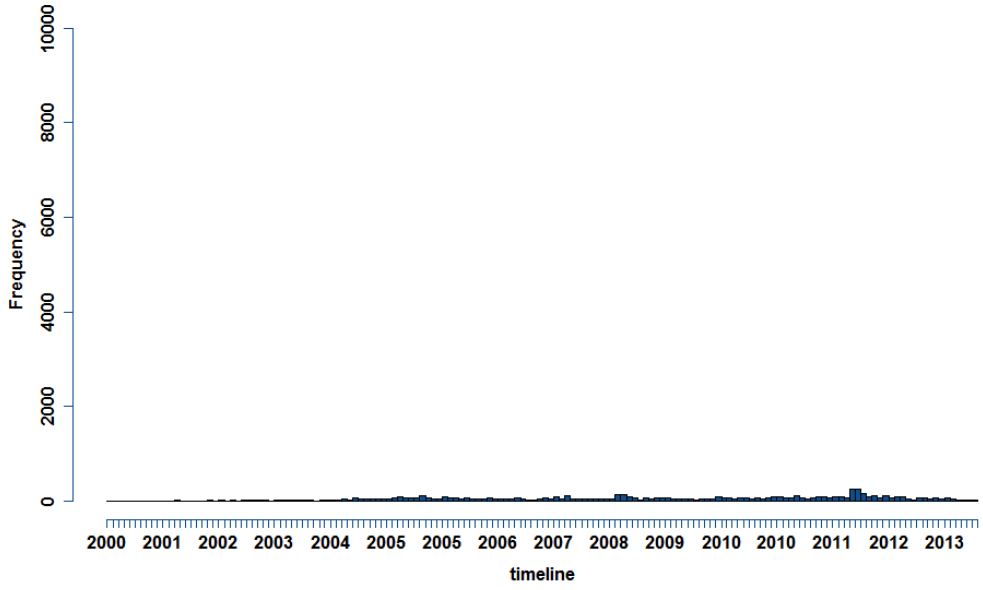


(b) bitly - 2013

Fig. 58 Histogram for URI-Ms by month from each sample collection - bitly sample.

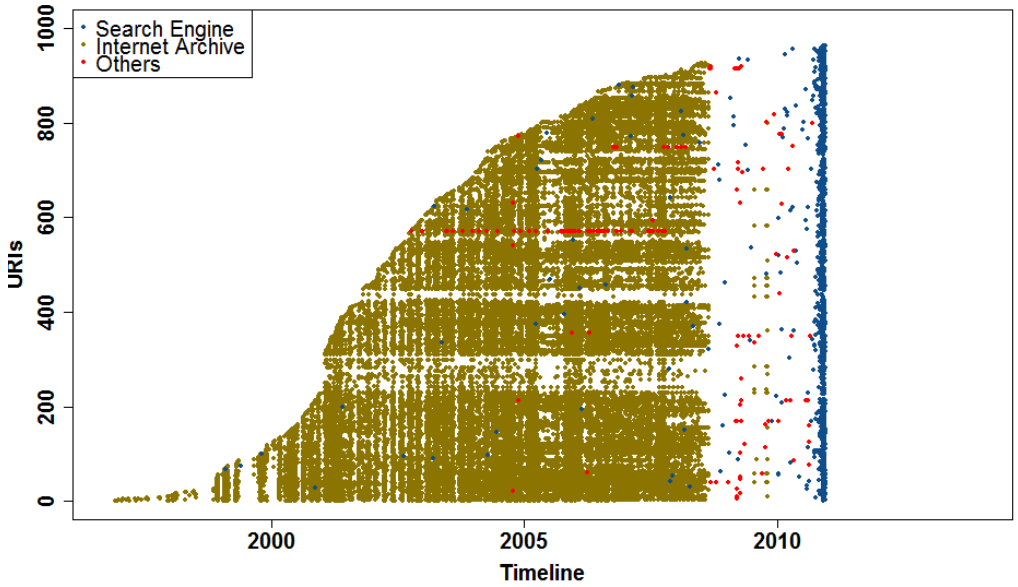


(a) Search Engines - 2010

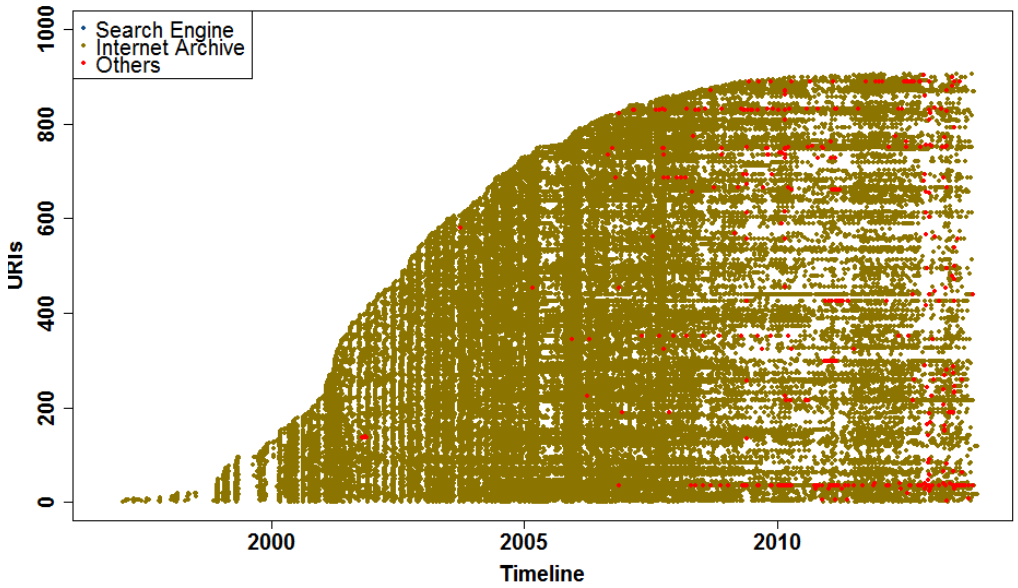


(b) Search Engines - 2013

Fig. 59 Histogram for URI-Ms by month from each sample collection - Search Engines sample.

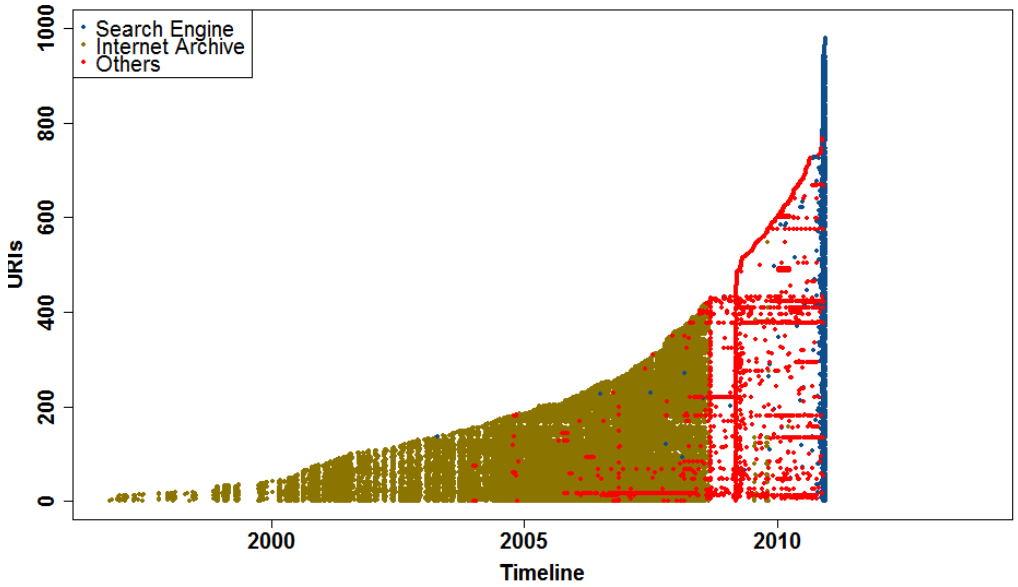


(a) DMOZ - 2010

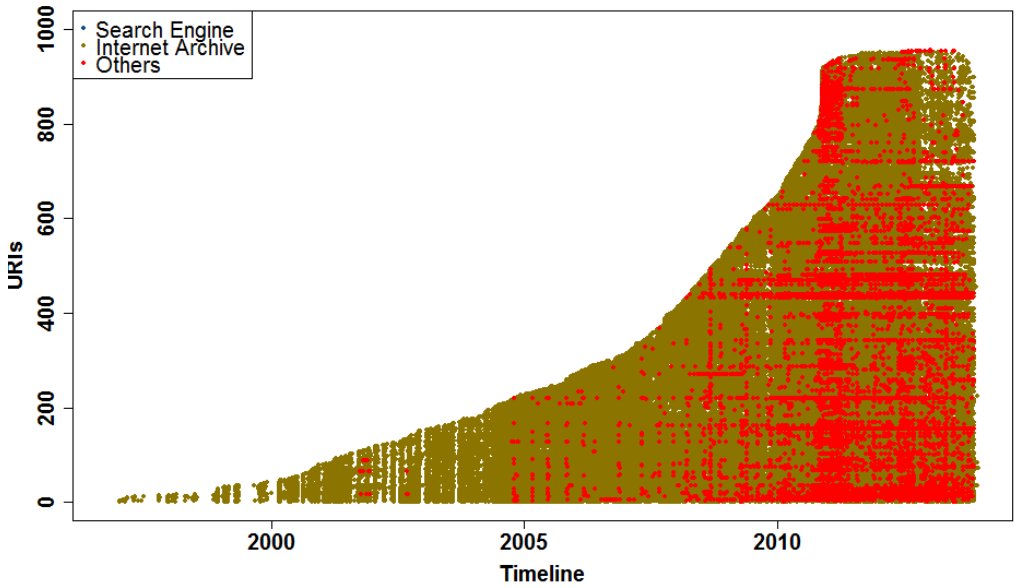


(b) DMOZ - 2013

Fig. 60 Distribution of $URI - M$ for each $URI - R$ from each sample collection, sorted by the first observation date - DMOZ sample.

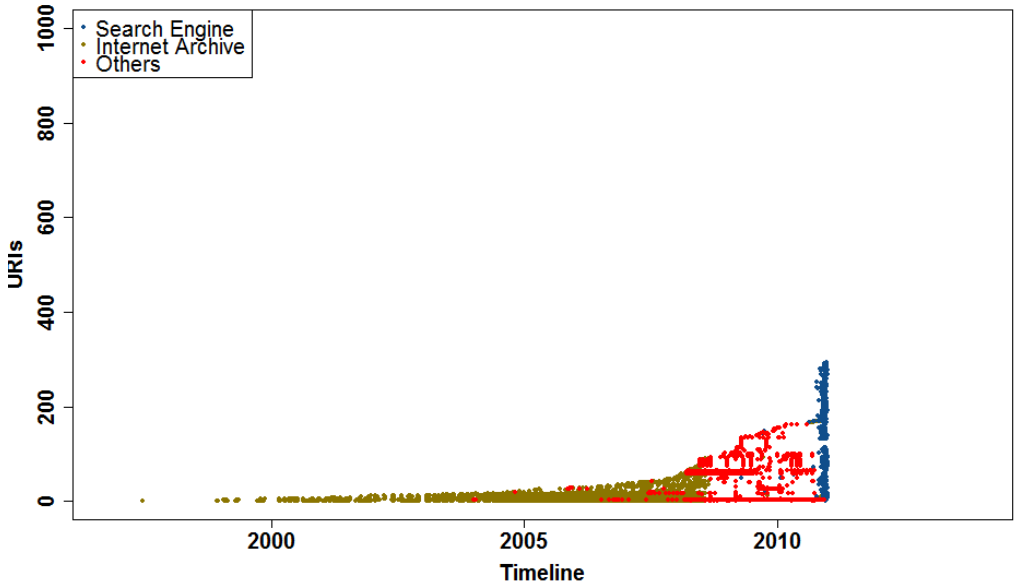


(a) Delicious - 2010

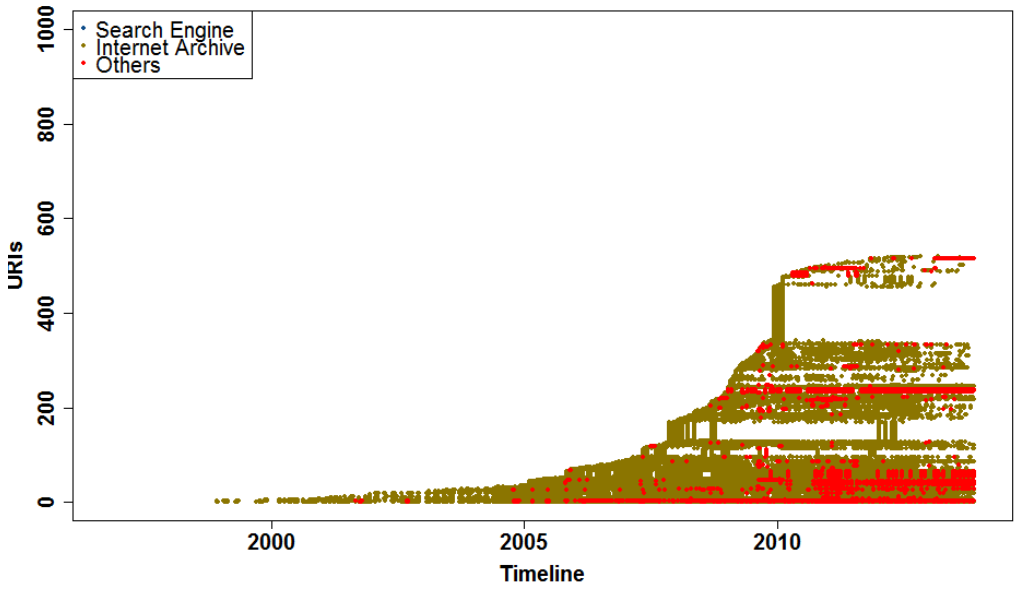


(b) Delicious - 2013

Fig. 61 Distribution of $URI - M$ for each $URI - R$ from each sample collection, sorted by the first observation date - Delicious sample.

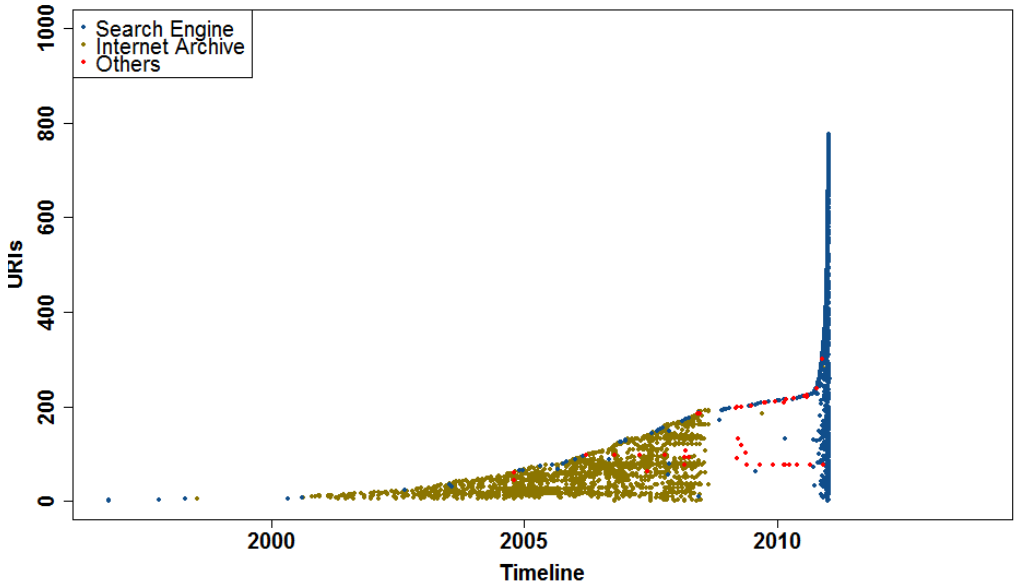


(a) bitly - 2010

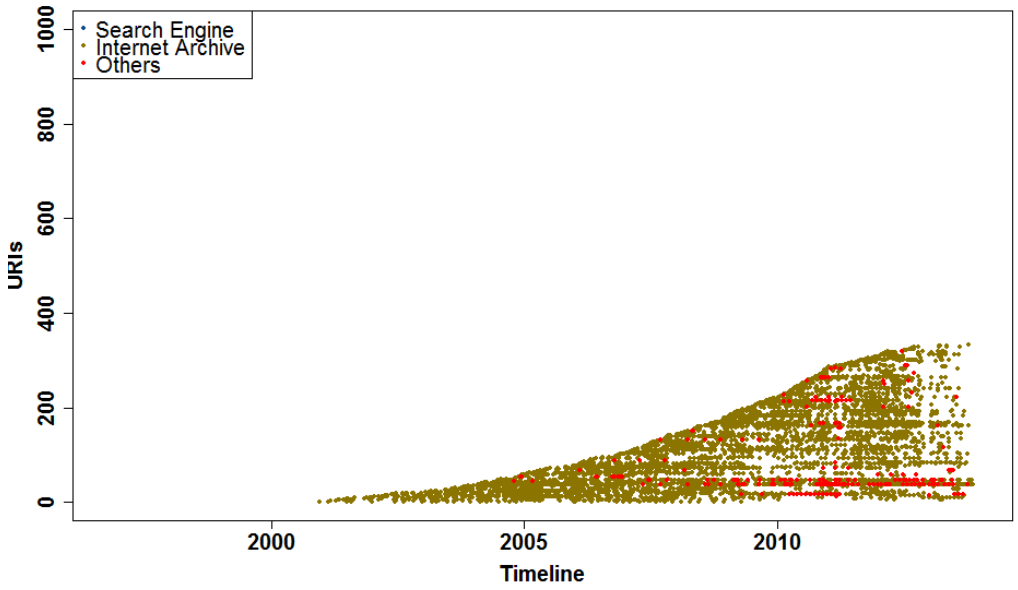


(b) bitly - 2013

Fig. 62 Distribution of $URI - M$ for each $URI - R$ from each sample collection, sorted by the first observation date - bitly sample.



(a) Search Engines - 2010



(b) Search Engines - 2013

Fig. 63 Distribution of $URI - M$ for each $URI - R$ from each sample collection, sorted by the first observation date - Search Engines sample.

Table 23 Archived percentage of each sample.

	2010		2013
Sample	Including SE cache	Excluding SE Cache	General
DMOZ	90%	79%	90%
Delicious	97%	68%	95%
bitly	35%	16%	52%
Search Engine	88%	19%	33%

popularity.

Consider the graphs in Figures 60-63. Clearly, the archival rate (URIs with at least one memento) is much higher for the DMOZ and Delicious samples than for the bitly and search engine samples, which also differ considerably from each other. URIs from DMOZ and Delicious have a very high probability of being archived at least once. On the other hand, URIs from search engine sampling have about 2/3 chance of being archived and bitly URIs just under 1/3. This leads us to consider the reasons for these differences.

Something that DMOZ and Delicious have in common is that a person actively submits each URI to the service. Search engine sample URIs, however, are not submitted in the same way. The process used by search engines is more passive, with URI discovery depending on search engine crawl heuristics. bitly is more of a mystery. Our research did not delve into how bitly URIs are used. But the low archival rate leads us to think that many private, stand-alone, or temporary resources are represented. These substantial archival rate differences have led us to think that the URI popularity receives may be a key driver of archival rate for publicly-accessible URIs.

8.2 THE DISTRIBUTION OF THE ARCHIVED WEB

As discussed in the previous section, the global archived web corpus is distributed between various web archives around the world. In this section, we build Archive Profiles that represent web archive holding characteristics such as age and top-level domains. The profiles are used in query optimization for the Memento Aggregator.

8.2.1 INTRODUCTION

Every archive has its own policy to crawl and preserve the web [249], and these rules control its selection policy to determine the set of Uniform Resource Identifiers (URI) for the web archive to crawl and preserve [197].

However, neither the selection policy nor the crawling log may be publicly available. This means that there is no way to determine what has been planned nor actually archived. This challenges our ability to search for a URI in the archives. For example, the British Library Web Archive is interested in preserving UK websites (domains ending with .uk or websites existing in the UK)[8], so searching it for The Japan Times[9] may not return anything because that URI is not in the BL's

[8] `http://www.bl.uk/aboutus/stratpolprog/coldevpol/index.html`
[9] `http://www.japantimes.co.jp/`

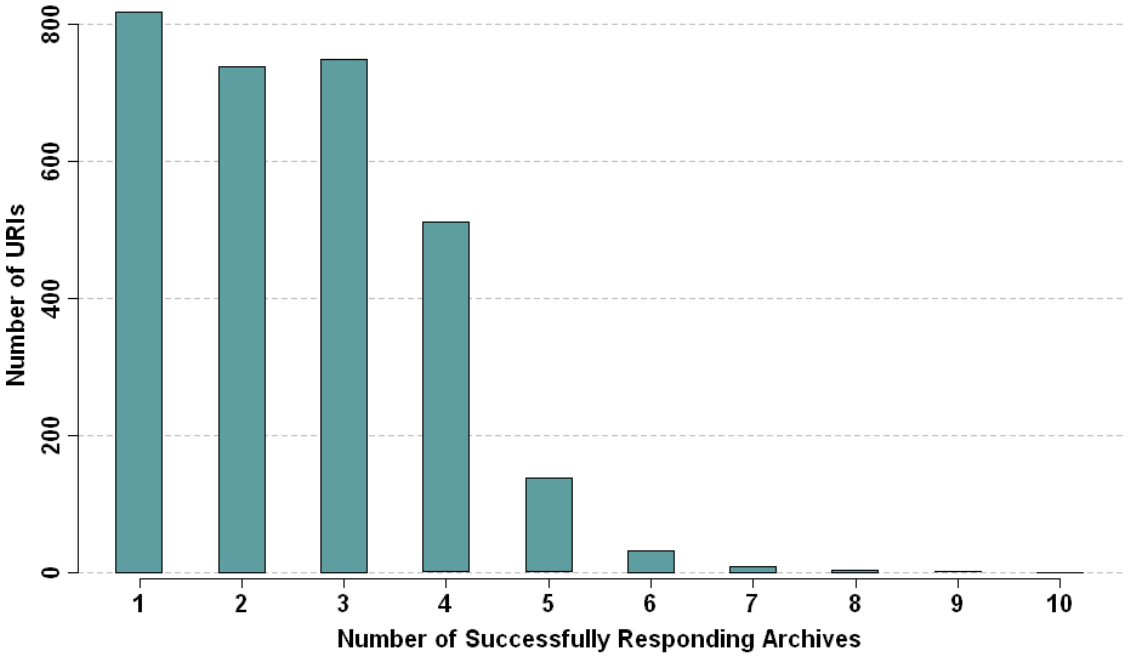


Fig. 64 Histogram for the number of archives holding mementos in the experiment described in [35].

selection policy. Furthermore, although `www.bbc.co.uk` is covered in the BL Web Archive, a request for this URI from the year 2000 should not be sent to the BL because it did not begin archiving until 2007. The Memento Aggregator is motivated by the fact that there is no single web archive that covers the whole Web [35]. Merging the results from different web archives provides better coverage for the archived web.

The research problem is a standard uncooperative distributed information retrieval model [86, 104] where the only interface between the Memento Aggregator and the web archive is the request (usually URI) and the response (TimeMap). In this research, we will discuss the web archive representation by building a Web Archive Profile and use these profiles in ranking the web archives in order to select the top matched archives. Merging the results is based on the Memento Aggregator mechanism to order the results based on the Memento-Datetime (the time the page was archived).

For each web archive, we determine a set of characteristics that distinguishes each web archive from the others and provides an insight about the archive content, e.g., the age of the archived copies and the supported domains for crawling. This profile enables the selection of the archives that may have the requested URI at a specific datetime. The main application for the profile is the Memento Aggregator. The profile will help the Memento Aggregator to minimize the number of requests sent to archives by matching the URI characteristics to the profiles. Also, query routing could be used by the user-agent or the web archive to redirect the request based on the requested URI characteristics to another web archive that may have the URI. The analysis of the profiles may help in determining

Table 24 List of web archives used in the experiment. "x" means the web archive has a full-text search interface.

Archive Name	Abbreviation	FullText search
Internet Archive	IA	
Archive-It	AIT	x
Library of Congress	LOC	
UK Gov. Web Archive	UK	x
UK Web Archive by British Library	BL	x
Singapore Web Archive	SG	
Web Archive of Catalonia	CAT	x
Croatian Web Archive	CR	x
Archive of the Czech Web	CZ	x
National Taiwan University	TW	x
Icelandic Web Archive	IC	
Library & Archives Canada	CAN	
Slovenia Web Archive	SI	
Web Citation	WEB	
Archive.is	AIS	

the missing portions of the web that need more coverage.

In this research, we perform a quantitative study to create profiles for 15 web archives around the world (see Table 24). We select these web archives because they have public, online, and consistent access interfaces. We access the archives through the following interfaces: Memento interface for the compliant web archives (e.g., Internet Archive and UK Web Archive at British Library), API interface if available (e.g., Croatian Web Arcive), or page scraping techniques. There are other web archives but most of them are dark archives (e.g., National Library of France (BnF) [263]) which provide on-site access only [132], so we can not include them in this study. To build these profiles, we use a dataset constructed from URIs for the live web, full-text search of the archives themselves, and access logs of the archives. We evaluate using the profiles in the web archive selection in the Memento Aggregator to optimize query routing across various archives.

8.2.2 WEB ARCHIVE PROFILE

A web archive profile is a set of characteristics that describe the content of the web archive. This description gives a high-level overview about the web archive. This overview summarizes the web archive content and it helps the Memento Aggregator, the user, other archives, or third party services to select the best web archive in case selection between different archives is required. The characteristics include the following:

- **Age** describes the age of the holdings of the web archive. It is defined by the Memento-Datetime of the oldest memento in the archive. It may differ from the web archive starting date. For example, the Portuguese Web Archive project started in 2007 but they included preserved materials that were captured before 1995[10] [42].

[10] `http://sobre.arquivo.pt/how-to-participate/supplying-historical-portuguese-web-contents`

In this rule, we record the Memento-Datetime for the archive's earliest memento and optionally its most recent one. The values for this rule are datetimes expressed according to RFC 1123 [77].

Example, `Age:  Wed, 02 Oct 2002 13:00:00 GMT`

- **Top-level domain (TLD)** describes the top-level domains of the captured URIs in the web archive. Some web archives have a special focus that will consider specific domains only. For example, Library and Archives Canada has focused on the `.gc.ca` TLD. This rule should include a list of the supported domains separated by commas. For each TLD, we include the weight of this TLD in the archive. It also could be * to mean all domains.

  Example, `TLD: gc.ca:0.9`

- **Language** describes the supported languages by the web archive. It varies depending on the motivation of the web archive creation. The Internet Archive has a wide range of languages, while the Icelandic web archive focuses on content in the Icelandic language. The values for this rule must follow RFC 5646 [228] standards for the language. For each language, we include the weight of this language in the archive.

  Example, `Language:  en:0.4,fr:0.2`

- **Growth rate** describes the growth of the web archive corpus in the number of URI-Rs and URI-Ms through time. Growth rate is defined by a list of key-value pairs, the key indicates the month in the form of YYYYMM. The value has two fields separated by comma; the first field is the number of new added URI-Rs in this month and the second field is the number of added URI-Ms in this month.

  Example,
  `GrowthRate:"199707":[4,4],"200202":[1,1]`

There are some challenges in building these profiles. Web archives do not publish statistical information about their holdings. Rarely do they publish general information about their collection policy or the current size of the archives. The web archives environment is an uncooperative distributed model; the only available interface is the query/response model. In order to build the web archive profile, we prepare URIs from various samples. We query each web archive with these URIs and record the retrieved TimeMap. We analyze the retrieved TimeMap and extract a summary about the web archive holdings and formalize it into the "Web Archive Profile" format.

8.2.3 URI DATASET SAMPLES

Table 25 Total number of URIs per sample.

Sample	Size in URIs
1. DMOZ Random	10,000
2. DMOZ Languages	4,000
	40 Lang. x 100 URIs
3. DMOZ TLD	16,000
	80 TLDs x 200 URIs
4. Top 1-Gram	21,285
AIT	4,093
BL	3,182
CAN	1,062
CR	1,223
CZ	3,360
CAT	4,241
TW	362
UK	3,762
5. Top query languages	54,091
AIT	14,143
BL	6,474
CAN	1,348
CR	1,744
CZ	6,567
CAT	10,093
TW	1,041
UK	12,681
6. IA Wayback Machine	10,000
7. Memento Aggregator	1,000
Total	116,376

We prepared various URI sample sets to profile the web archives between August and September 2013. We sampled URIs from three sources: live web, archive holdings, and archive access logs.

Open Directory (DMOZ)[11] is used as a source of URIs that are (or were) available on the live web. Recording web archives' full-text search responses represent what the web archives have already acquired. Finally, sampling from log files of the Internet Archive and Memento Aggregator represents what the users are looking for in the archives, regardless of whether or not the archives hold it. In all the samples, we used the hostname to create a top-level URI. For example, `http://example.org/a/b.html` will be `example.org`. Each sample has a unique set of hostnames, however the different samples may have an overlap of hostnames. Table 25 lists the number of URIs in each sample. Table 26 lists the overlap between each pair of samples.

Sampling from the Web

We uses the DMOZ RDF file of September 2013. We create three samples from DMOZ data:

1. **DMOZ Random**: We randomly sample 10,000 URIs from the total directory of more than 5M URIs. This sample is used to calculate the general coverage of each web archive.

[11] `http://www.dmoz.org`

Table 26 The overlap between the URI samples.

	DMOZ Lang	DMOZ TLD	IA logs	Aggregator logs	Top 1 Gram	Top Query
DMOZ Random	13	56	0	23	191	286
DMOZ Lang	-	160	1	0	10	22
DMOZ TLD	-	-	0	1	80	160
IA logs	-	-	-	1	2	5
Aggregator logs	-	-	-	-	65	112
Top 1-Gram	-	-	-	-	-	8889

2. **DMOZ Controlled (TLD)**: We classify the DMOZ directory's URIs by the TLD. For each TLD, we randomly select 200 of the available hostnames. We limit the study to a specific set of TLDs that are distributed around the world. The total number of URIs in this sample is 16,000. Table 27 lists the domain under experiment with the recorded country.

3. **DMOZ Controlled (Language)**: DMOZ categorizes a list of URIs per language. We extract these URIs and select randomly 100 URIs from each language. We select 40 languages from DMOZ directory. The total size of this sample is 4000 URIs. This sample results are analyzed to determine the content language for the web archive holding.

Sampling from Web Archives

Most of the web archives provide full-text search in addition to URI-lookup or collection browsing. We use the full-text search to discover the deep content of the web archives by submitting various queries and recording the responses.

This sample aims to calculate the overlap between the different archives and avoid biasing for the archives that use DMOZ as a URI source (such as the Internet Archive). We use the web archives that support full-text search (see Table 24)[12]. In order to reach a representative sample, we use two sets of queries:

4. **Top 1-Gram**: The first set of query terms is extracted from Bing Top 100k words as they appeared on April 2010[13]. We randomly sample 1000 terms where most of them are in English. We query each web archive and record the top 10 results from each response and extract the hostname only. The total number of URIs in this sample is 21,285. Table 25 shows the sampled URIs from each archive.

5. **Top Query Languages**: The second set of query terms is taken from Yahoo! Search query logs for nine languages[14]. This dataset has the 1000 most frequent web search queries issued

[12] UK Gov. Web Archive has a problem in searching with unicode characters
[13] `http://web-ngram.research.microsoft.com/info/BingBodyApr10_Top100KWords.zip`
[14] `http://webscope.sandbox.yahoo.com/catalog.php?datatype=l`

Table 27 List of Top-Level domains in DMOZ TLD sample.

TLD	Country	TLD	Country	TLD	Country
ae	UAE	fr	France	org	*Organization*
am	Armenia	gov	*Government*	pe	Peru
ar	Argentina	gr	Greece	ph	Philippines
at	Austria	hk	Hong Kong	pk	Pakistan
au	Australia	hr	Croatia	pl	Poland
az	Azerbaijan	hu	Hungary	pt	Portugal
ba	Bosnia	id	Indonesia	ro	Romania
be	Belgium	ie	Ireland	rs	Serbia
bg	Bulgaria	in	India	ru	Russia
br	Brazil	info	*info*	sa	Saudi Arabia
ca	Canada	ir	Iran	se	Sweden
cat	Catalan	is	Iceland	sg	Singapore
ch	Switzerland	it	Italy	si	Slovenia
cl	Chile	jp	Japan	sk	Slovakia
cn	China	kr	Korea	th	Thailand
co	Colombia	kz	Kazakhstan	tn	Tunisia
com	*Commercial*	lb	Lebanon	to	Tonga
cu	Cuba	lt	Lithuania	tr	Turkey
cy	Cyprus	ma	Morocco	tw	Taiwan
cz	Czech Republic	mt	Malta	ua	Ukraine
de	Germany	mx	Mexico	uk	UK
dk	Denmark	my	Malaysia	us	USA
edu	*Education*	na	Namibia	uy	Uruguay
ee	Estonia	net	*Net*	uz	Uzbekistan
eg	Egypt	nl	Netherlands	ve	Venezuela
es	Spain	no	Norway	za	South Africa
fi	Finland	nz	New Zealand		

to Yahoo Search in nine different languages. The languages are: Chinese, English, French, German, Italian, Japanese, Korean, Portuguese, and Spanish. As the query terms are not limited to the search engine languages, they may have other languages, especially English (e.g., Apple was one of the top query terms in Yahoo Japan `http://www.yahoo.co.jp/`). We manually filter each file to include the designated language only and exclude common terms (e.g., Obama, Facebook). We query each web archive and record the top 10 results from each response and extract the hostname only. The total number of URIs in this sample is 54,091. Table 28 shows the total number of unique hosts returned by querying each query set from the archive. The total column has the total number of unique hosts that are retrieved by each archive. The total column provides an indication of the size of the web archive. The languages codes appear as ISO 639-3 [6].

Sampling from Users' Requests

The third sample comes from users' requests to the past web as recorded by two log files. We sample from the logs disregarding the availability of the URI in the web archives.

Table 28 Total number of unique hostnames returned from the top query languages terms.

	Top Query Languages search									
	chi	eng	fre	ger	ita	jpn	kor	por	spa	Total
AIT	26	2397	3387	3790	3276	120	2	3071	3575	14143
BL	163	2365	2367	2254	2087	225	132	1944	2063	6474
CAN	31	804	802	602	576	54	99	651	587	1348
CR	56	768	820	784	763	81	19	675	675	1744
CZ	363	1892	1747	1855	1573	577	117	1443	1453	6567
CAT	28	2951	2870	2702	2472	209	129	2403	2678	10093
TW	370	187	184	178	160	107	9	208	123	1041
UK	0	3096	2832	2774	2836	0	0	2682	2900	12681

6. **IA Wayback Machine log files**: IA Wayback Machine (WM) is the access interface for the Internet Archive. WM receives more than 90M+ hits per day [37]. We select log files for one week from Feb 22, 2012 to Feb 26, 2012. We use only the requests to mementos or TimeMaps. For each memento or TimeMap, we extract the URI of the original resource. We then sample 10,000 hostnames randomly from this list.

7. **Memento Aggregator logs**: We sample 1,000 unique hosts from the LANL Memento Aggregator logs between 2011 to 2013.

8.2.4 EXPERIMENT AND RESULTS

The experiment aims to discover the representation of the web archive and build its designated profile. The experiment uses query based technique by analyzing the request (as URI) and the response (as TimeMap) for each URI at each web archive. Generally, for each hostname in the sample set (e.g., example.org), we converted it into a URI (i.e., http://example.org). In October 2013, we used the Memento proxies to retrieve the TimeMap for each URI on each one of the 15 archives. We queried each archive individually instead of using the Memento Aggregator. We recorded each discovered memento with its URI-M and Memento-Datetime. Recall that the components of a web archive profile include TLD coverage, language distribution, and growth rate. In this section, we report the results of developing profiles for the web archives listed in Table 24.

General Coverage

We calculate the coverage using two measurements.

1. $Coverage_{URI}(R)@WA = 1$ if the URI R has at least one memento at web archive WA (Equation 15). It is extended to $Coverage_{URI}(Sample)@WA$ as the percentage of the URIs of the sample that are available in a web archive (Equation 17).

2. $Coverage_{MEM}(R)@WA$ is the total number of mementos discovered in a web archive WA; if the URI appears in the web archive, we count it with the number of mementos, otherwise with 0 (Equation 16). It is extended to $Coverage_{MEM}(Sample)@WA$ which is the summation of the discovered mementos for this sample at web archive WA (Equation 18).

- $M(R)@WA$: is a memento M for original resource R in web archive WA.
- $TM(R)@WA = \{M(R)_{t1}@WA, M(R)_{t2}@WA, ...M(R)_{tn}@WA\}$.
- $|TM(R)@WA|$ is the number mementos $M(R)_{ti}$ in the TimeMap TM at web archive WA.
- $Sample = \{R_1, R_2, ..., R_N\}$ is a set of original resources R_i.

$$Coverage_{URI}(R)@WA = \begin{cases} 1 & |TM(R)@WA| >= 1 \\ 0 & \text{Otherwise} \end{cases} \tag{15}$$

$$Coverage_{MEM}(R)@WA = \begin{cases} n & n = |TM(R)@WA| \\ 0 & \text{Otherwise} \end{cases} \tag{16}$$

$$Coverage_{URI}(Sample)@WA = \frac{\sum_{i=1}^{N} Coverage_{URI}(R_i)@WA}{N} * 100\% \tag{17}$$

$$Coverage_{MEM}(Sample)@WA = \sum_{i=1}^{N} Coverage_{MEM}(R_i)@WA \tag{18}$$

$$CCoverage_{URI}(WA_1)@WA_2 = Coverage_{URI}(\text{Sample extracted from } WA_1)@WA_2 \tag{19}$$

$$NCoverage_{URI}(Sample)@WA = \frac{Coverage_{URI}(Sample)@WA}{\sum_{i=1}^{N} Coverage_{URI}(Sample)@WA_i} \tag{20}$$

$$NCoverage_{MEM}(Sample)@WA = \frac{Coverage_{MEM}(Sample)@WA}{\sum_{i=1}^{N} Coverage_{MEM}(Sample)@WA_i} \tag{21}$$

Fig. 65 Web Archive Coverage Teminology.

Table 29 The URI coverage percentage across the archives for fulltext search results top 1-Gram sample.

	Source Archive							
	AIT	BL	CAN	CR	CZ	CAT	TW	UK
ait	**90.0**	36.1	50.2	10.4	14.3	26.3	21.8	43.9
bl	8.5	**82.6**	0.8	2.0	3.8	4.0	3.3	17.4
can	1.0	0.3	**89.6**	0.1	0.4	0.1	0.8	0.0 0.5
cr	0.1	0.1	0.0	**80.9**	0.1	0.0	0.0	0.1
cz	22.1	14.4	8.9	12.2	**94.5**	11.8	10.5	16.2
cat	16.7	9.2	1.8	3.1	5.0	**79.1**	5.0	10.6
tw	1.1	0.8	0.1	0.6	0.5	0.3	**100.0**	0.8
uk	30.4	52.2	12.2	5.2	9.6	15.4	9.4	**84.8**
ia	95.4	96.6	87.9	98.8	97.0	86.7	97.0	94.2
ic	22.5	13.7	3.3	7.4	8.8	10.7	8.0	19.3
loc	4.2	1.2	0.8	0.1	1.8	1.6	1.9	2.0
sg	3.2	2.1	0.2	1.4	1.5	1.3	1.1	2.8
si	4.3	2.3	0.2	3.7	2.1	1.5	1.4	2.3
ais	40.1	41.1	11.4	24.9	23.5	25.1	35.9	35.4
web	8.6	6.3	2.2	2	3.4	2.8	4.1	6.4

For example, if we have three web archives WA_1, WA_2, and WA_3; and URI R has three mementos in WA_1, two mementos in WA_2 and zero memento in WA_3, then we can compute the coverage measurements as the following:

- $Coverage_{URI}(R)$: $@WA_1 = 1$, $@WA_2 = 1$, and $@WA_3 = 0$.
- $Coverage_{MEM}(R)$: $@WA_1 = 3$, $@WA_2 = 2$, and $@WA_3 = 0$.

In this section, we calculate the general coverage based on DMOZ random, IA logs, and Aggregator logs samples. The rest of samples will be used in other analysis purposes. Figure 66 shows the general coverage for each sample through the archives. We draw the Internet Archive in a separate plot with a different scale due to the large difference between IA and the rest of the archives. The results show that IA has the best coverage for all samples, $Coverage_{URI}@IA$ ranges between 77% and 98%. IA covers DMOZ samples with more than 95% because DMOZ is used as a URI source for IA web crawler. However, compared to the Internet Archive, the rest of archives have medium to low $Coverage_{URI}$. AIS shows good $Coverage_{URI}$ ranges between 4% and 21% but the size of the corpus is small as shown with $Coverage_{MEM}$.

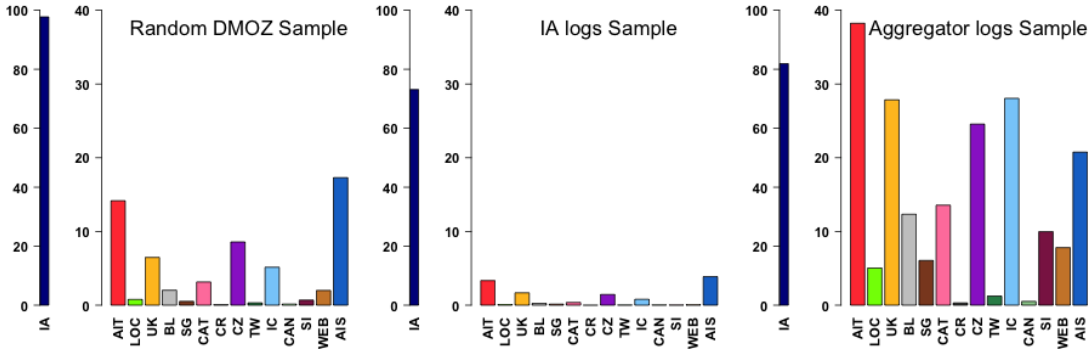
Cross URI Coverage

We use the two web archive samples (Top 1-Gram and Top Query languages) to compute the intersection between the web archives. $CCoverage_{URI}(WA_1)@WA_2$ is the percentage of the URIs that have been extracted from WA_1 and has at least one memento at WA_2 (Equation 19). These sample URIs are extracted directly from the full-text web archives index, then for each URI in the two samples, we query the 15 web archives to discover whether they held mementos for them.

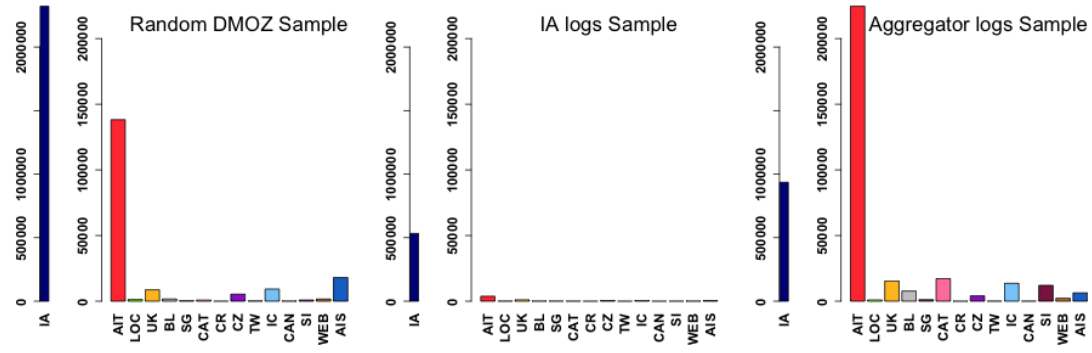
Tables 29 and 30 show the details of the $CCoverage_{URI}$ for both the Top 1-Gram and Top Query

Table 30 The URI coverage percentage across the archives for fulltext search results top query languages sample.

	Source Archive							
	AIT	BL	CAN	CR	CZ	CAT	TW	UK
ait	**85.0**	36.8	50.1	14.2	14.7	22.5	19.5	40.5
bl	4.2	**74.4**	1.3	3.4	3.3	3.4	2.6	8.0
can	0.5	0.3	**88.7**	0.1	0.2	0.1	0.8	0.1
cr	0.0	0.0	0.1	**74.7**	0.1	0.0	0.1	0.0
cz	14.2	17.4	9.0	13.0	**90.3**	10.5	11.1	14.6
cat	9.9	12.0	2.2	4.1	4.5	**77.8**	3.6	9.5
tw	0.5	0.7	0.1	0.6	0.5	0.2	**99.9**	0.3
uk	22.3	44.5	12.4	8.6	9.8	13.9	9.4	**75.9**
ia	86.0	95.3	88.1	97.9	96.4	85.6	94.4	92.2
ic	16.8	16.5	3.9	8.8	9.9	11.0	5.8	21.1
loc	2.1	1.7	0.9	0.6	1.2	1.4	1.2	1.6
sg	2.0	2.9	0.4	1.9	1.7	1.3	1.5	2.0
si	2.3	3.0	0.4	4.0	1.7	1.2	1.3	1.8
ais	29.6	39.6	11.2	26.4	20.1	22.6	42.0	32.3
web	4.4	5.8	1.8	3.0	2.8	2.5	3.7	3.9



(a) General coverage - $Coverage_{URI}$.



(b) General coverage - $Coverage_{MEM}$.

Fig. 66 The general coverage for three samples (Internet Archive is separated with a different scale on the left-hand side).

Languages samples respectively. The tables list the URIs' source archives (as shown in Table 28) as rows and the queried archive as columns. We can conclude the following from the tables:

1. Internet Archive has $CCoverage_{URI}(WA_x)@IA = 93\%$ on average of the holdings of the other archives which means IA is effectively a superset for all the archives.

2. Archive.is shows good $CCoverage_{URI}(WA_x)@AIS$. However, the number of mementos is much lower compared to the rest of the archives.

3. The overlap between the archives and each other is low, which means they are covering different portions of the web. The highest overlap was between BL and UK because both are focusing on UK domains. i.e., $CCoverage_{URI}(BL)@UK = 49.5\%$

4. The web archives may have inconsistent interfaces between full-text search and URI-lookup, as the overlap between the archive and itself is less than 100% (highlighted in bold in the tables). For example, $CCoverage_{URI}(BL)@BL$ ranges from 74% to 83%. It means querying BL with URIs that were extracted from the BL full-text search may not return successfully in all cases. One reason may be removing the URI path from the extracted URI. If the full-text search returns (i.e., www.example.com/a/b.html), we will use (i.e., www.example.com) in the sample set. For the collection based archive, the curator may be interested in a specific URI and not the hostname itself. For example, you can extract `http://www.icsc.org.uk/index.html` from BL by searching for the term "`Consumer Sciences`", but there are no mementos for `http://www.icsc.org.uk/`.

   ```
   curl -I "http://www.webarchive.org.uk/
    wayback/archive/*/http://www.icsc.org.uk/
    index.html"
   HTTP/1.1 200 OK
   ....

   curl -I "http://www.webarchive.org.uk/
    wayback/archive/*/http://www.icsc.org.uk"
   HTTP/1.1 404 Not Found
   ...
   ```

   This example is a configuration problem for URI canonicalization (both URIs are typically considered equivalent). Another example with a deep link clarifies the problem. You can extract `http://www.drivespringfield.com/used-cars/view/510039/peugeot-207-16-16v-gt-2dr` from UK by searching for the term "`Springfield used cars`", but there are no mementos for `http://www.drivespringfield.com`.

   ```
   curl -I "http://webarchive.nationalarchives.
       gov.uk/*/http://www.drivespringfield.
       com/used-cars/view/510039/peugeot-
   ```

```
    207-16-16v-gt-2dr"
HTTP/1.1 200 OK
....

curl -I "http://webarchive.nationalarchives.
    gov.uk/*/http://www.drivespringfield.com"
HTTP/1.1 404 Not Found
...
```

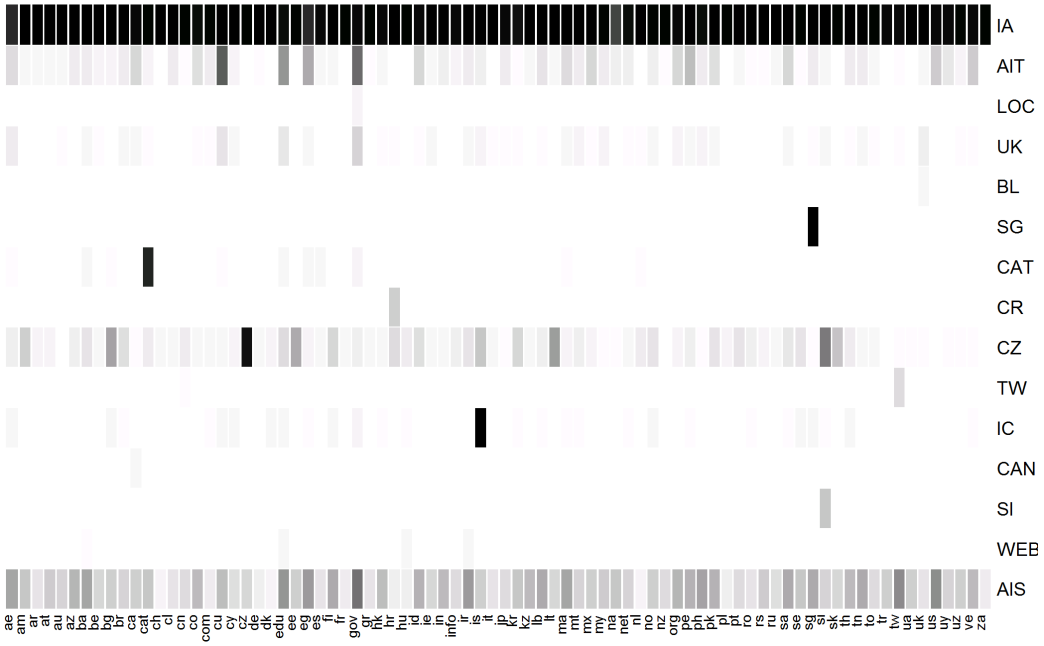
In this case we truncate the deep link to `www.drivespringfield.com` because the root level URI is more indicative of the policies of all the archives with respect to an entire domain (and the associated TLD) than arbitrary deep links to specific URIs.

Top-Level Domain Distribution

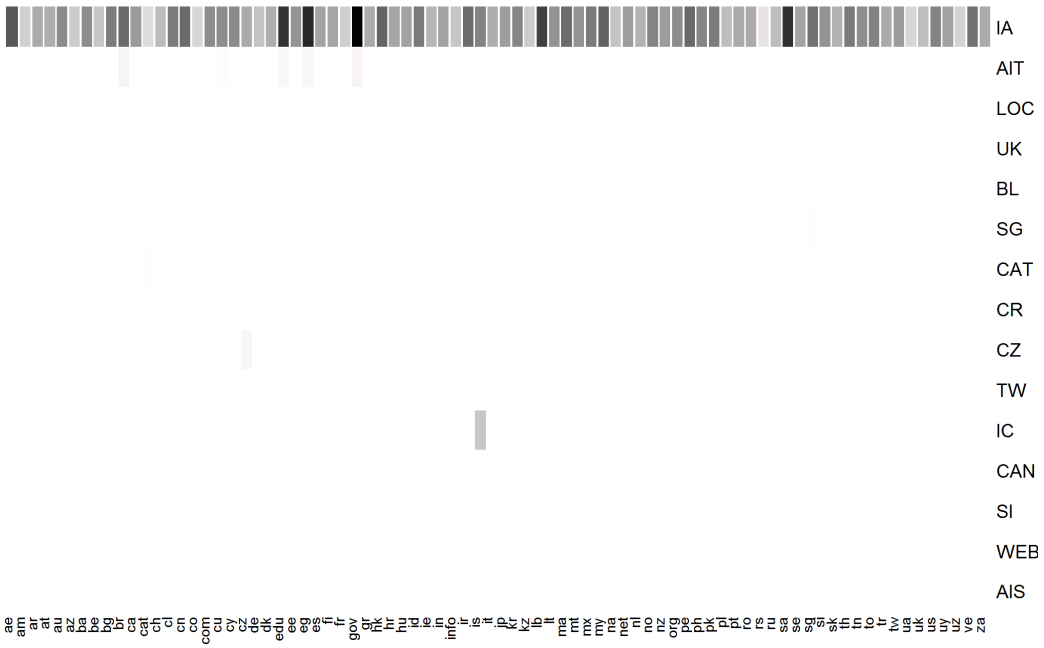
Figure 67 shows the coverage for each TLD sample (on columns) by the web archives (on rows). Figures 67(a) and 67(b) show $Coverage_{URI}$ and $Coverage_{MEM}$ respectively.

In Figure 67(a), we count the number of URIs for which an archive has at least one memento. A white block means $Coverage_{URI}(TLDSample) = 0\%$, a black block means $Coverage_{URI}(TLDSample) = 100\%$, where the archive has at least one memento for every URI in this sample TLD. The national libraries web archives show good $Coverage_{URI}$ for their national domains, for example, .is@IC=99.5%, .sg@SG=98.5%, .si@SI=22%, .cz@CZ=93%, .hr@CR=20%, and .cat@CAT=86%,. IC, CZ, and UK extend their crawling activity beyond these domains. Archive.is and Archive-It show a broad range of coverage through different domains. Figure 67(b) extends the results to include $Coverage_{MEM}$. The results show $Coverage_{MEM}@IA$ is the highest in all domains over the rest of archives except the Icelandic archive that has good $Coverage_{MEM}(.is\,Sample)@IC$.

Figure 68 shows the top TLDs per archive from both the fulltext search and the DMOZ TLD sample. Figure 68(a) is computed by analyzing the results from the fulltext search query as discussed in table 28. We extract the TLD for each URI, then we compute the percentage of each TLD in each archive. Figure 68(b) is computed by analyzing the discovered URIs on each archive using the DMOZ TLD sample. For each archive, we compute the number of URIs per TLD that show at least one memento in this archive. We sort these domains by the number of discovered URIs and show the highest 12 TLDs. IA and AIS have such broad coverage for all domains, no single domain dominates the collection for this DMOZ TLD sample shown in figure 68(b). The result is a somewhat counterintuitive image where .com is missing. The results in both figures show that there is a high correlation between both interfaces. Even though the national archives have excellent coverage of their associated domains, they have a surprising number of URIs from other domains as well. For example, TW supports a set of regional domains (i.e., .cn, .jp, and .sg). Figures 69 and 70 illustrate the normalized URI coverage ($NCoverage_{URI}@WA$ defined in Equation 20) and normalized memento coverage ($NCoverage_{MEM}@WA$ defined in Equation 21) for each domain per archive. Few archives show high $NCoverage_{URI}$ for the general domains over the others archives, for example, on average IA=0.68, AIS=0.14, CR=0.06, and AIT=0.05. However the national archives

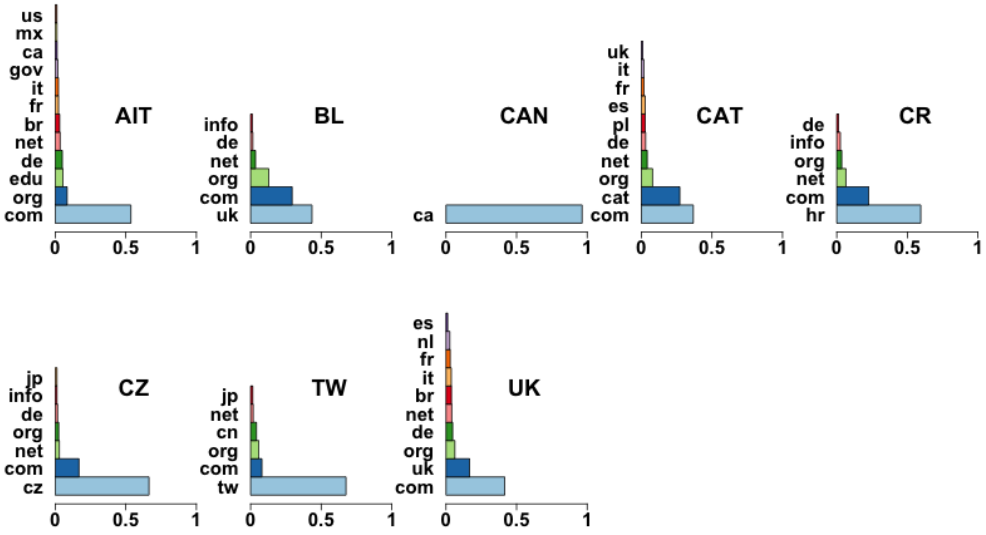


(a) Heat map of $Coverage_{URI}$ for TLD samples.

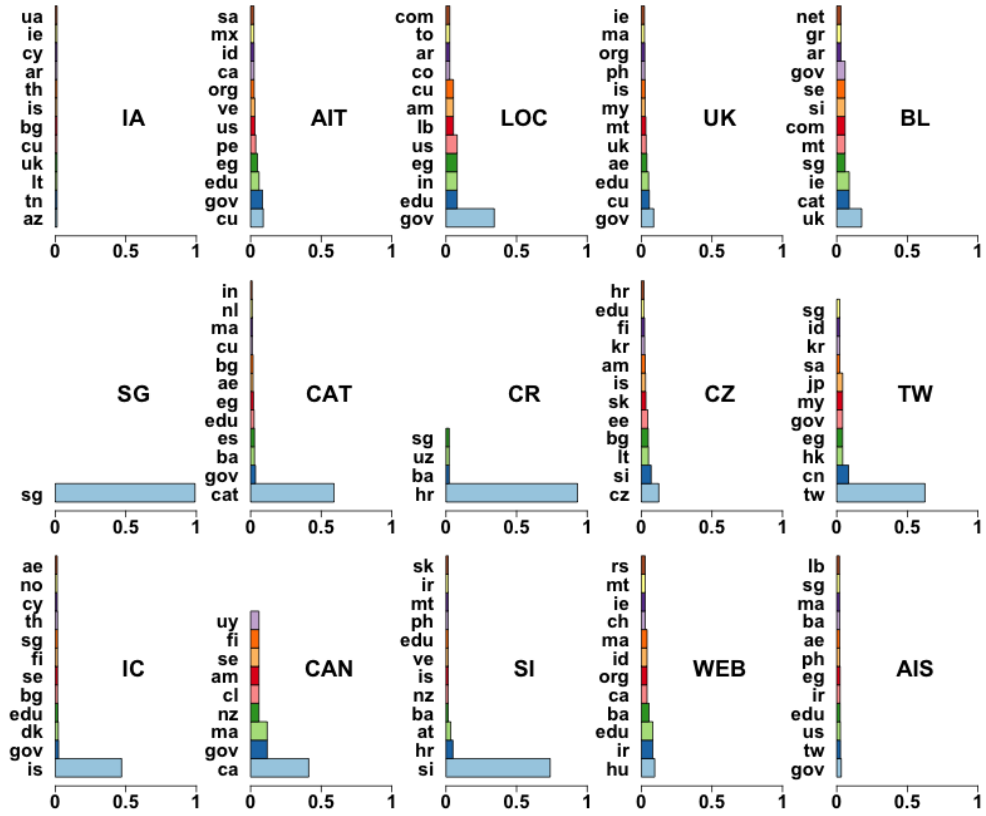


(b) Heat map of $Coverage_{MEM}$ for TLD samples.

Fig. 67 Heat map of archive coverage for TLD samples.

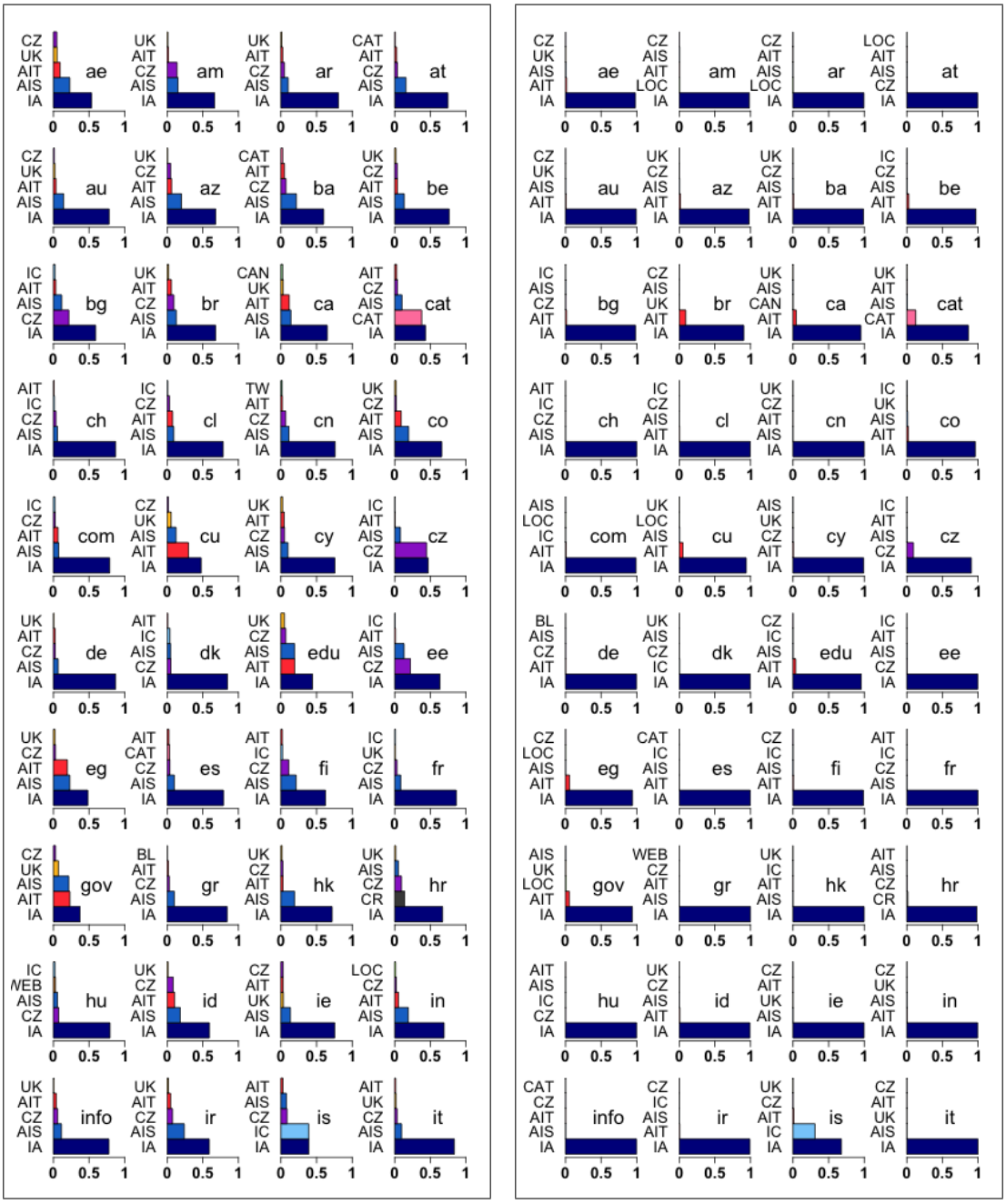


(a) The distribution of TLDs per archive (Full-text search).



(b) The distribution of TLDs per archive (DMOZ TLD sample).

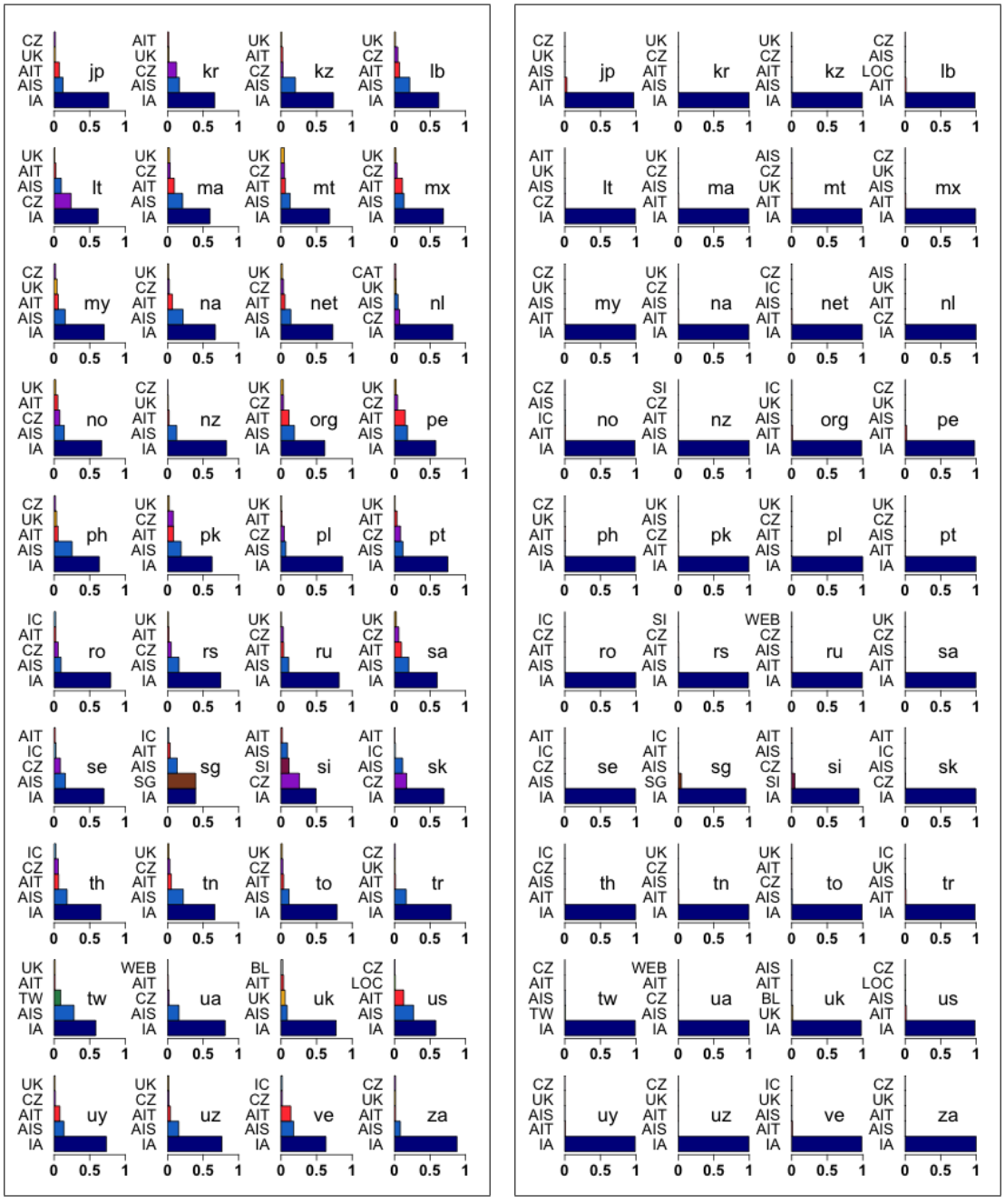
Fig. 68 The distribution of the TLDs through the archives.



(a) $NCoverage_{URI}$.

(b) $NCoverage_{MEM}$.

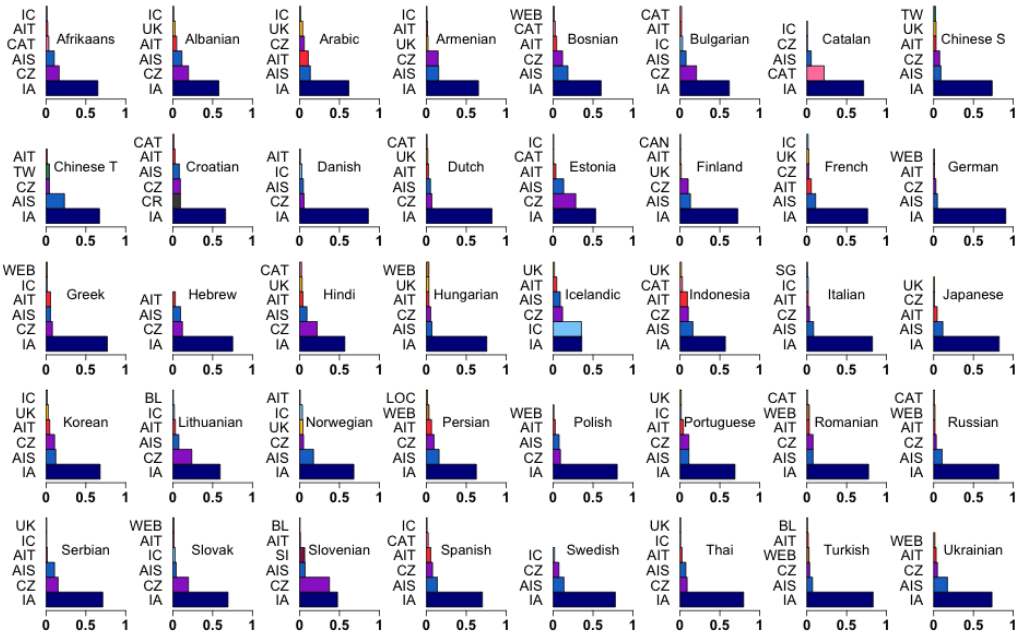
Fig. 69 Top-level domain distribution across the archives for DMOZ TLD sample (TLDs: ae to it).



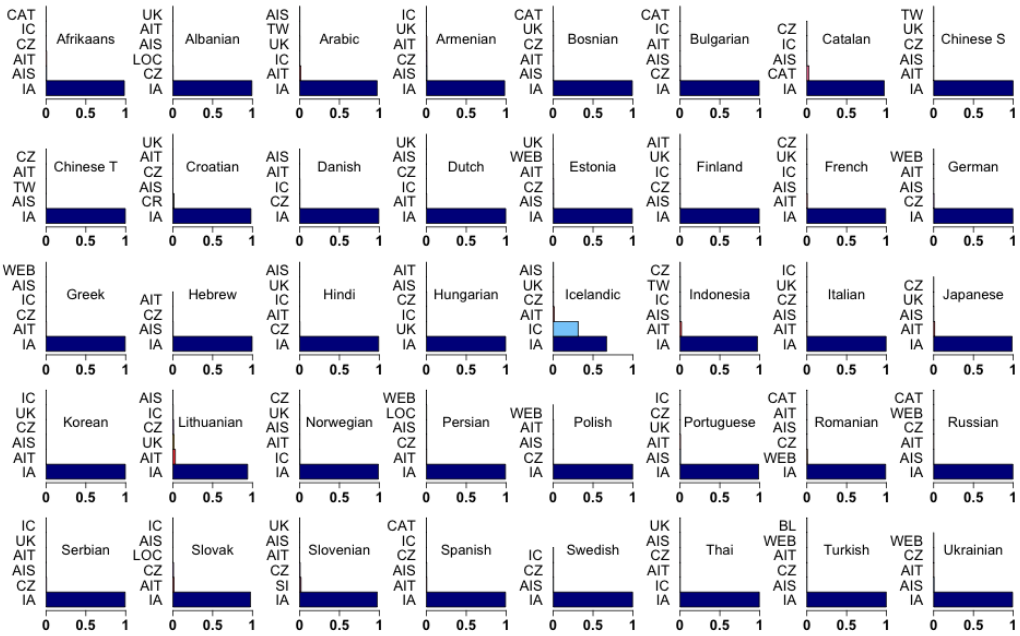
(a) $NCoverage_{URI}$.

(b) $NCoverage_{MEM}$.

Fig. 70 Top-level domain distribution across the archives for DMOZ TLD sample (TLDs: jp to za).



(a) $NCoverage_{URI}$.



(b) $NCoverage_{MEM}$.

Fig. 71 Language distribution per archive using DMOZ Language sample.

are doing similar $NCoverage_{URI}$ for their domains (e.g., CAT for `.cat` = 0.37 and IC for `.is` = 0.38). Figures 69 and 70 will be the basic for the web archive selection evaluation. Regarding the `http://www.japantimes.co.jp` example in section 1, the UK Web Archive at British Library (BL) does not contain .jp TLD, but it is highly probable to be found in IA, AIT, TW, and AIS.

Language Distribution

Figure 71 shows the $NCoverage$ for each web archive divided by the language sample. The highest coverage was $NCoverage_{URI}@IA = 0.70$ and $NCoverage_{MEM}@IA = 0.98$ are the highest among other archives. AIS and CZ have high $NCoverage_{URI} = 0.10$ for both but limited $NCoverage_{MEM} < 0.001$. The national libraries show good $NCoverage_{URI}$ for their national languages such as: Icelandic@IC=0.35, Catalan@CAT=0.21, and Croatian@CR=0.09. However we use the DMOZ language sample to avoid bias to the English language. The language distribution results appear to be similar to the previous results.

Growth Rate

Figure 72 shows the growth rate for each archive through time. The growth rate is computed by counting the number of new original URIs that appeared each month and the number of the new mementos that appeared each month. The growth rate is accumulated and normalized for the number of mementos and the number of new URIs added each month. TheFigure shows both LOC and CAN covered a limited period of time, then they stopped their crawling activities as their number of new URIs does not increase. CAT stopped adding new URIs a few years ago, but they continue to crawl the same URIs, resulting in more mementos. ThisFigure also gives an idea about the start date for each archive. For example, IA and CZ are the only archives that began before 2000.

Discussion

Reflecting on the findings presented in this section, it is clear that IA is the largest web archive, with mementos for 90% of the dataset, and AIT is in second place with 10%. This could be due in part to a bias in the dataset toward IA holdings, with logs from the Wayback Machine and the Memento Aggregator as well as from DMOZ, a known seed URI list for IA. Although we attempt to include content from a variety of archives, producing an unbiased dataset is difficult [35]. Another possible explanation is simply that IA has the oldest holdings starting from 1996.

There are surely other public web archives that exist that we simply did not know of. Some regions of the world do not appear to have active, public web archiving projects such as India and Africa. There are ongoing projects for the Arabic content by Bibliotheca Alexandrina and Latin America by the University of Texas[15]. Starting at about 2005 there appears to be a watershed moment for web archiving, when many projects begin to significantly grow their collections.

[15] `http://lanic.utexas.edu/project/archives/`

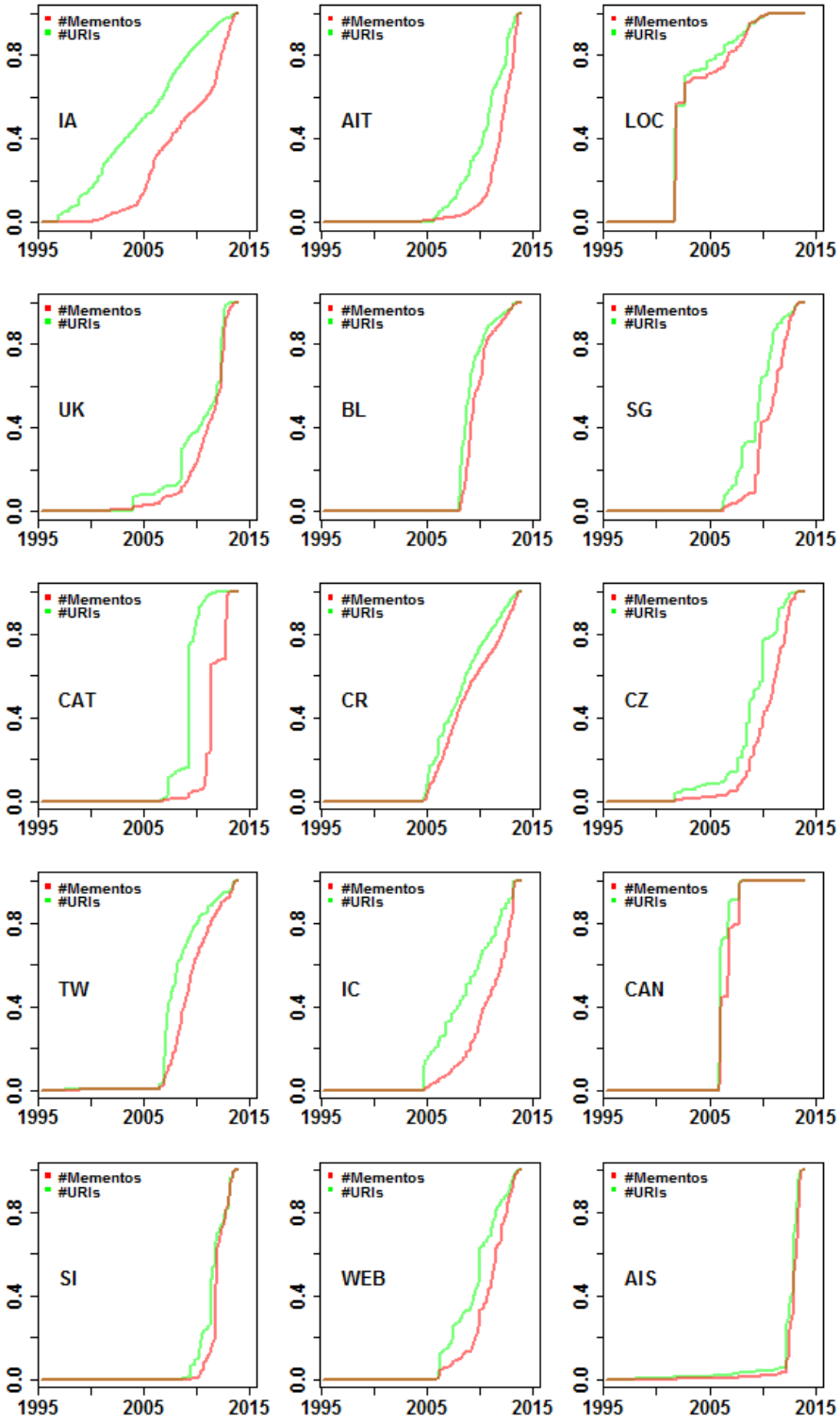
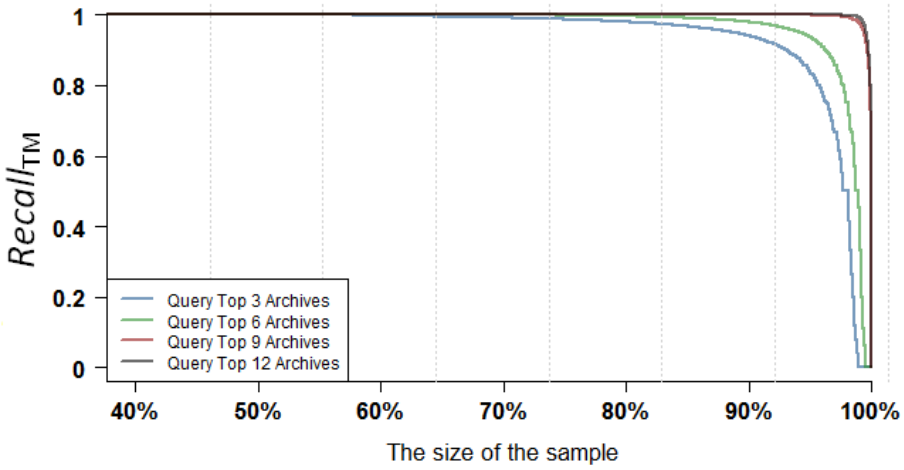
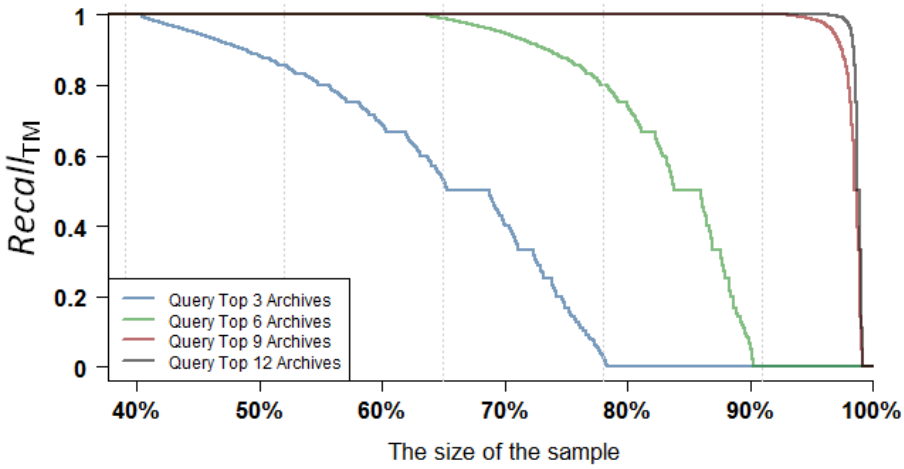


Fig. 72 Web Archive's corpus growth rate for URIs and Mementos.



(a) Including Internet Archive.



(b) Excluding Internet Archive.

Fig. 73 Query routing evaluation using TLD profile.

Building Web Archive Profile

The web archive profile is a description of the web archive holding. We use the TLD sample results (as appeared in Figure 68(b)) to fill the TLD field. The TLD field will be used as the main criteria for the web archive selection in the query routing. The language sample results (as appeared in Figure 71(a)) were used to fill the language field in the web archive profile. The language field is not used in this experiment, but it could help in the query routing for the full-text search. The growth rate results (as appeared in Figure 72) could be used to fill the age and the growth rate fields in the web archive profile. The growth rate could be used in ranking the archive for the requests that define specific datetime, e.g., Accept-Datetime header [278].

8.2.5 WEB ARCHIVE SELECTION

The Memento Aggregator performs a distributed search across the web archives. We assume each web archive query costs the same time for serving the request. As the number of archives grows, the performance will be worse. Even if the requests are performed in parallel, we still have the computation and communication costs. The Memento Aggregator uses the web archives profile to select a subset of the web archives to query in order to search for the mementos of the requested URI. First, the Memento Aggregator ranks the available web archives based on the profile information. Then, it extracts the selection criteria from the requested URI and selects the top subset from the archives list. For example, the Memento Aggregator could rank the web archives based on the TLD information, so for each TLD, it can rank the web archives based on the probability of the availability of this TLD. At query time, the Memento Aggregator will extract the TLD from the requested URI, then select the top-ranked web archives for this TLD. The selection algorithm aims to minimize the cost (minimize the number of queried web archives) and increase the efficiency (increase the number of retrieved mementos).

Recall$_{TM}$

To define the success criteria, we will revisit the "Recall" concept [52]. In URI-lookup environment, for a given query R and a given set of all the available mementos formatted as TimeMap. *Recall*$_{TM}$ is the fraction of TimeMap that has been returned by the request. For example, assume $TM(R) = \{M_1, M_2, M_3, M_4, M_5\}$. Suppose we query service S with R and service S responds with $TM(R)@S = \{M_1, M_2, M_3\}$. *Recall*$_{TM}$ as 0.6 (3 retrieved mementos out of 5 available mementos).

$$Recall_{TM} = \frac{\#\text{Relevant Retrieved mementos}}{|TM|} \tag{22}$$

where $|TM|$ = Size of TimeMap

$$Recall_{TM}(n) = \frac{|TM| \text{ using } n \text{ archives}}{|TM| \text{ using } N \text{ archives}} \tag{23}$$

In the Memento Aggregator, we extend *Recall*$_{TM}$ to include the number of queried web archives. Assume n is the number of queried web archives where $n \leq N$ and N is the number of available web archives. *Recall*$_{TM}(n)$ is the fraction of retrieved mementos that have been returned by n archives out of the total number of available mementos on N archives (Equation 23). Our goal is maximizing *Recall*$_{TM}(n)$ using the lowest possible n archives.

Experiment and Results

In this experiment, the Memento Aggregator will build a ranking mechanism by using the TLD field in the web archive profiles that is calculated based on the $NCoverage_{URI}$ as appeared inFigures 69(a) and 70(a). This selection algorithm is used in optimizing query routing for the Memento Aggregator.

In order to quantify the potential of a profile-based approach for the Memento Aggregator, we apply ten-fold cross-validation. For each URI, we query the Aggregator with a different subset of web archives using the top 3, top 6, top 9, and top 12 archives based on the requested URI's TLD.

Table 31 The Memento Aggregator average $Recall_{TM}$ values with different numbers of archives.

#Archives	Including IA	Excluding IA
$Recall_{TM}(3)$	0.960	0.647
$Recall_{TM}(6)$	0.980	0.830
$Recall_{TM}(9)$	0.998	0.983
$Recall_{TM}(12)$	0.999	0.987

Table 32 The percentage of TimeMaps that reached $Recall_TM(n) = 1.0$.

#Archives	Including IA	Excluding IA
$Recall_{TM}(3)$	57%	40%
$Recall_{TM}(6)$	74%	63%
$Recall_{TM}(9)$	95%	93%
$Recall_{TM}(12)$	97%	96%

We compute $Recall_{TM}(n)$ where $n \in \{1, 2, ..., N\}$. Here, the maximum number of archives is $N = 15$. We ran the experiment with all 15 archives and then repeated it excluding IA. $Recall_{TM}(3), (6), (9), and (12)$ are listed in Table 31 with including and excluding IA. Using just three archives achieves $Recall_{TM} = 0.96$ on average. Increasing the number of archives increases $Recall_{TM}$.

Figure 73(a) shows $Recall_{TM}(n)$ percentage for each TimeMap. $Recall_{TM}(3), (6), (9), and (12)$ are calculated for each URI and draw the values in descending order. The figure shows that 57% of TimeMaps got $Recall_{TM}(3) = 1.0$ while 75% of the TimeMaps reached $Recall_{TM}(6) = 1.0$. Table 32 shows the percentage of the TimeMaps that reached $Recall_{TM}(n) = 1.0$. Excluding Internet Archive, we were still able to achieve $Recall_{TM}(3) = 1.0$ in 40% of the cases.

What Figure 73(b) and Table 31 show is that even though the Internet Archive is the dominant archive in terms of URI coverage and number of mementos, even if the Internet Archive went away the aggregation of the other existing public web archives can service most needs. For example, in 96% of the cases using the next top 12 archives after the Internet Archive would result in no loss of mementos.

8.3 RELATED WORK AND BACKGROUND

Distributed information retrieval (also called federated search and metasearch) has been applied in different domains: databases [111, 189], metasearch engines [207, 252], and peer-to-peer (P2P) networks [296, 190]. Meng et al. [207] listed the motivation behind the federated search in the web as increasing the search coverage, solving the scalability of searching the web, facilitating the invocation of multiple search engines, and improving the retrieval effectiveness. Callan et al. [86, 251, 207] discussed the distributed information retrieval domain and defined three main problems:

- *Resource description*, how the resource/entity could be described; it includes defining the representation of the resource and the mechanism to build this representation.

- *Resource selection*, based on the previous description how the right resource/entity could be selected and searched.

- *Results merging*, how the results that retrieved from each resource/entity could be merged and ranked.

For the resource description problem, Shokouhi and Si [251] described the general technique to build collection representation in uncooperative environments: sending sampling (probe) queries to each collection, then gathering the received responses. The process is defined in three steps: 1) select query term and send it to the collection, 2) gather the top n retrieved documents, and 3) repeat these two steps until you reach your stopping criteria. The various algorithms vary in the details of each step. For the query term selection, Callan et al. [87, 88] proposed a query based on sampling with two variations: Random Sampling - Other Resource Description (RS-Ord), which picks the query terms randomly from the dictionary; Random Sampling - Learned Resource Description (RS-Lrd), which picks the next query term from the already sample documents. Ipeirotis et al. [151, 150] used "focused query probes" by sending a query based on a specific hierarchical taxonomy. Bar-Yossef and Gurevich [57] and Thomas and Hawking [273] proposed methods to avoid the search engine ranking bias in the uncooperative environment. The top n documents have been determined empirically as $n = 4$ by Callan and Connell [87]. The stopping criteria is usually defined in terms of the number of documents sampled or the number of sampling queries that have been issued. Callan and Connell [87] proposed that 300 - 500 documents are enough, but Shokouhi et al. [250] and Baillie et al. [54] proposed an adaptive system where the number of documents should vary from one collection to another allowing the algorithm to stop based on the rate of new terms that appeared in the sample documents.

For the resource selection problem, the broker ranks and selects a subset of the resources to be searched/queried based on the resource description. Gravano et al. [129, 130] proposed GlOSS (Glossary-of-Servers Server). GlOSS uses the statistical metadata about the term frequency at each source. It has two variations: bGlOSS (binary GlOSS) works with Boolean engines; and vGlOSS (vector GlOSS) works with vector-space search engines. Callan et al. [86, 89] proposed CORI (Collection Retrieval Interference network), which applies inference networks for collection selection. Powell and French [231] and Craswell et al. [104] showed that CORI is the most effective selection algorithm.

8.4 SUMMARY

In this chapter, we proposed an automatic technique to construct profiles for web archives. The results showed that the Internet Archive is the largest and has the broadest coverage. The national archives have good coverage of their domains and languages, and some of them extend their selection policies to cover more domains. We used the profiles in query routing optimization for the Memento Aggregator that selects the most probable web archives by matching the profiles with the query request. We proposed $Recall_{TM}(n)$ to evaluate the success of the web archive selection algorithm. We achieved $Recall_{TM}(3) = 0.96$. Even when the IA is excluded, we achieved $Recall_{TM}(3) = 0.47$.

In a future study, we plan to profile more characteristics such as respecting robots.txt, crawling

frequency, and crawling depth. Also, we will use full-text search to profile more characteristics in addition to the URI-lookup. The profile updating mechanism and frequency should be investigated.

CHAPTER 9

CONCLUSIONS AND FUTURE WORK

9.1 CONCLUSIONS

Web archives are performing an important role in preserving our culture heritage. The ways to discover and benefit from the preserved data are not revealed yet and have a lot of opportunities. Supporting the web archives with APIs access points will encourage third-party developers and researchers to build their customized applications that enrich access to and use of web archives.

The dissertation presents new service framework levels that will help the modularity of the web archive API development. The discussion in Chapter 4 explored these levels and the relationship between them.

The dissertation shows that the live web techniques may need modification to be applied in the web archive field. The temporal dimension of the data adds another level of complexity that should be solved separately. For example, Chapter 6 explores the creation of the temporal web graph and how it may differ from the regular web graph.

The dissertation gives the foundation of the novel API framework for web archives that works as middle-layer between the web archive and third-party developers. The dissertation is supported with a set of proposed applications that may benefit from the API. The dissertation encourages the web archive community to start to consider the implementation of these techniques.

We conclude a set of interesting results from this research:

- APIs are essential part for many use cases. We show various applications that could be built with the existence of this APIs.
- ArcSys system and its subsystems share a common interface technology that could be easily aggregated to give a complete information about the URI from different subsystems.
- Big data techniques could play an important role in developing new solutions to the web archive to overcome the increase in the corpus size.
- The different types of metadata that are extracted or derived from the web archives require novel computation techniques.
- Using crawler log values optimizes the processing of web archive corpus. For example, using the crawler log reduces the input corpus to 30% of the original size, and using the datetime values reduces the required number of thumbnails to 23% on average per TimeMap.
- We estimate the archived web to be between 33% to 95% based on the source of the sampled URI.

- We characterize the web archives' holding and formalize them in a "Web Archive Profile". The results prove that Internet Archive has the largest and oldest archived materials among all the web archives, and its scope covers all the domains and all the languages. The national libraries have good coverage for their national domains.

9.2 CONTRIBUTIONS

The dissertation has several contributions in the web archiving field:

1. The dissertation proposes a new service framework that depends on dividing the web archive corpus into four levels: *Content*, *Metadata*, *URI*, and *Archive*. We develop ArcSys as an integrated system to provide a coherent interface between web archives.

2. The development of ArcContent, the content service that is support the web archive interface with extracted versions of the mementos based on a set of predefined filters.

3. The development of ArcLink, a distributed system to extract, preserve, and expose the temporal web graph. The dissertation shows the challenges in building and the opportunities in using the temporal web graph.

4. The dissertation studies the optimization and summarization techniques to create the thumbnails for the web graph collections based on the HTML features such as SimHash fingerprints.

5. We extended the concept of URI-lookup in the web archive to include the HTTP status code to use the HTTP redirection to improve the discovery of mementos.

6. The dissertation covers the statistics about the web archive collections by estimating how much of the web is archived.

7. The concept of "Web Archive Profile" to characterize the web archive corpus is defined with an application on the distributed search in the Memento Aggregator.

9.3 FUTURE WORK

This research is the foundation for developing a standard service framework for the web archive. Each service may need advanced research to optimize the creation and preservation. ArcContent provides a set of filters on the top of web archive collection. The extraction and indexing techniques could be enhanced to provide faster processing. ArcContent could be extended to cover HTML transformation such as DOM tree serialization and version comparison. This dissertation covers two metadata techniques, implemented in ArcLink and ArcThumb. The metadata level has more information to be extracted such as title and description. The usage of ArcLink database as a foundation for temporal page rank techniques should be studied in detail, and the algorithm could improve the web archive full-text search results. The web archive profile could be extended to include more characteristics such as respecting robots.txt, crawling frequency, and crawling depth. The current profile structure depends on the URI-lookup technique. The profile could be improved

to include information the web archive content to support full-text distributed search between the archives. Finally, the updating mechanism should be investigated.

REFERENCE

[1] Apache Hadoop. URL http://hadoop.apache.org/

[2] British Library - Nominate a Site. URL http://www.webarchive.org.uk/ukwa/info/nominate

[3] International Internet Preservation Consortium (IIPC). URL http://www.netpreserve.org

[4] Internet Archive FAQ. URL http://www.archive.org/about/faqs.php#103

[5] Internet Archive FAQ - The Wayback Machine. URL http://www.archive.org/about/faqs.php#The_Wayback_Machine

[6] ISO 639-3. URL http://www-01.sil.org/iso639-3/

[7] Memento: Adding Time to the Web. URL http://www.mementoweb.org

[8] The Stanford WebBase Project. URL http://dbpubs.stanford.edu:8091/$\sim$testbed/doc2/WebBase/

[9] The Web Robots Pages. URL http://www.robotstxt.org/

[10] Web Archiving FAQs. URL http://www.loc.gov/webarchiving/faq.html#faqs_02

[11] PageValut (2003). URL http://www.projectcomputing.com/products/pageVault/

[12] DeepArc (2005). URL http://deeparc.sourceforge.net/

[13] Living Web Archives (LiWA) (2008). URL http://liwa-project.eu/

[14] Metadata Standards (2008). URL http://hul.harvard.edu/ois/digproj/metadata-standards.html

[15] Archiving of Databases: SIARD Suite (2009). URL http://www.bar.admin.ch/dienstleistungen/00823/00825/index.html

[16] ISO 28500:2009 Information and documentation – WARC file format (2009). URL http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=44717

[17] LAWA: Longitudinal Analytics of Web Archive data (2010). URL http://www.lawa-project.eu

[18] Memento Browser, A Web browser for Android (2010). URL http://code.google.com/p/memento-browser/

[19] My sites not archived! How can I add it? (2010). URL http://faq.web.archive.org/my-sites-not-archived-how-can-i-add-it/

[20] Twittervane: Crowdsourcing selection (2011). URL http://britishlibrary.typepad.co.uk/webarchive/2011/12/twittervane.html

[21] WayBack Releases: 1.6.0 Release notes. Tech. rep., Internet Archive (2011). URL http://archive-access.sourceforge.net/projects/wayback/release_notes.html

[22] Wayback software now Memento compliant (2011). URL http://archive-access.sourceforge.net/projects/wayback/release_notes.html

[23] Common Crawl (2012). URL http://commoncrawl.org

[24] Heritrix Crawler (2012). URL https://webarchive.jira.com/wiki/display/Heritrix/Heritrix

[25] Live Archiving Proxy (LAP) (2012). URL http://netpreserve.org/projects/live-archiving-http-proxy

[26] Memento Browser (2012). URL http://itunes.apple.com/us/app/memento-browser/id552478522

[27] Portuguese Web Archive - OpenSearch API (2012). URL https://code.google.com/p/pwa-technologies/wiki/OpenSearch

[28] Techtalk: Wayback and HDFS (2012). URL http://britishlibrary.typepad.co.uk/webarchive/2012/01/techtalk-wayback-hdfs.html

[29] WebART: Web Archive Retrieval Tools (2012). URL http://www.webarchiving.nl/

[30] Coca-Cola: A web archiving case study on preserving one of the most valuable and recognizable brands in the world. (2013). URL http://www.hanzoarchives.com/customers/coca-cola/

[31] Internet Archive - Wayback Machine APIs (2013). URL http://archive.org/help/wayback_api.php

[32] UK Web Archive Open Data (2013). URL http://data.webarchive.org.uk/opendata/

[33] Adar, E., Dontcheva, M., Fogarty, J., Weld, D.S.: Zoetrope: interacting with the ephemeral web. In: Proceedings of the 21st annual ACM symposium on User interface software and technology, UIST '08, pp. 239–248 (2008)

[34] Adle, M., Mitzenmacher, M.: Towards compressing Web graphs. In: Proceedings of Data Compression Conference, DCC 2001, pp. 203–212 (2001)

[35] Ainsworth, S.G., AlSum, A., SalahEldeen, H., Weigle, M.C., Nelson, M.L.: How much of the Web is Archived? In: Proceedings of the 11th annual international ACM/IEEE Joint Conference on Digital libraries, JCDL '11, pp. 133–136 (2011)

[36] Ainsworth, S.G., AlSum, A., SalahEldeen, H., Weigle, M.C., Nelson, M.L.: How Much of the Web Is Archived? Tech. Rep. arXiv:1212.6177 (2012). URL http://arxiv.org/abs/1212.6177

[37] AlNoamany, Y., Weigle, M.C., Nelson, M.L.: Access Patterns for Robots and Humans in Web Archives. Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital Libraries pp. 339–348 (2013)

[38] AlSum, A.: mcurl - Command Line Memento Client (2013). URL `http://ws-dl.blogspot.com/2013/05/2013-05-29-mcurl-command-line-memento.html`

[39] AlSum, A., Nelson, M.L.: ArcLink: Optimization Techniques to Build and Retrieve the Temporal Web Graph. Tech. rep., arXiv: 1305.5959 (2013)

[40] AlSum, A., Nelson, M.L.: ArcLink: optimization techniques to build and retrieve the temporal web graph. In: Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital libraries, JCDL '13, pp. 377–378. ACM Press, New York, New York, USA (2013)

[41] AlSum, A., Nelson, M.L.: Thumbnail Summarization Techniques for Web Archives. In: Proceedings of the 36th European Conference on Information Retrieval, ECIR 2014, pp. 299–310 (2014)

[42] AlSum, A., Weigle, M., Nelson, M., Sompel, H.: Profiling Web Archive Coverage for Top-Level Domain and Content Language. In: T. Aalberg, C. Papatheodorou, M. Dobreva, G. Tsakonas, C. Farrugia (eds.) Proceeding of the 17th International Conference of Theory of Practice of Digital Libraries, TPDL 2013, pp. 60–71. Springer Berlin Heidelberg (2013)

[43] Anand, A., Bedathur, S., Berberich, K., Schenkel, R.: Temporal Index Sharding for Space-time Efficiency in Archive Search. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '11, pp. 545–554. ACM, New York, NY, USA (2011)

[44] Anand, A., Bedathur, S., Berberich, K., Schenkel, R.: Index Maintenance for Time-travel Text Search. In: Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '12, pp. 235–244. ACM, New York, NY, USA (2012)

[45] Anand, A., Bedathur, S., Berberich, K., Schenkel, R., Tryfonopoulos, C.: EverLast: a distributed architecture for preserving the web. In: Proceedings of the 9th ACM/IEEE-CS joint conference on Digital libraries, JCDL '09, pp. 331–340. ACM (2009)

[46] Antoniades, D., Polakis, I., Kontaxis, G., Athanasopoulos, E., Ioannidis, S., Markatos, E.P., Karagiannis, T.: we.b: The web of short URLs. In: Proceedings of the 20th international conference on World Wide Web, pp. 715–724 (2011)

[47] Ashley, K.: What we want with webarchives: will we win? In: JISC, the DPC and the UK Web Archiving Consortium Workshop Missing links: the enduring web (2009)

[48] Aubry, S.: Introducing Web Archives as a New Library Service: the Experience of the National Library of France. LIBER Quarterly **20**(2), 179–199 (2010)

[49] Aula, A., Khan, R.M., Guan, Z., Fontes, P., Hong, P.: A comparison of visual and textual page previews in judging the helpfulness of web pages. In: Proceedings of the 19th international conference on World Wide Web, WWW '10, pp. 51–59. ACM Press (2010)

[50] Avcular, Y., Suel, T.: Scalable Manipulation of Archival Web Graphs. In: Proceeding of Large-Scale and Distributed Systems For Information Retrieval Workshop (LSDS-IR), pp. 27–32 (2011)

[51] Baeza-Yates, R., Castillo, C., Marin, M., Rodriguez, A.: Crawling a country: better strategies than breadth-first for Web page ordering. In: Special interest tracks and posters of the 14th international conference on World Wide Web, WWW '05, pp. 864–872 (2005)

[52] Baeza-Yates, R., Riberio-Neto, B.: Modern Information Retrieval: The Concepts and Technology behind Search, 2nd edn. Addison-Wesley Professional (2011)

[53] Bailey, S., Thompson, D.: UKWAC Building the UK's First Public Web Archive. D-Lib Magazine **12**(1) (2006)

[54] Baillie, M., Azzopardi, L., Crestani, F.: Adaptive Query-Based Sampling of Distributed Collections. In: F. Crestani, P. Ferragina, M. Sanderson (eds.) String Processing and Information Retrieval SE - 26, *Lecture Notes in Computer Science*, vol. 4209, pp. 316–328. Springer Berlin Heidelberg (2006)

[55] Balakireva, L.L.: SiteStory (2013). URL `http://mementoweb.github.io/SiteStory/`

[56] Bar-Yossef, Z., Broder, A.Z., Kumar, R., Tomkins, A.: Sic transit gloria telae: towards an understanding of the web's decay. In: Proceedings of the 13th international conference on World Wide Web, WWW '04, pp. 328–337 (2004)

[57] Bar-Yossef, Z., Gurevich, M.: Random sampling from a search engine's index. Journal of the ACM (JACM) **55**(5) (2008)

[58] Bar-Yossef, Z., Rajagopalan, S.: Template Detection via Data Mining and Its Applications. In: Proceedings of the 11th International Conference on World Wide Web, WWW '02, pp. 580–591. ACM, New York, NY, USA (2002)

[59] Baumgartner, R., Flesca, S., Gottlob, G.: Visual Web Information Extraction with Lixto. In: Proceedings of the 27th International Conference on Very Large Data Bases, VLDB '01, pp. 119–128. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2001)

[60] Becchetti, L., Castillo, C., Donato, D., Baeza-Yates, R., Leonardi, S.: Link analysis for Web spam detection. ACM Transactions on the Web **2**(1), 1–42 (2008)

[61] Ben Saad, M., Gançarski, S.: Archiving the web using page changes patterns: a case study. In: Proceeding of the 11th annual international ACM/IEEE Joint Conference on Digital libraries, JCDL '11, pp. 113–122 (2011)

[62] Ben Saad, M., Gançarski, S., Saad, M.B.: Improving the Quality of Web Archives through the Importance of Changes. In: Proceedings of the 22nd international conference on Database and expert systems applications, DEXA'11, pp. 394–409. Toulouse, France (2011)

[63] Berberich, K., Bedathur, S., Alonso, O., Weikum, G.: A Language Modeling Approach for Temporal Information Needs. In: Proceedings of the 32nd European Conference on Advances in Information Retrieval, ECIR '2010, pp. 13–25. Springer-Verlag, Berlin, Heidelberg (2010)

[64] Berberich, K., Bedathur, S., Neumann, T., Weikum, G.: A Time Machine for Text Search. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '07, pp. 519–526. ACM, New York, NY, USA (2007)

[65] Berners-Lee, T.: Information Management: A Proposal. Tech. rep., CERN (1990). URL `http://www.w3.org/History/1989/proposal.html`

[66] Berners-Lee, T.: Design Issues - Generic Resources. Tech. rep., W3C (2000). URL `http://www.w3.org/DesignIssues/Generic.html`

[67] Berners-Lee, T., Fielding, R., Masinter, L.: RFC 2396 - Uniform Resource Identifiers (URI): Generic Syntax (1998). URL `http://www.ietf.org/rfc/rfc2396.txt`

[68] Berners-Lee, T., Fielding, R., Masinter, L.: RFC 3986 - Uniform Resource Identifier (URI): Generic Syntax (2005). URL `http://www.ietf.org/rfc/rfc3986.txt`

[69] Bharat, K., Broder, A.: A technique for measuring the relative size and overlap of public Web search engines. Computer Networks and ISDN Systems **30**(1-7), 379–388 (1998)

[70] Bharat, K., Broder, A., Henzinger, M.R., Kumar, P., Venkatasubramanian, S.: The Connectivity Server: fast access to linkage information on the Web. Computer Networks and ISDN Systems **30**(1-7), 469–477 (1998)

[71] Bizer, C., Cyganiak, R., Heath, T.: How to Publish Linked Data on the Web (2007). URL `http://wifo5-03.informatik.uni-mannheim.de/bizer/pub/LinkedDataTutorial/`

[72] Bizer, C., Eckert, K., Meusel, R., Mühleisen, H., Schuhmacher, M., Völker, J.: Deployment of RDFa, Microdata, and Microformats on the Web A Quantitative Analysis. In: Proceedings of 12th International Semantic Web Conference, pp. 17–32 (2013)

[73] Bizer, C., Heath, T., Berners-Lee, T.: Linked Data - The Story So Far. International Journal on Semantic Web and Information Systems **5**(3) (2009)

[74] Boldi, P., Vigna, S.: The WebGraph Framework I : Compression Techniques. In: Proceedings of the 13h international conference on World Wide Web, WWW '04, pp. 595–602 (2004)

[75] Bordino, I., Boldi, P., Donato, D., Santini, M., Vigna, S.: Temporal Evolution of the UK Web. In: Proceedings of 2008 IEEE International Conference on Data Mining Workshops, pp. 909–918. IEEE (2008)

[76] Borthakur, D.: The Hadoop Distributed File System: Architecture and Design. Tech. rep., The Apache Software Foundation (2007). URL http://hadoop.apache.org/docs/r0.18.1/hdfs_design.pdf

[77] Braden, R.: RFC 1123 - Requirements for Internet Hosts - Application and Support (1989). URL http://www.ietf.org/rfc/rfc1123.txt

[78] Brin, S., Page, L.: The anatomy of a large-scale hypertextual Web search engine. Computer Networks and ISDN Systems **30**(1-7), 107–117 (1998)

[79] Broder, A., Glassman, S.: Syntactic clustering of the web. Computer Networks and ISDN Systems **29**(8-13) (1997)

[80] Brokes, A., Coufal, L., Flashkova, Z., Masanès, J., Oomen, J., Pop, R., Risse, T., Smulders, H.: Requirement Analysis Report "Living Web archive". Tech. rep., European Commission Seventh Framework Programme (2008). URL http://liwa-project.eu/images/uploads/d1-1.1_requirements_beg_v1.0.pdf

[81] Brown, A.: Archiving websites: a practical guide for information management professionals, 1st edn. Facet, London (2006)

[82] Brügger, N.: Archiving Websites. General Considerations and Strategies, 1st edn. The Center for Internet Research, Aarhus N (2005)

[83] Brügger, N.: Historical Network Analysis of the Web. Social Science Computer Review (2012)

[84] Brunelle, J.F.: Release of Warrick 2.0 Beta (2012). URL http://ws-dl.blogspot.com/2012/01/2012-01-23-release-of-warrick-20-beta.html

[85] Brunelle, J.F., Nelson, M.L., Balakireva, L., Sanderson, R., Van de Sompel, H.: Evaluating the SiteStory Transactional Web Archive with the ApacheBench Tool. In: Proceedings of the 17th International Conference on Theory and Practice of Digital Libraries, *TPDL 2013*, vol. 8092, pp. 204–215 (2013)

[86] Callan, J.: Distributed Information Retrieval. In: W. Croft (ed.) Advances in Information Retrieval SE - 5, *The Information Retrieval Series*, vol. 7, pp. 127–150. Springer US (2000)

[87] Callan, J., Connell, M.: Query-based sampling of text databases. ACM Transactions on Information Systems **19**(2), 97–130 (2001)

[88] Callan, J., Connell, M., Du, A.: Automatic discovery of language models for text databases. ACM SIGMOD Record **28**(2), 479–490 (1999)

[89] Callan, J.P., Lu, Z., Croft, W.B.: Searching distributed collections with inference networks. In: Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '95, pp. 21–28. ACM Press, New York, New York, USA (1995)

[90] Chakrabarti, S., Joshi, M.M., Punera, K., Pennock, D.M.: The structure of broad topics on the web. In: Proceedings of the 11th international conference on World Wide Web, WWW '02, pp. 251–260. ACM Press, New York, New York, USA (2002)

[91] Chang, C.H., Kayed, M., Girgis, M., Shaalan, K.: A Survey of Web Information Extraction Systems. IEEE Transactions on Knowledge and Data Engineering **18**(10), 1411–1428 (2006)

[92] Charikar, M.S.: Similarity estimation techniques from rounding algorithms. In: Proceedings of the thiry-fourth annual ACM symposium on Theory of computing, STOC '02, pp. 380–388 (2002)

[93] Chen, K., Chen, Y., Ting, P.: Developing National Taiwan University Web Archiving System. In: Proceedings of 8th International Web Archiving Workshop, IWAW '08 (2008)

[94] Cho, J., Garcia-Molina, H.: Effective page refresh policies for Web crawlers. ACM Transactions on Database Systems (TODS) **28**(4), 390–426 (2003)

[95] Cho, J., Garcia-Molina, H., Haveliwala, T., Lam, W., Paepcke, A., Raghavan, S., Wesley, G.: Stanford WebBase Components and Applications. ACM Transactions on Internet Technology **6**(2) (2006)

[96] Cho, J., Garcia-Molina, H., Page, L.: Efficient crawling through URL ordering. Computer Networks and ISDN Systems **30**(1-7), 161–172 (1998)

[97] Chu, W.T., Lin, C.H.: Automatic selection of representative photo and smart thumbnailing using near-duplicate detection. In: Proceeding of the 16th ACM international conference on Multimedia, MM '08, pp. 829–832 (2008)

[98] Clausen, L.R.: Overview of the Netarkivet web archiving system. In: Proceedings of 6th International Web Archiving Workshop, IWAW '06 (2006)

[99] Coelho, F., Ribeiro, C.: Image abstraction in crossmedia retrieval for text illustration. In: Proceedings of the 34th European conference on Advances in Information Retrieval, ECIR' 12, pp. 329–339 (2012)

[100] Costa, M., Gomes, D., Couto, F., Silva, M.: A Survey of Web Archive Search Architectures. In: Proceedings of the 22Nd International Conference on World Wide Web Companion, WWW '13 Companion, pp. 1045–1050. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland (2013)

[101] Costa, M., Silva, M.J.: Understanding the Information Needs of Web Archive Users. In: Proceedings of the 10th International Web Archiving Workshop, pp. 9–16. Vienna, Austria (2010)

[102] Costa, M., Silva, M.J.: Characterizing Search Behavior in Web Archives. In: Proceedings of the 1st International Temporal Web Analytics Workshop, *TWAW 2011*, vol. 707 (2011)

[103] Cowie, J., Lehnert, W.: Information extraction. Communications of the ACM **39**(1), 80–91 (1996)

[104] Craswell, N., Bailey, P., Hawking, D.: Server selection on the World Wide Web. In: Proceedings of the fifth ACM conference on Digital libraries, DL '00, pp. 37–46. ACM Press, New York, New York, USA (2000)

[105] Dean, J., Henzinger, M.R.: Finding related pages in the World Wide Web. Computer Networks **31**(11-16), 1467–1479 (1999)

[106] Debnath, S., Mitra, P., Pal, N., Giles, C.L.: Automatic identification of informative sections of Web pages. IEEE Transactions on Knowledge and Data Engineering **17**(9), 1233–1246 (2005)

[107] Denev, D., Mazeika, A., Spaniol, M., Weikum, G.: The SHARC Framework for Data Quality in Web Archiving. The VLDB Journal **20**(2), 183–207 (2011)

[108] Donato, D., Laura, L., Leonardi, S., Millozzi, S.: Large scale properties of the Webgraph. The European Physical Journal B - Condensed Matter **38**(2), 239–243 (2004)

[109] Dougherty, M., Meyer, E.T., Madsen, C., van den Heuvel, C., Thomas, A., Wyatt, S., Electronic: Researcher Engagement with Web Archives: State of the Art. Tech. rep., Joint Information Systems Committee (JISC) (2010). URL `http://ssrn.com/abstract=1714997`

[110] Drijfhout, W., Jundt, O., Wevers, L., Hiemstra, D.: Traitor: Associating Concepts using the World Wide Web. In: Proceedings of the 13th Dutch-Belgian Workshop on Information Retrieval, *DIR*, vol. 986, pp. 56–57. CEUR-WS.org (2013)

[111] D'Souza, D.J., Thom, J.A., Zobel, J.: A comparison of techniques for selecting text collections. In: Proceedings of 11th Australasian Database Conference, ADC 2000, pp. 28–32 (2000)

[112] Dyreson, C.E., Lin, H.l., Wang, Y.: Managing versions of web documents in a transaction-time web server. In: Proceedings of the 13th international conference on World Wide Web, WWW '04, pp. 422–432 (2004)

[113] Erdélyi, M., Benczúr, A.A.: Temporal Analysis for Web Spam Detection: An Overview. In: Proceedings of Temporal Web Analytics Workshop, TempWeb 2011, pp. 17–24 (2011)

[114] Erdélyi, M., Benczúr, A.A., Masanés, J., Siklósi, D.: Web spam filtering in internet archives. In: Proceedings of the 5th International Workshop on Adversarial Information Retrieval on the Web, AIRWeb '09, pp. 17–20. ACM (2009)

[115] Fetterly, D., Manasse, M., Najork, M.: Spam, damn spam, and statistics: using statistical analysis to locate spam web pages. In: Proceedings of the 7th International Workshop on the Web and Databases, WebDB '04, pp. 1–6 (2004)

[116] Fetterly, D., Manasse, M., Najork, M., Wiener, J.L.: A large-scale study of the evolution of web pages. Software: Practice and Experience **34**(2), 213–237 (2004)

[117] Fielding, R.T., Gettys, J., Mogul, J.C., Frystyk, H., Masinter, L., Leach, P.J., Berners-Lee, T.: RFC 2616 - Hypertext Transfer Protocol (1999). URL `http://www.ietf.org/rfc/rfc2616.txt`

[118] Flanders, D., Ramsey, M., McGregor, A.: The advantage of APIs How to jump the information gap. Tech. rep., Joint Information Systems Committee (JISC) (2012). URL http://www.jisc.ac.uk/reports/the-advantage-of-apis

[119] Franz, A., Brants, T.: All Our N-gram are Belong to You. URL http://googleresearch.blogspot.com/2006/08/all-our-n-gram-are-belong-to-you.html

[120] Freed, N., Borenstein, N.: RFC 2045 - Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies (1996). URL https://www.ietf.org/rfc/rfc2045.txt

[121] Garrett, J.J.: Ajax: A new approach to web applications (2005). URL http://www.adaptivepath.com/ideas/ajax-new-approach-web-applications/

[122] George, L.: HBase: The Definitive Guide. O'Reilly Media, Inc. (2011)

[123] Goel, V.: Web Archive Metadata File Specification (2011). URL https://webarchive.jira.com/wiki/display/Iresearch/Web+Archive+Metadata+File+Specification

[124] Gomes, D., Cruz, D., Miranda, J.a., Costa, M., Fontes, S.a.: Search the Past with the Portuguese Web Archive. In: Proceedings of the 22Nd International Conference on World Wide Web Companion, WWW '13 Companion, pp. 321–324. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland (2013)

[125] Gomes, D., Miranda, J.a., Costa, M.: A Survey on Web Archiving Initiatives. In: Proceedings of the 15th International Conference on Theory and Practice of Digital Libraries: Research and Advanced Technology for Digital Libraries, TPDL'11, pp. 408–420. Springer-Verlag, Berlin, Heidelberg (2011)

[126] Gomes, D., Nogueira, A., Miranda, J.a., Costa, M.: Introducing the Portuguese Web Archive Initiative. In: Proceedings of 8th International Web Archiving Workshop, IWAW '08 (2008)

[127] Gomes, D., Santos, A.L., Silva, M.J.: Managing duplicates in a web archive. In: Proceedings of the 2006 ACM symposium on Applied computing, SAC '06, pp. 818–825. ACM (2006)

[128] Graham, A., Garcia-Molina, H., Paepcke, A., Winograd, T.: Time as essence for photo browsing through personal digital libraries. In: Proceedings of the second ACM/IEEE-CS joint conference on Digital libraries, JCDL '02, pp. 326–335 (2002)

[129] Gravano, L., García-Molina, H., Tomasic, A.: The effectiveness of GIOSS for the text database discovery problem. ACM SIGMOD Record **23**(2), 126–137 (1994)

[130] Gravano, L., García-Molina, H., Tomasic, A.: GlOSS: text-source discovery over the Internet. ACM Transactions on Database Systems **24**(2), 229–264 (1999)

[131] Greenberg, J.: Metadata extraction and harvesting: A comparison of two automatic metadata generation applications. Journal of Internet Cataloging **6**(4), 59–82 (2004)

[132] Grotke, A.: IIPC 2008 Member Profile Survey Results. Tech. rep., International Internet Preservation Consortium Publications (2008). URL http://www.netpreserve.org/resources/2008-iipc-member-profile-survey-results

[133] Grotke, A.: WARC File Format Published as an International Standard (2009). URL http://netpreserve.org/press/pr20090601.php

[134] Guillaume, J., Latapy, M., Viennot, L.: Efficient and Simple Encodings for the Web Graph. In: Proceedings of the Third International Conference on Advances in Web-Age Information Management, WAIM '02, pp. 328–337 (2002)

[135] Gulli, A., Signorini, A.: The indexable web is more than 11.5 billion pages. In: International World Wide Web Conference, pp. 902–903 (2005)

[136] Gupta, S., Kaiser, G., Neistadt, D., Grimm, P.: DOM-based Content Extraction of HTML Documents. In: Proceedings of the 12th International Conference on World Wide Web, WWW '03, pp. 207–214. ACM, New York, NY, USA (2003)

[137] He, J., Suel, T.: Faster Temporal Range Queries over Versioned Text. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '11, pp. 565–574. ACM, New York, NY, USA (2011)

[138] He, J., Zeng, J., Suel, T.: Improved Index Compression Techniques for Versioned Document Collections. In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management, CIKM '10, pp. 1239–1248. ACM, New York, NY, USA (2010)

[139] Helen Hockx-Yu: UK Web Archive in the eyes of scholars (2012). URL http://britishlibrary.typepad.co.uk/webarchive/2012/07/uk-web-archive-in-the-eyes-of-scholars.html

[140] Henzinger, M.: Finding near-duplicate web pages. In: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '06, pp. 284–291 (2006)

[141] Heslop, H., Davis, S., Wilson, A.: An Approach to the Preservation of Digital Records. Tech. rep., National Archives of Australia (2002). URL http://www.naa.gov.au/Images/An-approach-Green-Paper_tcm16-47161.pdf

[142] Heuser, C.A., Mecca, G., Raunich, S., Pappalardo, A.: A new algorithm for clustering search results. Data & Knowledge Engineering **62**(3), 504–522 (2007)

[143] Hewitt, E.: Cassandra: The Definitive Guide. O'Reilly Media, Inc. (2010)

[144] Hockx-Yu, H.: The Past Issue of the Web. In: Proceedings of 3rd International Conference on Web Science, WebSci '11, pp. 1–8 (2011)

[145] Holtman, K., Mutz, A.: RFC 2295 - Transparent Content Negotiation in HTTP (1998). URL http://www.ietf.org/rfc/rfc2295.txt

[146] Howe, J.: The Rise of Crowdsourcing. WIRED Magazine (14.06) (2006)

[147] Huurdeman, H.C., Ben-David, A., Sammar, T.: Sprint Methods for Web Archive Research. In: Proceedings of the 5th Annual ACM Web Science Conference, WebSci '13, pp. 182–190. ACM, New York, NY, USA (2013)

[148] IIPC Access Working Group: Prototypes related to the IIPC Access Working Group Use Cases. Tech. rep., International Internet Preservation Consortium Publications (2006). URL `http://www.netpreserve.org/resources/prototypes-related-iipc-access-working-group-use-cases`

[149] IIPC Access Working Group: Use cases for Access to Internet Archives. Tech. rep., International Internet Preservation Consortium Publications (2006). URL `http://netpreserve.org/resources/use-cases-access-internet-archives`

[150] Ipeirotis, P.G., Gravano, L.: Distributed search over the hidden web: hierarchical database sampling and selection. In: Proceeding of the 28th Very-Large Database conference, VLDB '02, pp. 394–405 (2002)

[151] Ipeirotis, P.G., Gravano, L., Sahami, M.: Probe, count, and classify. ACM SIGMOD Record **30**(2), 67–78 (2001)

[152] Jacobs, I., Walsh, N.: Architecture of the World Wide Web. Technical report, W3C (2004). URL `http://www.w3.org/TR/webarch/`

[153] Janssen, W.C.: Document Icons and Page Thumbnails: Issues in Construction of Document Thumbnails for Page-Image Digital Libraries. In: Proceedings of 8th European Conference on Digital Libraries, ECDL, pp. 111–121 (2004)

[154] Jatowt, A., Kawai, Y., Nakamura, S., Kidawara, Y., Tanaka, K.: A browser for browsing the past web. In: Proceedings of the 15th international conference on World Wide Web, WWW '06, pp. 877—-878 (2006)

[155] Jatowt, A., Kawai, Y., Nakamura, S., Kidawara, Y., Tanaka, K.: Journey to the past: proposal of a framework for past web browser. In: Proceedings of the 17th conference on Hypertext and hypermedia, HYPERTEXT '06, pp. 135–144. ACM (2006)

[156] Jatowt, A., Kawai, Y., Ohshima, H., Tanaka, K.: What can history tell us?: towards different models of interaction with document histories. In: Proceedings of the 19th ACM conference on Hypertext and hypermedia, HT '08, pp. 5–14. ACM (2008)

[157] Jatowt, A., Kawai, Y., Tanaka, K.: Detecting age of page content. In: Proceedings of the 9th annual ACM international workshop on Web information and data management, WIDM '07, pp. 137–144. ACM (2007)

[158] Jatowt, A., Kawai, Y., Tanaka, K.: Visualizing historical content of web pages. In: Proceeding of the 17th international conference on World Wide Web, WWW '08, pp. 1221–1222. ACM (2008)

[159] Jatowt, A., Kawai, Y., Tanaka, K.: Page History Explorer: Visualizing and Comparing Page Histories. IEICE Transactions on Information and Systems **E94-D**(3), 564–577 (2011)

[160] Jiang, L., Wang, Y., Hoffart, J., Weikum, G.: Crowdsourced Entity Markup. In: Proceedings of the 1st International Workshop on Crowdsourcing the Semantic Web, Sydney, Australia, *CrowdSem 2013*, vol. 1030, pp. 59–68. CEUR-WS.org (2013)

[161] Kaasten, S., Greenberg, S., Edwards, C.: How People Recognise Previously Seen Web Pages from Titles, URLs and Thumbnails. In: People and Computers XVI - Memorable Yet Invisible SE, January, pp. 247–265. Springer London (2002)

[162] Kang, U., Tsourakakis, C.E., Faloutsos, C.: PEGASUS: mining peta-scale graphs. Knowledge and Information Systems **27**(2), 303–325 (2010)

[163] Kanhabua, N., Berberich, K., Nørvåg, K.: Learning to Select a Time-aware Retrieval Model. In: Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '12, pp. 1099–1100. ACM, New York, NY, USA (2012)

[164] Kanhabua, N., Nørvåg, K.: Determining Time of Queries for Re-ranking Search Results. In: Proceedings of the 14th European Conference on Research and Advanced Technology for Digital Libraries, ECDL'10, pp. 261–272. Springer-Verlag, Berlin, Heidelberg (2010)

[165] Kavcic-colic, A., Grobelnik, M.: Archiving the Slovenian Web : Recent Experiences. In: Proceedings of 4th International Web Archiving Workshop, IWAW '04 (2004)

[166] Kay, A.: The Best Way to Predict the Future is to Invent it. Mathematical Social Sciences **30**(3), 326 (1995)

[167] Kelly, B., Ashley, K., Guy, M., Pinsent, E., Davis, R., Hatcher, J.: PoWR: The Preservation of Web Resources Handbook. Tech. rep., Joint Information Systems Committee (JISC) (2008). URL `http://www.jisc.ac.uk/publications/programmerelated/2008/powrhandbook.aspx`

[168] Kelly, M., Weigle, M.C.: WARCreate. In: Proceedings of the 12th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL '12, pp. 437–438 (2012)

[169] Kherfi, M.L., Ziou, D.: Image Collection Organization and Its Application to Indexing, Browsing, Summarization, and Semantic Retrieval. IEEE Transactions on Multimedia **9**(4), 893–900 (2007)

[170] Klein, M.: Adventures with the Delicious API (2011). URL `http://ws-dl.blogspot.com/2011/03/2011-03-09-adventures-with-delicious.html`

[171] Klein, M.: Launching Synchronicity - A Firefox Add-on for Rediscovering Missing Web Pages in Real Time (2011). URL `http://ws-dl.blogspot.com/2011/06/2011-06-10-launching-synchronicity.html`

[172] Klein, M., Aly, M., Nelson, M.L.: Synchronicity - Automatically Rediscover Missing Web Pages in Real Time. In: Proceeding of the 11th annual international ACM/IEEE Joint Conference on Digital libraries, JCDL '11, pp. 475–476 (2011)

[173] Klein, M., Shipman, J.L., Nelson, M.L.: Is this a good title? In: Proceedings of the 21st ACM conference on Hypertext and hypermedia, HT '10, pp. 3–12 (2010)

[174] Klein, M., Ware, J., Nelson, M.L.: Rediscovering Missing Web Pages Using Link Neighborhood Lexical Signatures. In: Proceedings of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries, JCDL '11, pp. 137–140 (2011)

[175] Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. Journal of the ACM **46**(5), 604–632 (1999)

[176] Klyne, G., Carroll, J.J.: Resource Description Framework (RDF): Concepts and Abstract Syntax. Tech. Rep. W3C Recommendation 10 February 2004, W3C (2004). URL `http://www.w3.org/TR/rdf-concepts/`

[177] Kohlschütter, C., Fankhauser, P., Nejdl, W.: Boilerplate detection using shallow text features. In: Proceedings of the third ACM international conference on Web search and data mining, WSDM '10, pp. 441–450 (2010)

[178] Koster, M.: A Method for Web Robots Control (1996). URL `http://www.robotstxt.org/norobots-rfc.txt`

[179] Kovacevic, M., Diligenti, M., Gori, M., Milutinovic, V.: Recognition of common areas in a Web page using visual information: a possible application in a page classification. In: Proceedings of 2002 IEEE International Conference on Data Mining, ICDM 2003, pp. 250–257. IEEE Comput. Soc (2002)

[180] Kules, B., Wilson, M.L., Shneiderman, B.: From Keyword Search to Exploration: How Result Visualization Aids Discovery on the Web. Tech. rep., HCIL-2008-06 (2008). URL `http://hcil2.cs.umd.edu/trs/2008-06/2008-06.pdf`

[181] Laender, A.H.F., Ribeiro-Neto, B.A., da Silva, A.S., Teixeira, J.S.: A Brief Survey of Web Data Extraction Tools. ACM SIGMOD Record **31**(2), 84–93 (2002)

[182] Lam, H., Baudisch, P.: Summary thumbnails: Readable Overviews for Small Screen Web Browsers. In: Proceedings of the SIGCHI conference on Human factors in computing systems, CHI '05, pp. 681–690 (2005)

[183] Lassila, O.: Web metadata: a matter of semantics. IEEE Internet Computing **2**(4), 30–37 (1998)

[184] Lee, C.P., Golub, G.H., Zenios, S.A.: A Two-Stage Algorithm for Computing PageRank and Multistage Generalizations. Internet Mathematics **4**(4), 299–327 (2007)

[185] Leetaru, K.H.: A Vision of the Role and Future of Web Archives. Tech. rep., International Internet Preservation Consortium Publications (2012). URL http://netpreserve.org/resources/vision-role-and-future-web-archives

[186] Lin, S.H., Ho, J.M.: Discovering Informative Content Blocks from Web Documents. In: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '02, pp. 588–593. ACM, New York, NY, USA (2002)

[187] Liu, B., Grossman, R., Zhai, Y.: Mining Data Records in Web Pages. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '03, pp. 601–606. ACM, New York, NY, USA (2003)

[188] Lohmann, S., Ziegler, J., Tetzlaff, L.: Comparison of Tag Cloud Layouts: Task-Related Performance and Visual Exploration. In: Proceedings of the 12th IFIP TC 13 International Conference on Human-Computer Interaction: Part I, INTERACT '09, pp. 392–404 (2009)

[189] Losee, R., Church, L.: Information retrieval with distributed databases: analytic models of performance. IEEE Transactions on Parallel and Distributed Systems **15**(1), 18–27 (2004)

[190] Lu, J., Callan, J.: Federated Search of Text-Based Digital Libraries in Hierarchical Peer-to-Peer Networks. In: Proceedings of 27th European Conference on Information Retrieval Research, ECIR '05, pp. 52–66 (2005)

[191] Malewicz, G., Austern, M.H., Bik, A.J.C., Dehnert, J.C., Horn, I., Leiser, N., Czajkowski, G.: Pregel: A System for Large-Scale Graph Processing. In: Proceedings of the 2010 international conference on Management of data, SIGMOD '10, pp. 135–146 (2010)

[192] Manku, G.S., Jain, A., Das Sarma, A.: Detecting near-duplicates for web crawling. In: Proceedings of the 16th international conference on World Wide Web, WWW '07, pp. 141–149 (2007)

[193] Marks, J.: Federal agencies are gearing up for a massive data dump (2012). URL http://www.nextgov.com/big-data/2012/08/federal-agencies-are-gearing-massive-data-dump/57599/

[194] Marshall, C., McCown, F., Nelson, M.L.: Evaluating Personal Archiving Strategies for Internet-based Information. In: Proceedings of IS&T Archiving 2007, pp. 151–156 (2007)

[195] Masanès, J.: Web Archiving Methods and Approaches: A Comparative Study. Library Trends **54**(1), 72–90 (2005)

[196] Masanès, J.: Selection for Web Archives, pp. 71–91 (2006)

[197] Masanès, J.: Web archiving. Springer, Berlin, Heidelberg (2006)

[198] Mayer, R.E., Moreno, R.: Nine ways to reduce cognitive load in multimedia learning. Educational psychologist **38**(1), 43–52 (2003)

[199] Mayr, M.: Harvesting Practices Report. Tech. rep., International Internet Preservation Consortium Publications (2011). URL http://www.netpreserve.org/resources/harvesting-practices-report

[200] McCallum, A.: Information Extraction: Distilling Structured Data from Unstructured Text. ACM Queue **3**(9), 48–57 (2005)

[201] McCown, F.: Memento Browser for Android is available (2010). URL http://frankmccown.blogspot.com/2010/09/memento-browser-for-android-is.html

[202] McCown, F., Marshall, C.C., Nelson, M.L.: Why web sites are lost (and how they're sometimes found). Communications of the ACM **52**(11), 141–145 (2009)

[203] McCown, F., Nelson, M.: Recovering a Website's Server Components from the Web Infrastructure. In: Proceedings of the 8th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL '08, pp. 124–133. ACM, New York, NY, USA (2008)

[204] McCown, F., Nelson, M.L.: Agreeing to Disagree: Search Engines and Their Public Interfaces. In: Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries, JCDL '07, pp. 309–318 (2007)

[205] McCown, F., Nelson, M.L.: Characterization of Search Engine Caches. In: Proceedings of IS&T Archiving 2007, pp. 48–52 (2007)

[206] McCown, F., Smith, J.A., Nelson, M.L., Bollen, J.: Lazy preservation: reconstructing websites by crawling the crawlers. In: Proceedings of the 8th annual ACM international workshop on Web information and data management, WIDM '06, p. 67 (2006)

[207] Meng, W., Yu, C., Liu, K.L.: Building efficient and effective metasearch engines. ACM Computing Surveys **34**(1), 48–89 (2002)

[208] Mesbah, A., Bozdag, E., van Deursen, A.: Crawling AJAX by Inferring User Interface State Changes. In: Proceedings of the 2008 Eighth International Conference on Web Engineering, ICWE '08, pp. 122–134 (2008)

[209] Miller, E.: An Introduction to the Resource Description Framework. D-Lib Magazine **4**(5) (1998)

[210] Mohr, G., Stack, M., Ranitovic, I., Avery, D., Kimpton, M.: An Introduction to Heritrix An open source archival quality web crawler. In: Workshop Proceedings of the 4th International Web Archiving Workshop (IWAW04), pp. 43–49 (2004)

[211] Monroe, G., French, J., Powell, A.: Obtaining Language Models of Web Collections Using Query-Based Sampling Techniques. Hawaii International Conference on System Sciences **3**, 67b (2002)

[212] Mühleisen, H., Bizer, C.: Web Data Commons - Extracting Structured Data from Two Large Web Corpora. In: Proceedings of 4th Linked Data on the Web Workshop, LDOW2012 (2012)

[213] Najork, M.: The scalable hyperlink store. In: Proceedings of the 20th ACM conference on Hypertext and hypermedia, HT '09, p. 89 (2009)

[214] Najork, M., Wiener, J.L.: Breadth-first crawling yields high-quality pages. In: Proceedings of the 10th international conference on World Wide Web, WWW '01, pp. 114–118. ACM (2001)

[215] Nelson, M.L.: Memento-Datetime is not Last-Modified. URL `http://ws-dl.blogspot.com/2010/11/2010-11-05-memento-datetime-is-not-last.html`

[216] Nelson, M.L.: 2010-11-15: Memento Presentation at UNC; Memento ID (2010). URL `http://ws-dl.blogspot.com/2010/11/2010-11-15-memento-presentation-at-unc.html`

[217] Niu, J.: An Overview of Web Archiving. D-Lib Magazine **18**(3/4) (2012)

[218] Niu, J.: Functionalities of Web Archives. D-Lib Magazine **18**(3/4) (2012)

[219] Ntoulas, A., Cho, J., Olston, C.: What's new on the web? The Evolution of theWeb from a Search Engine Perspective. In: Proceedings of the 13th international conference on World Wide Web, WWW '04, pp. 1–12 (2004)

[220] Oita, M., Senellart, P.: Archiving Data Objects using Web Feeds. In: Proceedings of the 10th International Web Archiving Workshop, IWAW '10 (2010)

[221] Olston, C., Pandey, S.: Recrawl scheduling based on information longevity. In: Proceeding of the 17th international conference on World Wide Web, pp. 437–446. ACM (2008)

[222] Padia, K., AlNoamany, Y., Weigle, M.C.: Visualizing digital collections at Archive-It. In: Proceedings of the 12th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL 12, pp. 15–18 (2012)

[223] Pant, G., Srinivasan, P., Menczer, F.: Crawling the Web. In: M. Levene, A. Poulovassilis (eds.) Web Dynamics: Adapting to Change in Content, Size, Topology and Use, pp. 153–178. Springer (2004)

[224] Park, H.S., Jun, C.H.: A simple and fast algorithm for K-medoids clustering. Expert Systems with Applications **36**(2, Part 2), 3336–3341 (2009)

[225] Paskin, N.: Digital object identifiers. Information Services and Use **22**(2-3), 97–112 (2002)

[226] Pawlik, M., Augsten, N.: RTED: a robust algorithm for the tree edit distance. Proceedings of the VLDB Endowment **5**(4), 334–345 (2011)

[227] Peitz, S., Mansour, S., Peter, J.T., Schmidt, C., Wuebker, J., Huck, M., Freitag, M., Ney, H.: The RWTH Aachen Machine Translation System for WMT 2013. In: Proceedings of Eighth Workshop on Statistical Machine Translation, ACL 2013, pp. 193–199. Sofia, Bulgaria (2013)

[228] Phillips, A., Davis, M.: RFC 5646 - Tags for Identifying Languages (2009). URL `http://tools.ietf.org/html/rfc5646`

[229] Platt, J.C.: AutoAlbum: clustering digital photographs using probabilistic model merging. In: Proceedings of IEEE Workshop on Content-based Access of Image and Video Libraries, pp. 96–100 (2000)

[230] Pop, R., Vasile, G., Masanès, J.: Archiving Web Video. In: Proceedings of the 10th International Web Archiving Workshop, *IWAW '10*, vol. 204 (2010)

[231] Powell, A.L., French, J.C.: Comparing the performance of collection selection algorithms. ACM Transactions on Information Systems **21**(4), 412–456 (2003)

[232] Preibusch, S., Bonneau, J.: The Privacy Landscape: Product Differentiation on Data Collection. In: B. Schneier (ed.) Economics of Information Security and Privacy III SE - 12, pp. 263–283. Springer New York (2013)

[233] Randall, K.H., Stata, R., Wiener, J.L., Wickremesinghe, R.G.: The Link Database: Fast Access to Graphs of the Web. In: Proceedings of Data Compression Conference, DCC '02, pp. 122–131 (2002)

[234] Ras, M., van Bussel, S.: Web Archiving User Survey. Tech. rep., National Library of the Netherlands (Koninklijke Bibliotheek) (2007). URL `http://www.kb.nl/sites/default/files/docs/KB_UserSurvey_Webarchive_EN.pdf`

[235] Rigaux, P.: Understanding HBase The data model (2012). URL `http://internetmemory.org/en/index.php/synapse/understanding_the_hbase_data_model/`

[236] Rosenthal, D.S.H., Carpenter Negulescu, K.: IIPC Future of the Web Workshop Introduction & Overview. Tech. rep., International Internet Preservation Consortium (2012). URL `http://www.netpreserve.org/sites/default/files/resources/OverviewFutureWebWorkshop.pdf`

[237] Rossi, A.: Fixing Broken Links on the Internet (2013). URL `http://blog.archive.org/2013/10/25/fixing-broken-links/`

[238] Rothenberg, J.: Avoiding Technological Quicksand: Finding a Viable Technical Foundation for Digital Preservation. Council on Library and Information Resources (1999)

[239] Sanderson, R.: Memento Tools: Proxy Scripts (2010). URL `http://www.mementoweb.org/tools/proxy/`

[240] Sanderson, R.: Global Web Archive Integration with Memento. In: Proceedings of the 12th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL '12, pp. 379–380. ACM, New York, NY, USA (2012)

[241] Sanderson, R., AlSum, A.: MementoFox 0.9.5 (2010). URL `https://addons.mozilla.org/en-US/firefox/addon/mementofox/`

[242] Sanderson, R., Shankar, H., Ainsworth, S., McCown, F., Adams, S.: Implementing Time Travel for the Web. Code4Lib Journal (13) (2011)

[243] Sanderson, R., Shankar, H., AlSum, A.: Memento Aggregator source code (2010). URL `https://code.google.com/p/memento-server`

[244] Schenkel, R.: Temporal Shingling for Version Identification in Web Archives. In: C. Gurrin, U. Kruschwitz (eds.) Proceedings of the 32nd European Conference on Information Retrieval, ECIR 2010, pp. 508–519 (2010)

[245] Schneider, R., McCown, F.: First steps in archiving the mobile web. In: Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries, JCDL '13, p. 53 (2013)

[246] Sfakianakis, G., Patlakas, I., Ntarmos, N., Triantafillou, P.: Interval indexing and querying on key-value cloud stores. In: Proceedings of 29th IEEE International Conference on Data Engineering, ICDE 2013, pp. 805–816 (2013)

[247] Shankar, H.: Memento Time Travel for Chrome (2013). URL `https://chrome.google.com/webstore/detail/memento-time-travel/jgbfpjledahoajcppakbgilmojkaghgm`

[248] Shelby, Z.: RFC 6690 - Constrained RESTful Environments (CoRE) Link Format (2012). URL `http://tools.ietf.org/html/rfc6690`

[249] Shiozaki, R., Eisenschitz, T.: Role and justification of web archiving by national libraries: A questionnaire survey. Journal of Librarianship and Information Science **41**(2), 90–107 (2009)

[250] Shokouhi, M., Scholer, F., Zobel, J.: Sample Sizes for Query Probing in Uncooperative Distributed Information Retrieval. In: X. Zhou, J. Li, H. Shen, M. Kitsuregawa, Y. Zhang (eds.) Frontiers of WWW Research and Development - APWeb 2006 SE - 7, *Lecture Notes in Computer Science*, vol. 3841, pp. 63–75. Springer Berlin Heidelberg (2006)

[251] Shokouhi, M., Si, L.: Federated Search. Foundations and Trends in Information Retrieval **5**(1), 1–102 (2011)

[252] Si, L., Callan, J.: Modeling search engine effectiveness for federated search. In: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '05, pp. 83–92. ACM Press, New York, New York, USA (2005)

[253] Smith, C.: Content and Context: Delivering coordinated UK web archives to user communities. In: JISC, the DPC and the UK Web Archiving Consortium Workshop Missing links: the enduring web (2009)

[254] Soman, S., Chhajta, A., Bonomo, A., Paepcke, A.: ArcSpread for Analyzing Web Archives. Tech. rep., Stanford InfoLab (2012)

[255] Song, S., JaJa, J.: Archiving Temporal Web Information: Organization of Web Contents for Fast Access and Compact Storage. In: 8th International Web Archiving Workshop. UMIACS TR-2008-8, Aarhus, Denmark. (2008)

[256] Song, S., JaJa, J.: Fast browsing of archived Web contents. In: Proceedings of 8th International Web Archiving Workshop. UMIACS TR-2008-8, Aarhus, Denmark. (2008)

[257] Spaniol, M., Benczúr, A.A., Viharos, Z., Weikum, G.: Big Web Analytics: Toward a Virtual Web Observatory. ERCIM News **2012**(89) (2012)

[258] Spaniol, M., Denev, D., Mazeika, A., Weikum, G., Senellart, P.: Data quality in web archiving. In: Proceedings of the 3rd workshop on Information credibility on the web, WICOW '09, pp. 19–26. ACM (2009)

[259] Spaniol, M., Mazeika, A., Denev, D., Weikum, G.: "Catch me if you can": Visual Analysis of Coherence Defects in Web Archiving. Proceedings of the 9th International Web Archiving Workshop pp. 27–37 (2009)

[260] Spaniol, M., Weikum, G.: Tracking Entities in Web Archives: The LAWA Project. In: Proceedings of the 21st International Conference Companion on World Wide Web, WWW '12 Companion, pp. 287–290. ACM, New York, NY, USA (2012)

[261] Stack, M.: Full Text Searching of Web Archive Collections. In: Proceedings of the 5th International Web Archiving Workshop, IWAW 2005, pp. 27–37 (2005)

[262] Stirling, P., Chevallier, P., Illien, G.: Web Archives for Researchers: Representations, Expectations and Potential Uses. D-Lib Magazine **18**(3/4) (2012)

[263] Stirling, P., Illien, G., Sanz, P., Sepetjan, S.: The state of e-legal deposit in France: looking back at five years of putting new legislation into practice and envisioning the future. In: World Library and Information Congress : 77th IFLA General Conference and Assembly (2011)

[264] Stoev, S.L., Straßer, W.: A case study on interactive exploration and guidance aids for visualizing historical data. In: Proceedings of the conference on Visualization, VIS '01, pp. 485–488 (2001)

[265] Suel, T., Yuan, J.: Compressing the graph structure of the Web. In: Proceedings of Data Compression Conference, DCC 2001, pp. 213–222 (2001)

[266] Tahmasebi, N., Niklas, K., Theuerkauf, T., Risse, T.: Using Word Sense Discrimination on Historic Document Collections. In: Proceedings of the 10th Annual Joint Conference on Digital Libraries, JCDL '10, pp. 89–98. ACM, New York, NY, USA (2010)

[267] Tahmasebi, N., Niklas, K., Zenz, G., Risse, T.: On the applicability of word sense discrimination on 201 years of modern English. International Journal on Digital Libraries **13**(3-4), 135–153 (2013)

[268] Teevan, J., Cutrell, E., Fisher, D., Drucker, S.M., Ramos, G., André, P., Hu, C.: Visual Snippets: Summarizing Web Pages for Search and Revisitation. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '09, pp. 2023–2032. ACM (2009)

[269] Teevan, J., Dumais, S.T., Liebling, D.J., Hughes, R.L.: Changing how people view changes on the web. In: Proceedings of the 22nd annual ACM symposium on User interface software and technology, UIST '09, pp. 237–246 (2009)

[270] Tennison, J.: Hash URIs (2011). URL http://www.jenitennison.com/blog/node/154

[271] Thelwall, M., Vaughan, L.: A fair history of the Web? Examining country balance in the Internet Archive. Library and Information Science Research **26**(2), 162–176 (2004)

[272] Thomas, A., Meyer, E.T., Dougherty, M., van den Heuvel, C., Madsen, C., Wyatt, S.: Researcher Engagement with Web Archives: Challenges and Opportunities for Investment. Tech. rep., Joint Information Systems Committee (JISC) (2010). URL http://ssrn.com/abstract=1715000

[273] Thomas, P., Hawking, D.: Evaluating sampling methods for uncooperative collections. In: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '07, pp. 503–512 (2007)

[274] Traeger, A., Joukov, N., Sipek, J., Zadok, E.: Using Free Web Storage for Data Backup. In: Proceedings of the second ACM workshop on Storage security and survivability, StorageSS '06, pp. 73–78. ACM Press, New York, New York, USA (2006)

[275] Treharne, K., Powers, D.M.W.: Search Engine Result Visualisation: Challenges and Opportunities. In: Proceedings of 13th International Conference on Information Visualisation, pp. 633–638 (2009)

[276] Tsang, M., Morris, N., Balakrishnan, R.: Temporal Thumbnails: rapid visualization of time-based viewing data. In: Proceedings of the working conference on Advanced visual interfaces, AVI '04, pp. 175–178 (2004)

[277] Tweedy, H., McCown, F., Nelson, M.L.: A memento web browser for iOS. In: Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries, JCDL '13, p. 371 (2013)

[278] Van de Sompel, H., Nelson, M.L., Sanderson, R.: RFC 7089 - HTTP framework for time-based access to resource states – Memento (2013). URL http://tools.ietf.org/html/rfc7089

[279] Van de Sompel, H., Nelson, M.L., Sanderson, R., Balakireva, L.L., Ainsworth, S., Shankar, H.: Memento: Time Travel for the Web. Tech. Rep. arXiv:0911.1112 (2009)

[280] Van de Sompel, H., Nelson, M.L., Sanderson, R., Balakireva, L.L., Ainsworth, S., Shankar, H.: Memento: Updated Technical Details (February 2010) (2010). URL http://www.slideshare.net/hvdsomp/memento-updated-technical-details-february-2010

[281] Van de Sompel, H., Nelson, M.L., Sanderson, R., Balakireva, L.L., Ainsworth, S., Shankar, H.: Memento: Updated Technical Details (May 2011) (2011). URL http://www.slideshare.net/hvdsomp/memento-updated-technical-details-may-2011

[282] Van de Sompel, H., Sanderson, R., Nelson, M.L., Balakireva, L.L., Shankar, H., Ainsworth, S., Sompel, H.V.D.: An HTTP-Based Versioning Mechanism for Linked Data. In: Proceedings of the Linked Data on the Web Workshop, LDOW 2010 (2010)

[283] Vlcek, I.: Identification and Archiving of the Czech Web Outside the National Domain. In: Proceedings of 8th International Web Archiving Workshop, IWAW '08 (2008)

[284] Wang, Y., Dylla, M., Ren, Z., Spaniol, M., Weikum, G.: PRAVDA-live: Interactive Knowledge Harvesting. In: Proceedings of the 21st ACM International Conference on Information and Knowledge Management, CIKM '12, pp. 2674–2676. ACM, New York, NY, USA (2012)

[285] Waters, D., Garrett, J.: Preserving Digital Information: Report of the Task Force on Archiving of Digital Information (1996). URL `http://www.clir.org/pubs/abstract//reports/pub63`

[286] Weber, M.S.: Newspapers and the Long-Term Implications of Hyperlinking. Journal of Computer-Mediated Communication **17**(2), 187–201 (2012)

[287] Weikum, G., Ntarmos, N., Spaniol, M., Triantafillou, P., Benczúr, A.A., Kirkpatrick, S., Rigaux, P., Williamson, M.: Longitudinal Analytics on Web Archive Data: It's About Time! In: Proceedings of the Fifth Biennial Conference on Innovative Data Systems Research, CIDR 2011, pp. 199–202 (2011)

[288] White, T.: Hadoop: The Definitive Guide, original edn. O'Reilly Media, Inc. (2012)

[289] Woodruff, A., Faulring, A., Rosenholtz, R., Morrsion, J., Pirolli, P.: Using thumbnails to search the Web. In: Proceedings of the SIGCHI conference on Human factors in computing systems, CHI '01, pp. 198–205 (2001)

[290] Yahya, M., Berberich, K., Ramanath, M., Weikum, G.: On the SPOT: Question Answering over Temporally Enhanced Structured Data. In: F. Diaz, S. Dumais, K. Radinsky, M. de Rijke, M. Shokouhi (eds.) Proceedings of Workshop on Time-aware Information Access, TAIA2013. Dublin, Ireland (2013)

[291] Yan, H., Huang, L., Chen, C., Xie, Z.: A New Data Storage and Service Model of China Web. In: Proceedings of 4th International Web Archiving Workshop, IWAW '04 (2004)

[292] Yi, L., Liu, B., Li, X.: Eliminating Noisy Information in Web Pages for Data Mining. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '03, pp. 296–305. ACM, New York, NY, USA (2003)

[293] Yosef, M.A., Bauer, S., Hoffart, J., Spaniol, M., Weikum, G.: HYENA-live: Fine-Grained Online Entity Type Classification from Natural-language Text. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, pp. 133–138. The Association for Computer Linguistics (2013)

[294] Zenz, G., Tahmasebi, N., Risse, T.: Language Evolution On The Go. In: Proceedings of 3rd International Workshop on Semantic Ambient Media Experience, SAME 2010 (2010)

[295] Zheng, S., Song, R., Wen, J.R., Wu, D.: Joint Optimization of Wrapper Generation and Template Detection. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '07, pp. 894–902. ACM, New York, NY, USA (2007)

[296] Zhuge, H., Liu, J., Feng, L., Sun, X., He, C.: Query routing in a peer-to-peer semantic link network. Computational Intelligence **21**(2), 197–216 (2005)

APPENDIX A

WEB ARCHIVES LIST

In this appendix, we will give a summary about the web archives that we mentioned or used in this dissertation.

- **Internet Archive**
  - URI: `http://www.archive.org`
  - TimeGate: `http://web.archive.org/web/{URI}`
  - TimeMap: `http://web.archive.org/web/timemap/link/{URI}`

  The Internet Archive is a non-profit that was founded on 1996 to build an Internet library. Its purposes include offering permanent access to historical collections that exist in digital format. The Internet Archive has been receiving data donations from Alexa Internet and others. In late 1999, the organization started to grow to include more well-rounded collections. Now the Internet Archive includes texts, audio, moving images, and software as well as archived web pages. The Internet A rchive web interface is called Wayback Machine which is a service to enable users to type URI, and select a date range; then the users can view all the available copies that were preserved by internet archive during this time. Internet archive unveiled a new beta version of the WayBack machine `http://web.archive.org`, it provides a new user interface supported with calendar interface with mini-timeline banner in the top for each page to easily move to another date.

- **Archive It**
  - URI: `http://www.archive-It.org`
  - TimeGate: `http://wayback.archive-it.org/all/{URI}`
  - TimeMap: `http://wayback.archive-it.org/all/timemap/link/{URI}`

  Archive-It is a subscription service that allows institutions to build and preserve collections of born digital content. Subscribers include many types or organizations and individuals, such as: state archives, university libraries, federal institutions, state libraries, non government non profits, museums, historians, and independent researchers. Subscribers can create their own collections (web archives), catalogue and manage it, and select their own harvesting frequency, from daily to annual, or on demand. Collections are full-text searchable. It is hosted at the Internet Archive data center and are accessible to the public with full-text search. These features distinguish Archive-It from Internet Archive.

- **Web Cite**
  - URI: `http://www.webcitation.org`

  - TimeGate: `http://mementoproxy.cs.odu.edu/web/timegate/{URI}`
  - TimeMap: `http://mementoproxy.cs.odu.edu/web/timemap/link/{URI}`

  WebCite, is an on-demand archiving system for webreferences, which can be used by authors, editors, and publishers of scholarly papers and books, to ensure that cited webmaterial will remain available to readers in the future. If cited webreferences in journal articles, books etc. are not archived, future readers may encounter a "404 File Not Found" error when clicking on a cited URL. A WebCite-enhanced reference is a reference which contains - in addition to the original live URL (which can and probably will disappear in the future, or its content may change) - a link to an archived copy of the material, exactly as the citing author saw it when he accessed the cited material.

- **UK Web archive (British Library)**

  - URI: `http://www.webarchive.org.uk`
  - TimeGate: `http://www.webarchive.org.uk/wayback/memento/timegate/{URI}`
  - TimeMap: `http://www.webarchive.org.uk/wayback/list/timemap/link/{URI}`

  The UK Web Archive contains websites that publish research, that reflect the diversity of lives, interests and activities throughout the UK, and demonstrate web innovation. This includes "grey literature" sites: those that carry briefings, reports, policy statements, and other ephemeral but significant forms of information. The archive is free to view, accessed directly from the Web itself and, since archiving began in 2004, has collected thousands of websites.

- **National Archives UK**

  - URI: `http://webarchive.nationalarchives.gov.uk/`
  - TimeGate: `http://webarchive.nationalarchives.gov.uk/timegate/{URI}`
  - TimeMap: `http://webarchive.nationalarchives.gov.uk/timemap/{URI}`

  The National Archives is a government department and an executive agency of the Ministry of Justice. The National Archives is home to millions of documents, files and images that cover 1,000 years of history. The web archive division is responsible for archiving UK governmental websites.

- **Library and Archives Canada - LAC**

  - URI: `http://www.collectionscanada.gc.ca/webarchives/`
  - TimeGate: `http://mementoproxy.cs.odu.edu/can/timegate/{URI}`
  - TimeMap: `http://mementoproxy.cs.odu.edu/can/timemap/link/{URI}`

  The Library and Archives of Canada collects a representative sample of Canadian websites. LAC began in December 2005 with harvesting Canadian Federal Government websites. The user can access the web archive through searching by keyword, URI, and department name.

- **Library of Congress**
  - URI: `http://webarchives.loc.gov/`
  - TimeGate: `http://mementoproxy.cs.odu.edu/loc/timegate/{URI}`
  - TimeMap: `http://mementoproxy.cs.odu.edu/loc/timemap/link/{URI}`

  The Library of Congress Web Archives (LCWA) is composed of collections of archived web sites selected by subject specialists to represent web-based information on a designated topic. It is part of a continuing effort by the Library to evaluate, select, collect, catalog, provide access to, and preserve digital materials for future generations of researchers.

- **Archive.is**
  - URI: `http://archive.is`
  - TimeGate: `http://archive.is/timegate/{URI}`
  - TimeMap: `http://archive.is/timemap/{URI}`

  Archive.is is a user submission web archive based on URI promoted by the regular users. It provides URI-lookup only. Archive.is captures the web page in its original format, it generates also a thumbnail image for the page.

- **Web Archive Singapore**
  - URI: `http://was.nl.sg`
  - TimeGate: `http://mementoproxy.cs.odu.edu/sg/timegate/{URI}`
  - TimeMap: `http://mementoproxy.cs.odu.edu/sg/timemap/link/1/{URI}`

  Web Archive Singapore preserves 1,000 websites that are Singapore-related. It started the crawling in 2006.

- **National Taiwan University Web Archive System - NTUWAS**
  - URI: `http://webarchive.lib.ntu.edu.tw/`
  - TimeGate: `http://mementoproxy.cs.odu.edu/tw/timegate/{URI}`
  - TimeMap: `http://mementoproxy.cs.odu.edu/tw/timemap/link/1/{URI}`

  NTUWAS started in December 2006, it is using HTTRrack as crawling tool. NTUWAS has full-text and URI-lookup search. NTUWAS has a selective scope. NTUWAS does not cover all the Taiwanese websites, but focuses on specific set of websites that have long-term research value.

- **National and University Library of Slovenia**
  - URI: `http://arhiv.nuk.uni-lj.si/`
  - TimeGate: `http://mementoproxy.cs.odu.edu/web/timegate/{URI}`

  - TimeMap: `http://mementoproxy.cs.odu.edu/web/timemap/link/{URI}`

  National and University Library of Slovenia Web Archive started to preserve the Slovenian websites since 2008. It has both full-text and URI-lookup search, in addition to browsing using categories and alphabetic list of websites.

- **Library of Catalonia - PADICAT**

  - URI: `http://www.padi.cat/`
  - TimeGate: `http://mementoproxy.cs.odu.edu/cat/timegate/{URI}`
  - TimeMap: `http://mementoproxy.cs.odu.edu/cat/timemap/link/1/{URI}`

  Starting from July 2006, Library of Catalonia started to preserve the Catalan websites as a part of the preservation the Catalan culture. The web archive is known "Digital Heritage of Catalonia" (PADICAT). It has both full-text and URI-lookup search, in addition to browsing using categories and alphabetic list of websites.

- **National and University Library of Iceland**

  - URI: `http://Vefsafn.is`
  - TimeGate: `http://mementoproxy.cs.odu.edu/ic/timegate/{URI}`
  - TimeMap: `http://mementoproxy.cs.odu.edu/ic/timemap/link/{URI}`

- **Croatian Web Archive**

  - URI: `http://haw.nsk.hr`
  - TimeGate: `http://mementoproxy.cs.odu.edu/cr/timegate/{URI}`
  - TimeMap: `http://mementoproxy.cs.odu.edu/cr/timemap/link/1/{URI}`

  Croatian Web Archive at National and University Library in Zagreb is a collection of selected content downloaded from the Internet and stored on a computer server Libraries. It is designed for downloading and permanent storage of publications from the Internet as part of Croatian cultural heritage. Archived content can be searched via title, URL, keyword and subject areas.

- **WebArchiv - Archive of the Czech Web**

  - URI: `http://webarchiv.cz/`
  - TimeGate: `http://mementoproxy.cs.odu.edu/cz/timegate/{URI}`
  - TimeMap: `http://mementoproxy.cs.odu.edu/cz/timemap/link/1/{URI}`

WebArchiv started in 2000 to preserve the websites within Czech web.

APPENDIX B

SAMPLE SNIPPETS

Listing 2 Taiwan Web Archive Profile in JSON format.

```
{"Profile":{
  "Name":"Taiwan Web Archive",
  "URI":"http://webarchive.lib.ntu.edu.tw",
  "TimeGate":
  "http://mementoproxy.cs.odu.edu/tw/timegate/",
  "Code":"TW",
  "Age":"Tue, 15 Jul 1997 00:00:00 GMT",
  "TLD":[{"tw":0.6},{"cn":0.08},{"hk:0.04},
         {"eg":0.04},{"gov":0.04},{"my":0.04},
         {"jp":0.04},{"kr":0.02}],
  "Language":[{"zh-TW":0.5},{"zh-CN":0.25},
              {"id":0.08},{"ar":0.08}],
  "GrowthRate":[
    {"199707":[4,4]},{"200202":[1,1]},
    {"200607":[30,62]},{"200608":[20,80]},
    {"200609":[5,9]},{"200612":[77,129]},
    ... // other values truncated
    {"201308":[7,94]},{"201309":[2,94]}]
  }
}

//GrowthRate format is: {YYYYMM:[#URI-R,#URI-M]}
```

Listing 3 Memento TimeMap in application/link-format [278].

```
GET /timemap/http://a.example.org HTTP/1.1
Host: arxiv.example.net
Accept: application/link-format;q=1.0

HTTP/1.1 200 OK
Date: Thu, 21 Jan 2010 00:06:50 GMT
Server: Apache
Content-Length: 4883
Content-Type: application/link-format
Connection: close

 <http://a.example.org>;rel="original",
 <http://arxiv.example.net/timemap/http://a.example.org>
   ; rel="self";type="application/link-format"
   ; from="Tue, 20 Jun 2000 18:02:59 GMT"
   ; until="Wed, 09 Apr 2008 20:30:51 GMT",
 <http://arxiv.example.net/timegate/http://a.example.org>
   ; rel="timegate",
 <http://arxiv.example.net/web/20000620180259/http://a.example.org>
   ; rel="first memento";datetime="Tue, 20 Jun 2000 18:02:59 GMT"
   ; license="http://creativecommons.org/publicdomain/zero/1.0/",
 <http://arxiv.example.net/web/20091027204954/http://a.example.org>
    ; rel="last memento";datetime="Tue, 27 Oct 2009 20:49:54 GMT"
    ; license="http://creativecommons.org/publicdomain/zero/1.0/",
 <http://arxiv.example.net/web/20000621011731/http://a.example.org>
   ; rel="memento";datetime="Wed, 21 Jun 2000 01:17:31 GMT"
   ; license="http://creativecommons.org/publicdomain/zero/1.0/",
 <http://arxiv.example.net/web/20000621044156/http://a.example.org>
   ; rel="memento";datetime="Wed, 21 Jun 2000 04:41:56 GMT"
   ; license="http://creativecommons.org/publicdomain/zero/1.0/",
```

Listing 4 Link Structure partial response session.

```
> curl "http://localhost:8080/LinkService/linkQuery?uri=vancouver2010.com
    "

<?xml version="1.0"?><rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf
    -syntax-ns#"
xmlns:twg="http://www.mementoweb.org/TemporalWebGraph/">
<rdf:Description rdf:about="vancouver2010.com">
 <twg:hasOutlinks rdf:parseType="Collection">
   <rdf:Description rdf:about="paralympic-games)/news">
    <twg:type>href</twg:type> <twg:text>News</twg:text>
    <twg:timestamp>  <rdf:Bag>
     <rdf:li>20091103011307</rdf:li><rdf:li>20100130003005</rdf:li> ...
    </rdf:Bag> </twg:timestamp> </rdf:Description>
   <rdf:Description rdf:about="olympic-cross-country-skiing)/">
    <twg:type>href</twg:type> <twg:text>Cross-Country Skiing</twg:text>
    <twg:timestamp> <rdf:Bag>
     <rdf:li>20091110011557</rdf:li> <rdf:li>20100227081100</rdf:li>  ...
    </rdf:Bag> </twg:timestamp> </rdf:Description>
.....
 </twg:hasOutlinks>
 <twg:hasInlinks rdf:parseType="Collection">
   <rdf:Description rdf:about="http://vancouver2010.teamgb.com/gallery/
       gillian-cooke/">
    <twg:type>href</twg:type> <twg:text>Official Vancouver Games site</
       twg:text>
    <twg:timestamp> <rdf:Bag>
     <rdf:li>20100217101229</rdf:li>
    </rdf:Bag> </twg:timestamp> </rdf:Description>
     <rdf:Description rdf:about="http://www.swissolympic.ch/olympiablog/?
         tag=/verletzung">
      <twg:type>href</twg:type> <twg:text>VANOC 2010</twg:text>
     <twg:timestamp> <rdf:Bag>
      <rdf:li>20100220104902</rdf:li>
     </rdf:Bag> </twg:timestamp> </rdf:Description>
....
 </twg:hasInlinks>
</rdf:Description></rdf:RDF>
```

Listing 5 WAT file record for vancouver2010.com.

```
WARC/1.0
WARC-Type: metadata
WARC-Target-URI: http://www.vancouver2010.com/
WARC-Date: 2009-11-03T01:13:07Z
WARC-Record-ID: <urn:uuid:9f1ad7bb-585d-4312-a523-79720573b71f>
WARC-Refers-To: <urn:uuid:1fb143f9-5573-43f5-899e-0cac2cbed605>
Content-Type: application/json
Content-Length: 35238

{
   "Envelope":{
   "Format":"WARC",
   "WARC-Header-Length":"344",
   "Block-Digest":"sha1:3CYDPYZPMDZTCCFJZLXBCSHYMWRW4RDC",
   "Actual-Content-Length":"64203",
   "WARC-Header-Metadata":{
   "WARC-Type":"response",
   "WARC-Date":"2009-11-03T01:13:07Z",
   "Content-Length":"64203",
   "WARC-Record-ID":"<urn:uuid:1fb143f9-5573-43f5-899e-0cac2cbed605>",
   "WARC-IP-Address":"72.246.105.28",
   "WARC-Payload-Digest":"sha1:XY4DS2MESSVPUYZOJWHKUVF3UOMR5CMA",
   "WARC-Target-URI":"http://www.vancouver2010.com/",
   "Content-Type":"application/http; msgtype=response"
   },
   "Payload-Metadata":{
   "Trailing-Slop-Length":"4",
   "Actual-Content-Type":"application/http; msgtype=response",
   "HTTP-Response-Metadata":{
   "Headers":{
      "Date":"Tue, 03 Nov 2009 01:13:07 GMT",
      "Expires":"Tue, 03 Nov 2009 01:14:27 GMT",
      "Last-Modified":"Tue, 03 Nov 2009 00:17:29 GMT",
      "X-CNode":"v35",
      "X-N":"S",
      "Connection":"close",
      "Content-Type":"text/html; charset=UTF-8",
      "Server":"Apache",
      "Cache-Control":"max-age=80"
   },
   "Headers-Length":"266",
   "Entity-Length":"63937",
   "Entity-Trailing-Slop-Bytes":"0",
```

```
    "Response-Message":{
       "Status":"200",
       "Version":"HTTP/1.0",
       "Reason":"OK"
    },
    "HTML-Metadata":{
       "Links":[
       {
       "alt":"Vancouver 2010 Olympic Games",
       "path":"IMG@/src",
       "url":"/gfx/00/07/33/lg_vancouver2010_16d-TX.gif"
       },
       {
       "title":"Vancouver 2010 Olympic Games",
       "path":"A@/href",
       "url":"/"
       },
       {
       "path":"FORM@/action",
       "method":"get",
       "url":"/search/index_cf-XG.html"
       },
       ...
       ],
       "Metas":[
       {
       "content":"IE=7",
       "http-equiv":"X-UA-Compatible"
       },
       {
       "content":"D9A144C1CA668AC1E22C9046D32D4050",
       "name":"msvalidate.01"
       },
       {
       "content":"en",
       "name":"language"
       },
       {
       "content":"2010 Vancouver Olympic Games Medals Results Schedule
          Sports : Vancouver 2010 Winter Olympics",
       "name":"title"
       },
       {
```

```
        "content":"Official source of Olympic Games tickets, merchandise,
            results, medals, schedules, athletes, teams, news and photos
            for the Vancouver 2010 Olympics",
88      "name":"description"
        },
        {
        "content":"Official source of Olympic Games tickets, merchandise,
            results, medals, schedules, athletes, teams, news and photos
            for the Vancouver 2010 Olympics",
        "name":"abstract"
93      }
        ],
        "Title":"2010 Vancouver Olympic Games Medals Results Schedule
            Sports : Vancouver 2010 Winter Olympics"
        }
    },
98  "Entity-Digest":"sha1:XY4DS2MESSVPUYZOJWHKUVF3UOMR5CMA"
    }
    }
    },
    "Container":{
103 "Compressed":true,
    "Gzip-Metadata":{
    "Footer-Length":"8",
    "Deflate-Length":"12936",
    "Header-Length":"10",
108 "Inflated-CRC":"528011319",
    "Inflated-Length":"64551"
    },
    "Offset":"226620",
    "Filename":"ARCHIVEIT-1645-WEEKLY-WEWUKW-20091103011302-00000-
        crawling01.us.archive.org.warc.gz"
113 }
}
```

APPENDIX C

WEB ARCHIVING SERVICES CATALOG

In this appendix, we will give a high-level overview of all the services that could be useful to access the web archive. For each service, we will give a description about the following:

1. *Input*: It lists the input parameters that could be passed to the service. The service could receive different type of parameters. We used the following notataions to describe the input parameters: "," denotes OR (one is required) and "+" denotes AND (both are required).
2. *Output*: It describes the expected output from the service. As possible, the output will be supported by mathematical formula to illustrate the output.
3. *Challenges*: It gives a high level idea about the challenges or the research questions to implement this service.
4. *Service Level*: It gives the service level which may be archive, URI, metadata, or content level.

C.1 WEB PAGE SNAPSHOT REPLAY

The main goal of the web archiving delivery method is the ability to view the web page as it has appeared in the past. $URI - M(t_i)$ is a representation of the URI-R at time t_i. The web archive should be able to replay the archived snapshot of the web page.

- *Input*: $URI - R$ + Datetime, or $URI - M$.
- *Output*: The page representation of the $URI - R$ as it appeared in the specified Datetime.
- *Challenges*: URI rewriting, javascript, and embedded resources.
- *Service Level*: Content level with GUI interface.

Fig. 74 Web Page Snapshot Replay of www.digitalpreservation.com at datetime Sep 16, 2010 from Internet Archive.

C.2 VARIOUS ARCHIVED LIST REPRESENTATION

The web archive may preserve one or more snapshot for the same URI in different date/time. A TimeMap [7] for an original resource ($URI-R$) is a resource from which a list of URIs of Mementos ($URI-M$) of the Original Resource is available. A TimeMap [7] for an original resource ($URI-R$) is a resource from which a list of URIs of Mementos ($URI-M$) of the Original Resource is available.

API version of the timeline is required for different applications. For example, Wayback Machine provides XML interface, also Memento protocol provides two types for TimeMap (Application-link [248] and XML-RDF). The required API should be able to format the TimeMap in different formats such as: JSON[1]. This API is already supported by Memento protocol.

- *Input*: $URI-R$, $URI-R$ + Date interval.
- *Output*: A list of the mementos for this URI.

  $URI-T = \{< URI-MT_0), T_0 >, < URI-M(T_1), T_1 >, \ldots, < URI-M(T_n), T_n >\}$
- *Service Level*: URI level with API interface.

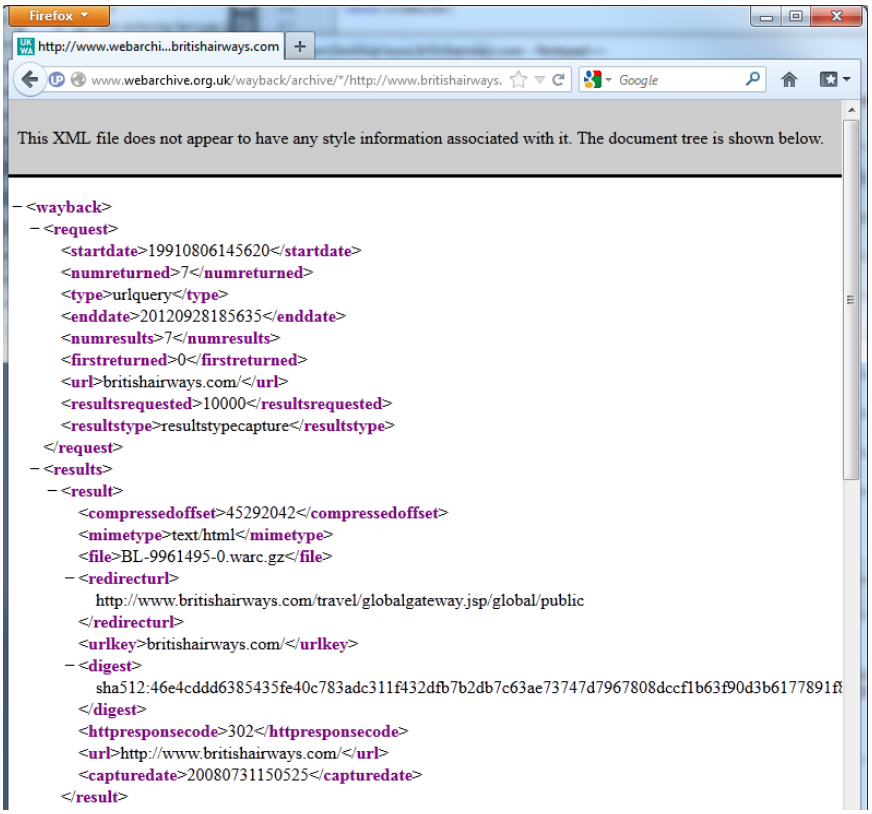


Fig. 75 TimeMap in XML interface for www.BritishAirways.com from the UK Web Archive.

[1] `http://www.json.org`

C.3 TIMELINE VIEW

The visualization of the TimeMap is the main GUI screen for the web archives. A page that contains a list of the available mementos for specific URI-G exists on all archives. WayBack Machines provides two kinds of views, the open source version enables the developer to customize the view for each archive.

- *Input*: $URI - R$, $URI - R$ + Date interval.
- *Output*: A visualization for the available mementos for the requested URI, each one is defined by its datetime.
- *Challenges*: Each one of these mementos have been captured in different time, and also may be preserved in a separate locations. So the service should be able to locate all the mementos for specific URIs.
- *Service Level*: URI level with GUI interface.

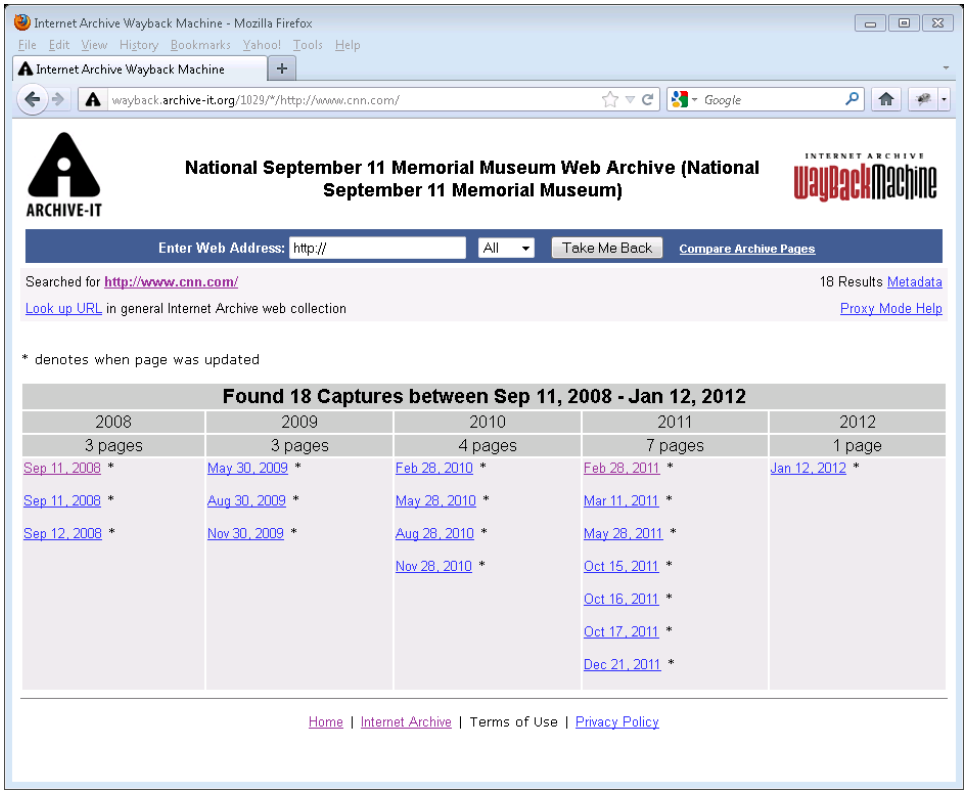


Fig. 76 Timeline view for www.cnn.com from Archive-It Sep 11 collection.

C.4 THUMBNAIL VIEW

TimeMap views define the memento by its capture date/time. Additional to the date/time for each memento in the timeline, the thumbnail view will provide a thumbnail for each memento. This service enables the user to get better insight about the timeline[2].

- *Input*: $URI - R$, $URI - R$ + Date interval.
- *Output*: A list of the available mementos for the requested URI, each one is defined by its thumbnail.

  $URI - T(Thumbnail) = \{< URI - M(T_0), T_0, Thumbnail_0 >, < URI - M(T_1), T_1, Thumbnail_1 >,$

  $\ldots, < URI - M(T_n), T_n, Thumbnail_n >\}$
- *Challenges*: There is a trade-off between building the thumbnail in the real time (which is time-consuming) and pre-building of the thumbnail (which needs additional storage). Also, the trade-off between representing the thumbnail by URI or by embedded binary data.
- *Service Level*: Metadata level with GUI interface.

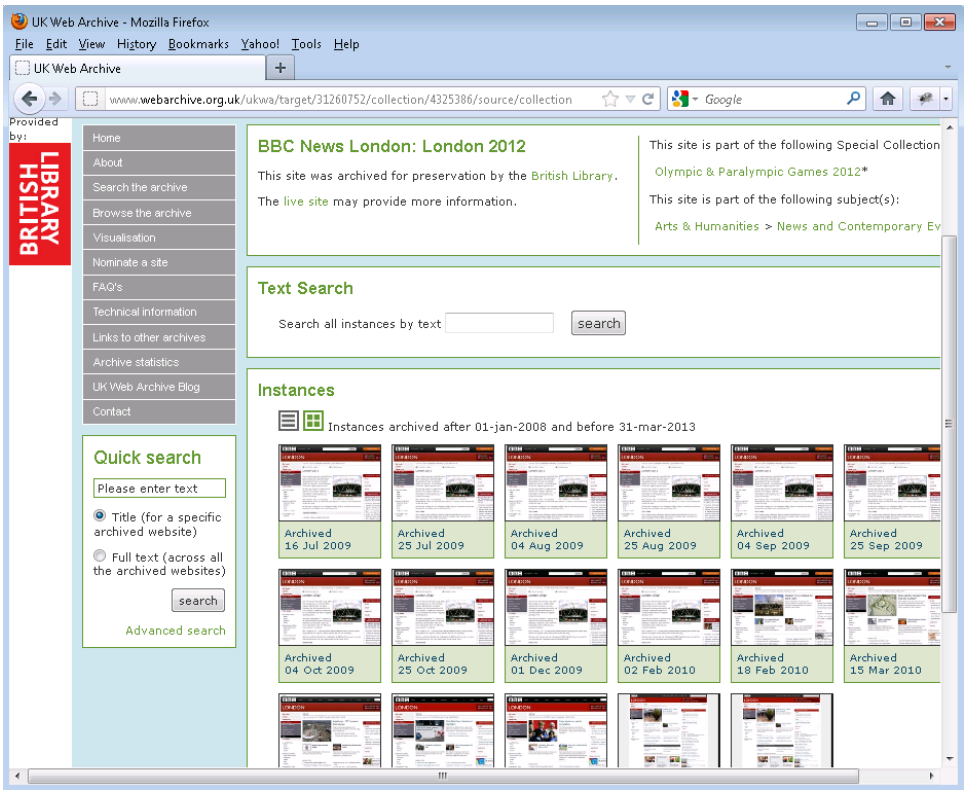


Fig. 77 Thumbnail view for BBC News London: London 2012 from UK Web Archive.

[2] `http://www.webarchive.org.uk/ukwa/target/31260752/collection/4325386/source/collection`

C.5 STATISTICS

Additional to the complete list of available mementos, the users may be interested in getting statistics about the TimeMap itself. So we could consider statistics view as a compressed view or metadata about the TimeMap. For example, the number of captures per URI, the distribution of mementos through the time, date range, etc.

Sometime the users visit the archive not to get content but to get statistics about the content. For example, the user may be interested in retrieving the number of available archived copies more than the list itself. The date range (first observations and the last observations) may be important information. The Web Archive may provide this kind of API to facilitate the user response, additional to save the Web archive resources from doing unuseful computation.

- *Input*: $URI - R$.
- *Output*: The number of captures, the first memento and the last memento date, etc.
- *Challenges*: This statics data could be pre-processed to be ready for the user request, instead of calculating it on fly.
- *Service Level*: URI level with API interface.

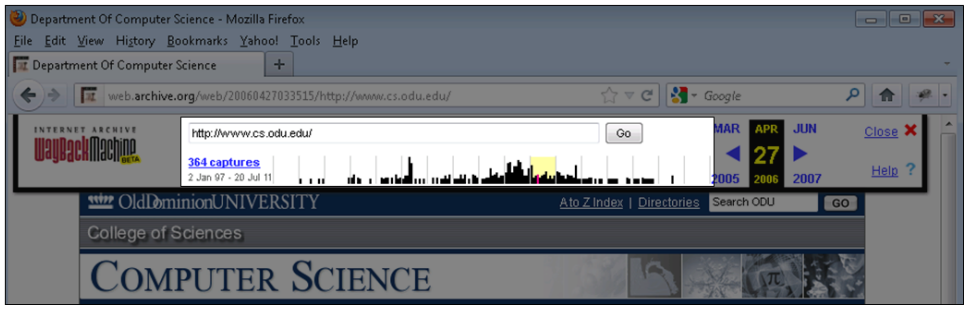


Fig. 78 Wayback Machine statistics about the TimeMap of www.cs.odu.edu.

C.6 TITLE VIEW

The title of the web page could be used as an indication of the content of the page [173]. TimeMap could be built based on the combination of the web page titles through the time. The title gives the user an idea about the content of the page, especially the page title is changed through the time, so retrieving the titles is useful to build new types of user interface.

- *Input*: $URI - R$ + Date interval, $URI - M$, or URI-T.
- *Output*: A list of the available mementos, each one is defined by its datetime and title.
  $URI - T(Title) = \{< URI - M(T_0), T_0, Title_0 >, < URI - M(T_1), T_1, Title_1 >,$
  $\cdots < URI - M(T_n), T_n, Title_n >\}$
- *Challenges*: There are two options to extract the title for the memento: first, extracting all the titles and keeping it as a metadata about the memento. Second, extracting the title from the HTML content on the real time.
- *Service Level*: Metadata level with GUI interface..

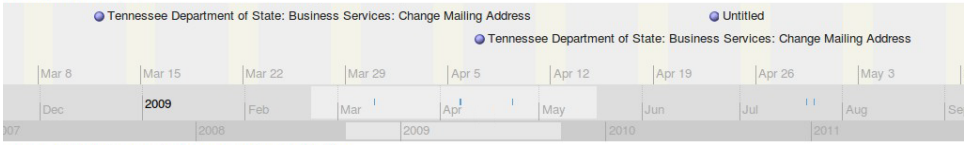


Fig. 79 Title View Prototype.

C.7 PUBLISHING THE RECENT ARCHIVED WEB SITES

The advertising of the available or new arrival content becomes a popular trends in the Web. News websites used Atom/RSS feeds, twitter account, or subscriber emails to tell the audience with its new stuff. Web archive should be able to publish its new archived/accessible mementos. The web archive should allow the users to select their favorite channels. For example, web archive should have: levels (e.g., New $URI-M$, new $URI-R$, or new website that is discovered for the first time), topics (e.g., News, sports), format (e.g., Pdf, videos), and collections. UK Web archive[3] used Atom/RSS feeds inform the users with the new content.

- *Output*: New discovered $URI-M$, $URI-R$, hostnames, or TLD.
- *Challenges*: For the large web archives, this service may be a challenge due to the large number of websites that are getting public.
- *Service Level*: Archive with API interface.

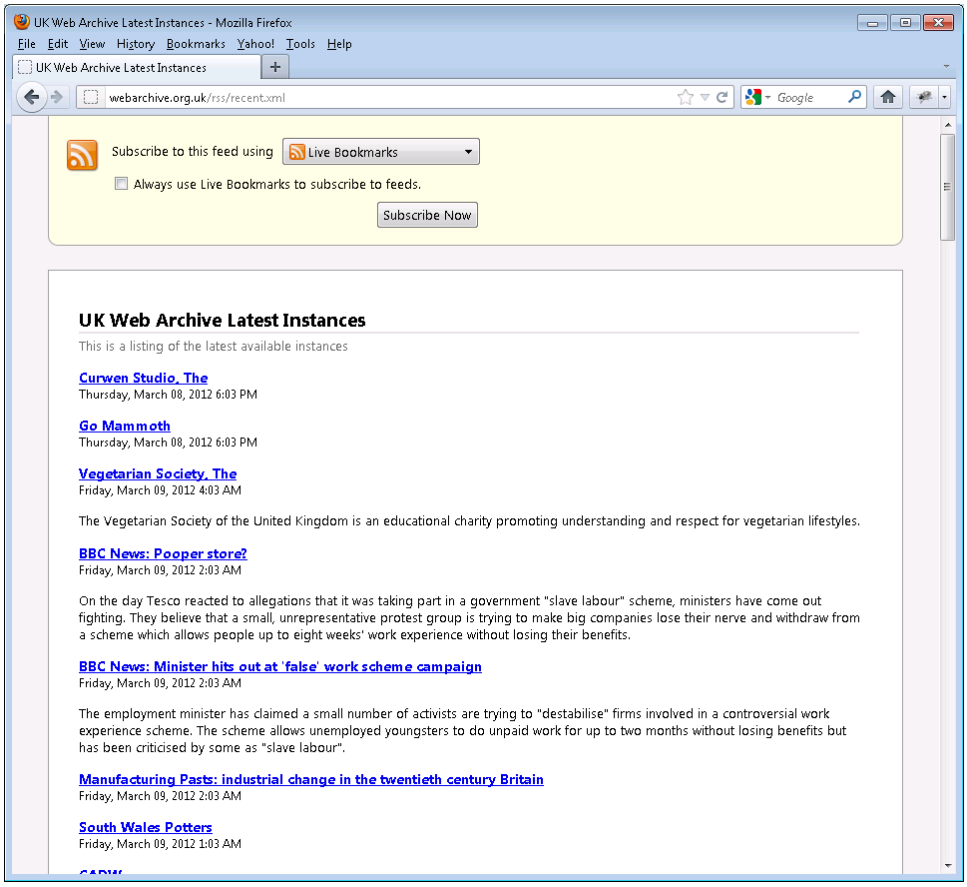


Fig. 80 UK Web Archive Latest Instances in RSS feed format.

[3] `http://webarchive.org.uk/rss/recent.xml`

C.8 DOM REPRESENTATION

The HTML representation for a memento may not be suitable for post-processing, specially for the mementos comparison. The web archive could represent the memento as a DOM tree. The service could be extended by adding additional IDs for the DOM elements to facilitate the comparison/matching services.

- *Input*: $URI - R$ + Datetime, or $URI - M$.
- *Output*: Serialized form of mementos HTML content that have been rewritten to build a unified DOM tree with the same IDs.
- *Challenges*: The change of the page content may be in the page structure which will make it is difficult for the archive to generate the DOM with the same IDs.
- *Service Level*: Content level with API interface.

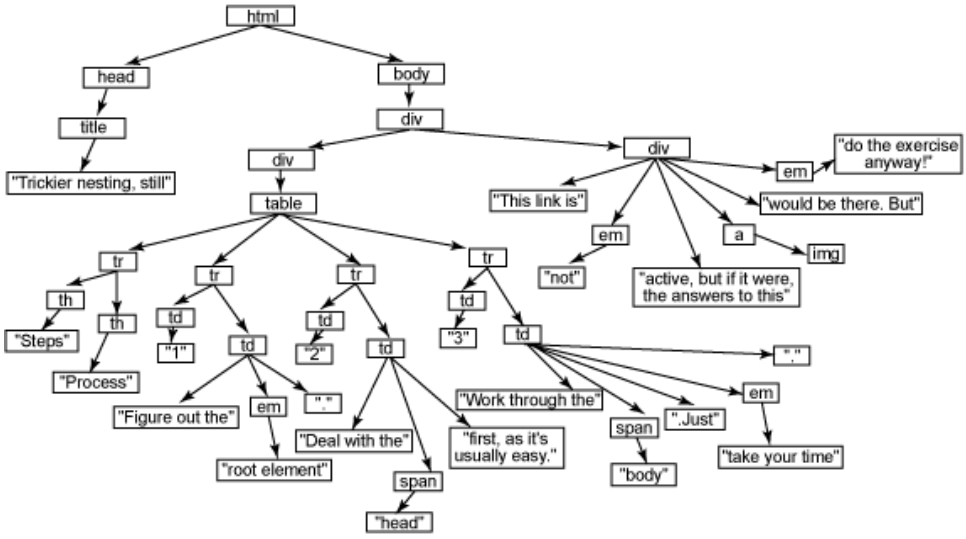


Fig. 81 DOM Tree representation of a web page.

C.9 DIFFERENCE HIGHLIGHTING

Instead of displaying the web page as it appeared on the past, the web archive could highlight the differences between two snapshots of the same URI. This service could take benefits from the DOM representation services. The proposed service has GUI interface only, but it could be extended to serialize the differences between two mementos or for a complete TimeMap.

- *Input*: $URI - M$, or URI-T.
- *Output*: A list of differences between both mementos. Each difference records should contain the operation (addition, deletion), the line number, and the changed text.
- *Challenges*: The change in the page may be in content or in the structure. Content difference could be easily detected based on comparing the DOM elements. Structure comparison may be harder.
- *Service Level*: Content with GUI interface.

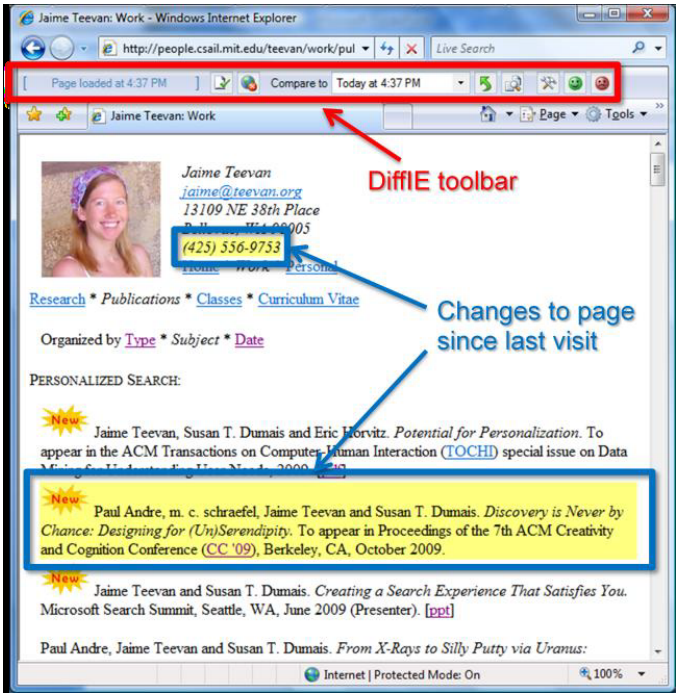


Fig. 82 Difference highlight between two snapshots using DiffIE toolbar (figure is taken from [269]).

C.10 PAGE REPLAY

The web archive could display the mementos in sequence as a slide show. This kind of display gives the user the ability to notice the changes of the website through the time.

- *Input*: $URI - R$, or URI-T.
- *Output*: A sequence of available mementos that could be displayed sequentially.
- *Challenges*: Accessing and rendering a lot of pages sequentially may be processing consuming task.
- *Service Level*: Content with GUI interface.

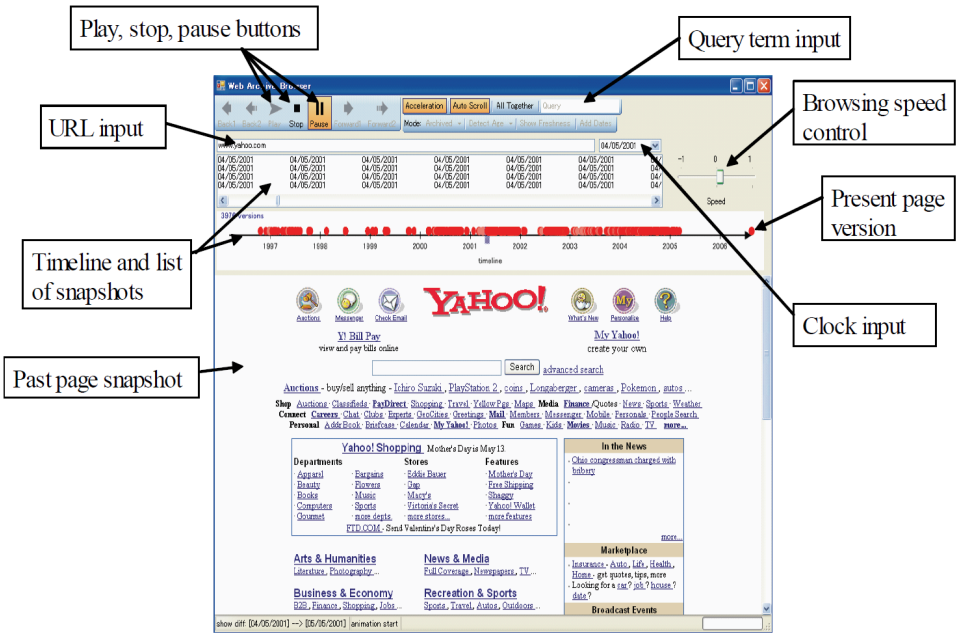


Fig. 83 Page replay (figure is taken from [156]).

C.11 TAGCLOUDS VIEW

TagClouds [188] provides an overview of the most frequent terms in a document or a set of documents. TagClouds could be used to give an overview about a single memento, set of mementos, TimeMap or even a complete collection. The TagClouds could be an easy way to compare different collections.

- *Input*: $URI - R$, Site, or CollectionId.
- *Output*: Term frequency of the requested document(s).
- *Challenges*: Term frequency algorithms should be normalize to take the mementos density (number of mementos per time unit) to avoid high density mementos to squeeze the Tagclouds frequency.
- *Service Level*: Content with API and GUI interfaces.

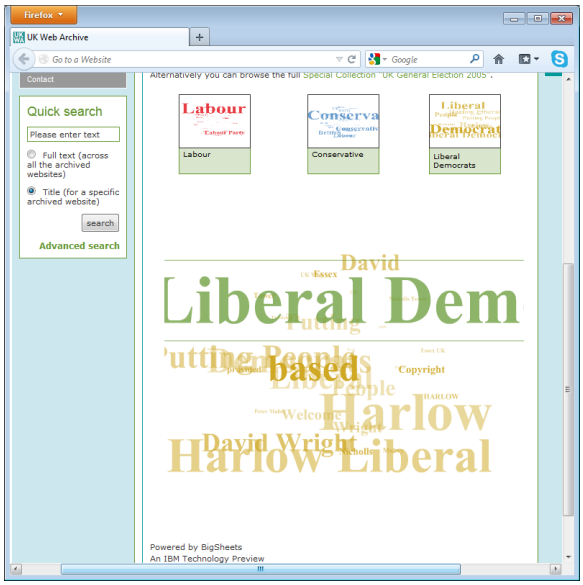


Fig. 84 UK Web Archive Tag cloud view (http://www.webarchive.org.uk/ukwa/cloud).

C.12 PAGE COMPLETENESS DEGREE

Currently, there is no single web archive that evaluates the quality of the web page from the user perspective. The completeness degree is the percentage of the embedded resources that could be retrieved successfully. The embedded resources include images, style sheets, javascript files, etc. The proposed solution has GUI interface, it could be extended to communicate this information to the user through headers, or as a field in the TimeMap.

- *Input*: $URI - M$.
- *Output*: The completeness degree of the memento (between 0-1).

  $$Completeness(\text{URI-M}) = \frac{\text{No of embedded resources that returned 200}}{\text{The total number of embedded resources in the page}}$$

- *Challenges*: The completeness degree could be calculated on the real time by using the preserved HTTP status for the embedded resources.
- *Service Level*: Content level with GUI interface.

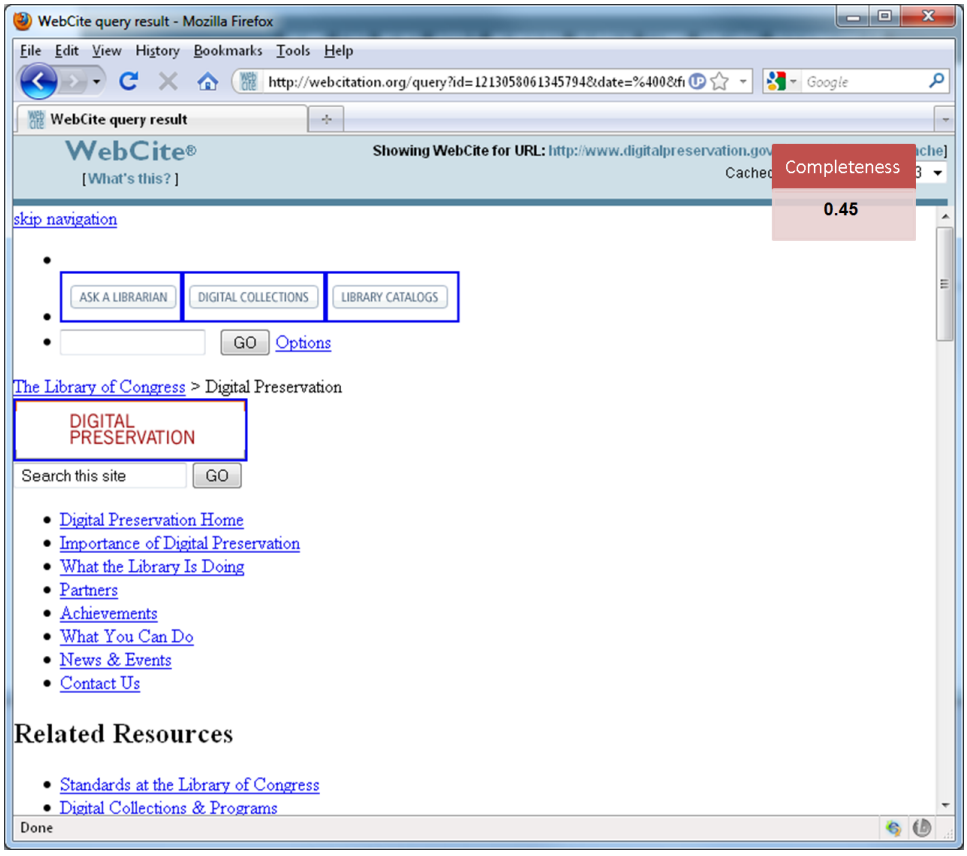


Fig. 85 Memento with 40% completeness.

C.13 PAGE EMBEDDED RESOURCES TIME-SPAN

The coherence between the memento could be calculated by computing the time-span of the embedded resources that compose the memento. The embedded resources that appeared in the archived web page may not be captured in the same time as the actual text. For example, Wayback Machine tries to get the nearest embedded resources for the memento datetime. It may result in a wide range of datetimes for the embedded resources. In this service, an indicator for the user will appear to give him an idea about the range of the embedded resources retrieval in the displayed memento.

- *Input*: $URI - M$.
- *Output*: The date range for the first and the last embedded resource datetime.
- *Service Level*: Content level with GUI interface.

Fig. 86 Memento with an indicator of the timespan of the embedded resources.

C.14 PUSH CONTENT TO ARCHIVE

Few web archives provide the facility to instant archiving to specific URI (such as: Webcitation.org[4]) but the ability to include your website in the general public archiving has a long process [5]. Moreover, some specialized Web Archives did not provide this facility, others required approval for the Website before including [2].

The "Push Content" service will enable the partner to contribute the actual content to the web archive, not only the URI, but the complete crawled data. This service could be used in different ways. The transactional archive content could be pushed to a public web archive to be accessible to the user. Personal digital archiving tools, such as WARCreate [168], preserved the personal web pages in WARC format. Some people may be interested in sharing their content in the public web archives. Also, the site administrator could use this interface to push the content for the sunset websites before removing them from the web.

- *Challenges*: Tdhis automatic feature could affect the Web Archive corpus if it is used to feed the Web Archive with web spam or any malicious pages.
- *Service Level*: Content with API interface.

[4] `http://www.webcitation.org/archive`

C.15 EXPORT CONTENT IN WARC FORMAT

Usually, web archives do not permit to other researchers to use their infrastructure to run their own research (e.g., data mining experiments use infrastructure resources intensively). On the other hand, the researchers may find constraints to build and run their own archives. The proposed solution is to enable web archives to populate (or donate) their content to other environments to run their own experiments. This kind of API is different from the regular interface which enables the users to retrieve the required archived page, because web archive used to do some processes before delivering the archived web page to the users (e.g., URIs rewritten and adding web archive banner). The proposed APIs will populate the raw content (on the ARC or WARC format for Heritix-based crawlers) to the other parties.

- *Input*: $URI - R$, Set or URIs, or DateInterval.
- *Output*: WARC file that contains the archived web pages for the requested URIs.
- *Challenges*: The challenges in this API is closely integrated with the underlying archive infrastructure. For Heritix-based archives, the ARC and WARC files are saved based on the crawling session, so the result of asking for the content for specific URI in all available datetimes will be a set of ARC/WARC files (each of them is around 100M) with the CDX index files. Also, there are some legal issues related to the copyrights of the websites themselves.
- *Service Level*: Content with API interface.

C.16 DOWNLOAD HTML CONTENT

This service will enable the users to download a specific memento or site at specific datetime. It is useful in the case of restoring the website from the web archive in the absence of backup. This service is different from "Export content in WARC format" in the ability to retrieve the website in its HTML original format, not in WARC format which is hard to be read manually. Figure 87 shows an example of Warrick tool[5] [84, 206] that reconstructs a web site by collecting the snapshots from web archives.

- *Input*: $URI-M$, $URI-R$, or Site.
- *Output*: A set of web pages in HTML format and resources in its original format that reflect the original structure of the website/URI.
- *Challenges*: The web archive may face legal issues due to the copyrights that may prevent the web archive from reproducing the content without approval.
- *Service Level*: Content level with API interface.

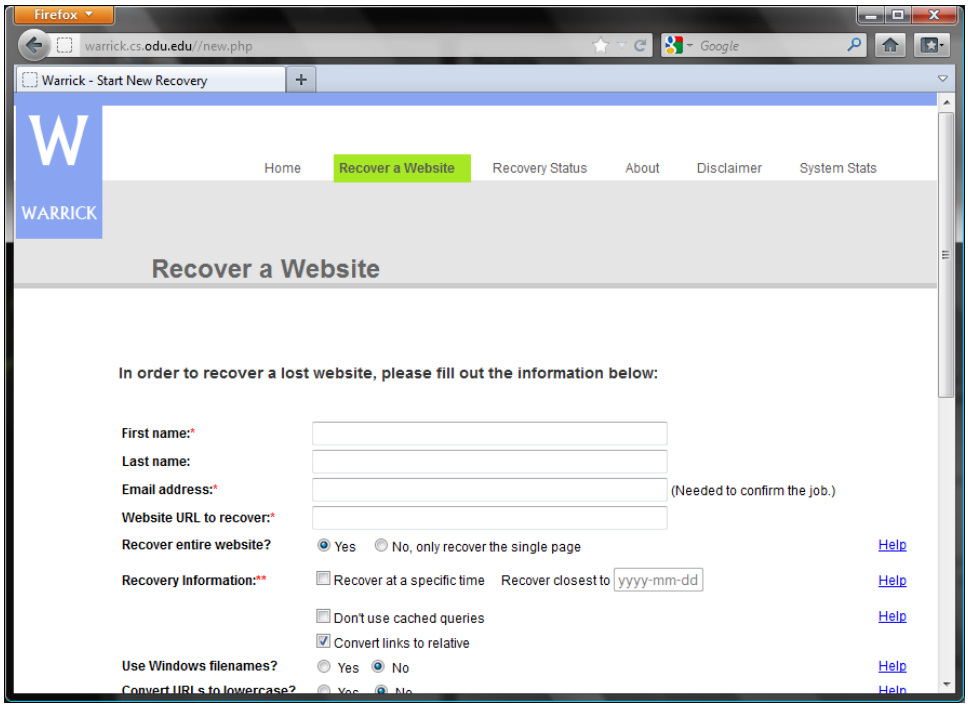


Fig. 87 Warrick tool.

[5] `http://warrick.cs.odu.edu//new.php`

C.17 WEB ARCHIVE FULL-TEXT SEARCH

Archive-It and the British Library are two web archives that provide the full-text search. The limitation of having full-text search for the web archive exists in the infrastructure limitation as the indexing and the ranking tasks are very intensive operations in time and space [261]. In this service, you can search the entire past web for specific term or sentence, which will be easier than remembering a specific URI with a specific datetime. Figure 88 shows an example of search results from Library and Archives Canada[6].

- *Input*: QueryString + (Domains or CollectionId) + Date interval.
- *Output*: List of the documents that are relevant to the query.
- *Challenges*: The time dimension in the search requires a change in the ranking algorithms. Also, the results list interface and the search tools should be updated to facilitate the users digging into the past web.
- *Service Level*: Content level with GUI and API interfaces.

Fig. 88 Web Archive Full-text Search for the query "linux" from Library and Archives Canada.

[6] `http://www.collectionscanada.gc.ca/webarchives/search/results/index-e.html?q=linux`

VITA

Ahmed AlSum
Department of Computer Science
Old Dominion University
Norfolk, VA 23529

EDUCATION

Ph.D.	Computer Science, Old Dominion University, 2014
M.S.	Computer Science, Arab Academy for Science and Technology, 2009
B.S.	Computer Science, Mansoura University, 2003

EMPLOYMENT

02/2014 - Present	Web Archiving Engineer, Stanford University.	Stanford CA, USA.
01/2012 - 04/2012	Software Engineer Intern, Internet Archive.	San Francisco CA, USA.
08/2009 - 02/2014	Research Assistant, Old Dominion University.	Norfolk VA, USA.
03/2006 - 07/2009	Staff Software Engineer, IBM.	Cairo, Egypt.
09/2005 - 02/2006	Software Developer, ITS.	Cairo, Egypt.
02/2004 - 08/2005	Teaching Assistant, Mansoura University.	Mansoura, Egypt.

PUBLICATIONS

A complete list is available at `http://www.aalsum.com/pubs/`.

Typeset using LaTeX.