On Tool Building and Evaluation of the Archived Web

Mat Kelly

Old Dominion University

Web Science & Digital Libraries Research Group

Department of Computer Science

Norfolk, Virginia USA

mkelly@cs.odu.edu

OLD DOMINION UNIVERSITY

Seminar, Penn State University

February 13, 2019

WSDL Old Dominion University

Who I Am

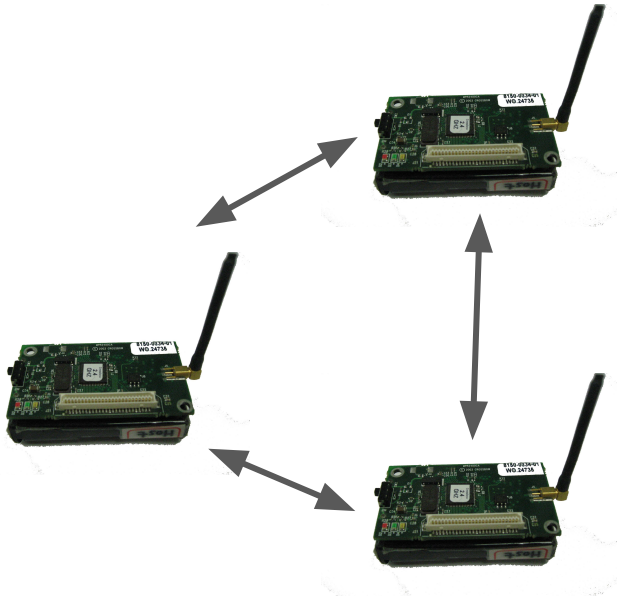
- PhD Candidate (ADB) of Computer Science
- Defending Dissertation in 2019
- Floridian moving progressively Northward

My Research Topic

- Personal, Private, and Public Web archiving
- Technical perspective involving standards
- Tool Builder to support research
  - Often with a solution seeking a problem

The Origin Topic

- Started off researching wireless sensor networks
- Focus: distributed emergency detection
- Exploratory NesC trilateration implementation



https://github.com/machawk1/alert-codebase

Shift Topics & Labs with My PhD Advisor

(Dr. Michele C. Weigle)

Wireless Sensor Networks

Networking (iNeTs) Research Group

Old Dominion University

Web Science and Digital Libraries (WS-DL) Research Group

Old Dominion University

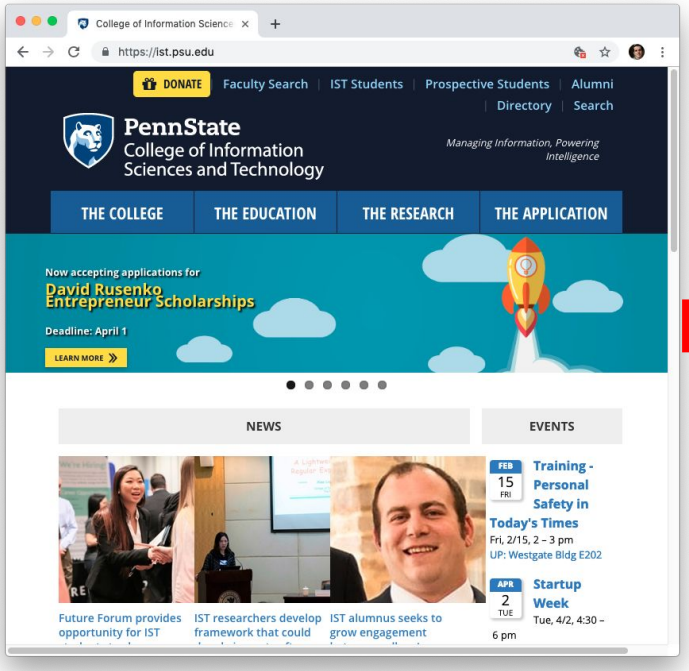
Web Archiving

Web Archives? Like Internet Archive?

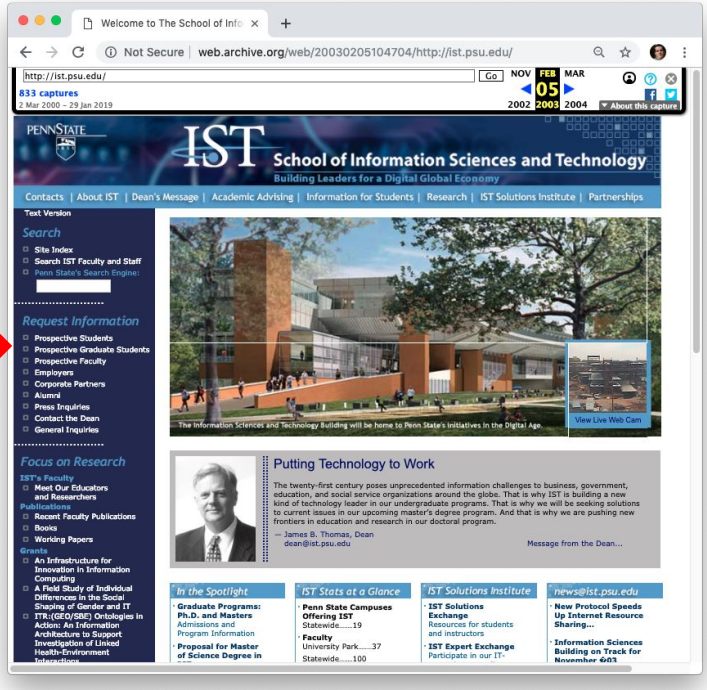
- Saving pages on the Web of today
  - For exploration and research later
- The Archived Web: A Culturally significant resource
- The Internet Archive (IA) started saving the Web in 1996

But there are **other institutional, public archiving efforts** beyond IA

Digital History on the Web

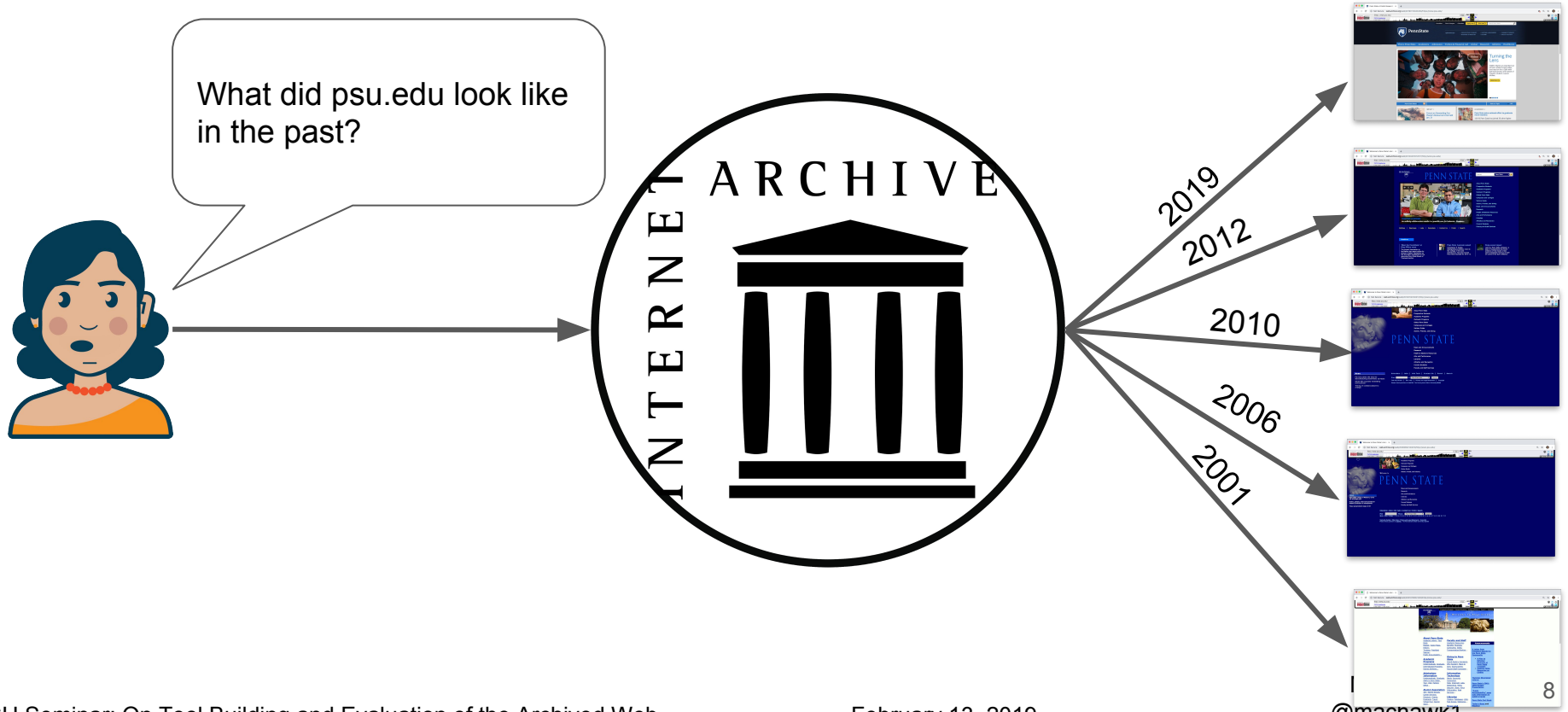


Now



February 2003

PSU.edu of the past

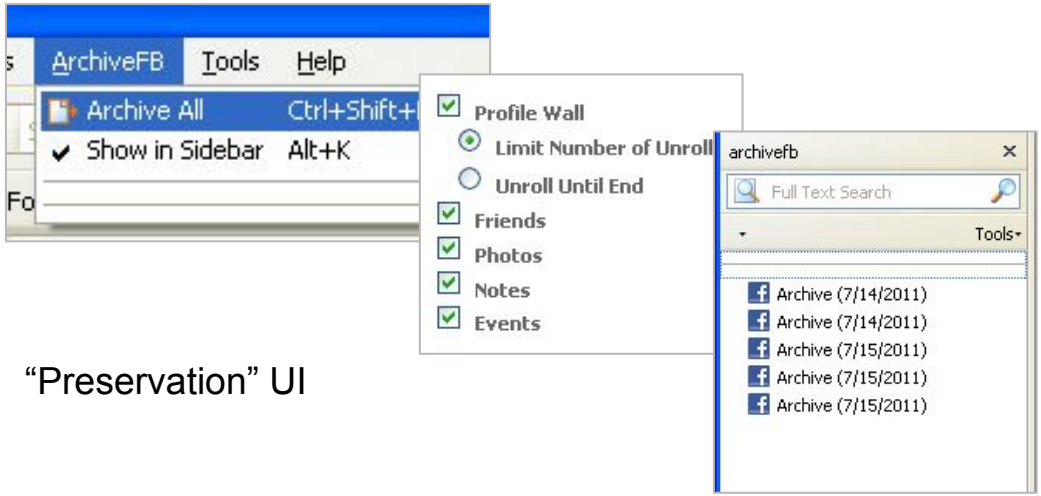


Multiple archival efforts (3 of many)

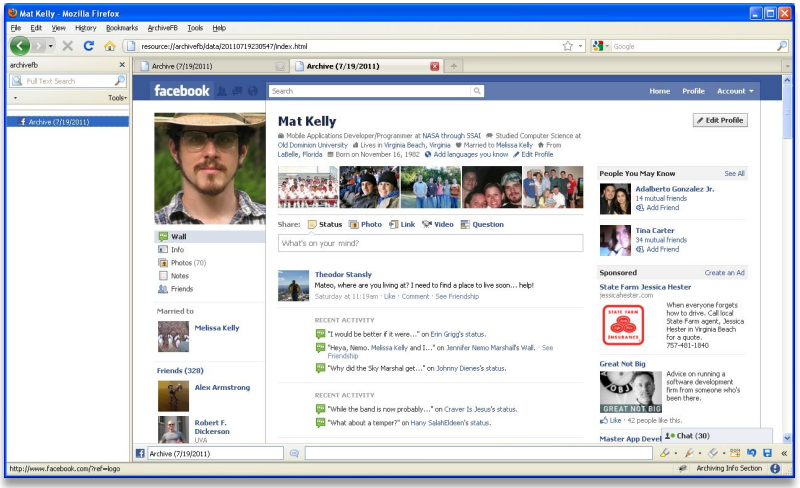
First brush with Web Archiving beyond IA

- Tasked to revamp and maintain ArchiveFacebook
  - Mozilla Firefox extension/add-on
- Provided mechanism to allow preservation of user's Facebook contents
- Created browser-accessible cache of FB web pages

"Replay" UI



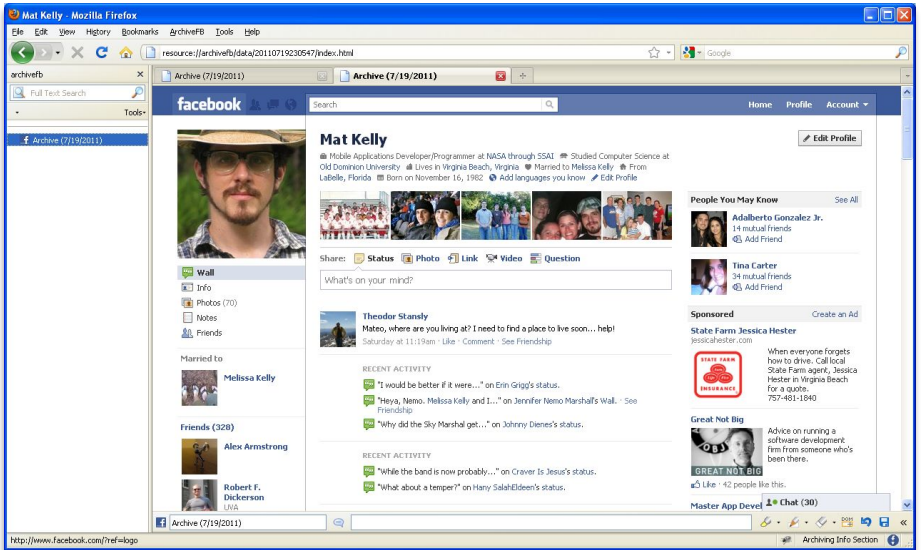
"Preservation" UI



Data Liberation vs. WYSIWYG

Facebook Native Profile Download

Archive Facebook Archiving Session Result

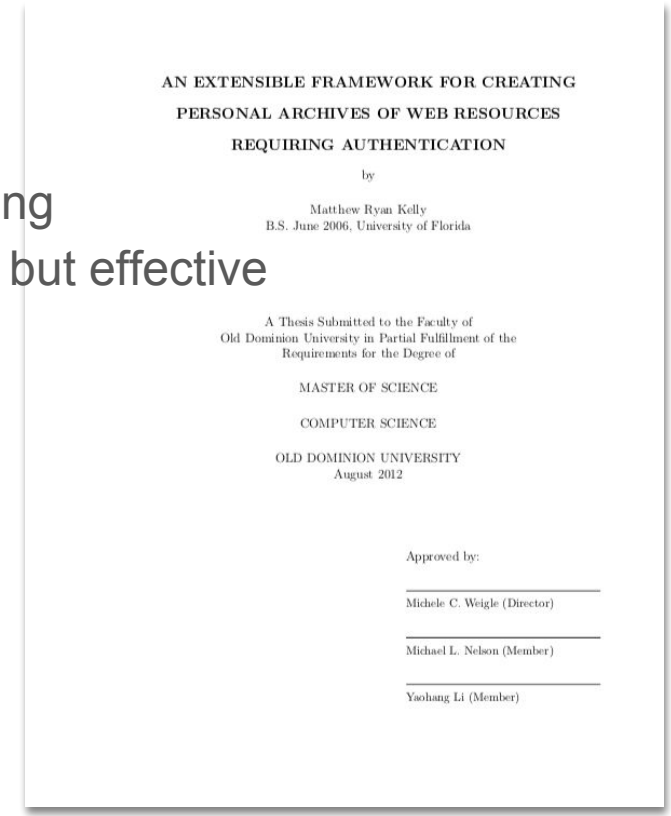


Research Software Beyond my Use Case

- Rapid prototyping
- Public releases of software
- Open source, permissively licensed (GPL or MIT)
- Rationales for Tool Build:
  - Data generation for further experimentation
  - Medium melding and merging (e.g., live & archived Web)
  - Exploration on the dynamics of previously unpreserved

Lessons Learned

- Site-specific scrapers are fragile
- Little guidance on the Web on Archival Tool Building
- Testing was ad hoc and laborious (moving target) but effective
- Created Framework for MS Thesis
  - Made these sort of tools more robust and adaptive



AN EXTENSIBLE FRAMEWORK FOR CREATING
PERSONAL ARCHIVES OF WEB RESOURCES
REQUIRING AUTHENTICATION

by

Matthew Ryan Kelly
B.S. June 2006, University of Florida

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

MASTER OF SCIENCE

COMPUTER SCIENCE

OLD DOMINION UNIVERSITY
August 2012

Approved by:

Michele C. Weigle (Director)

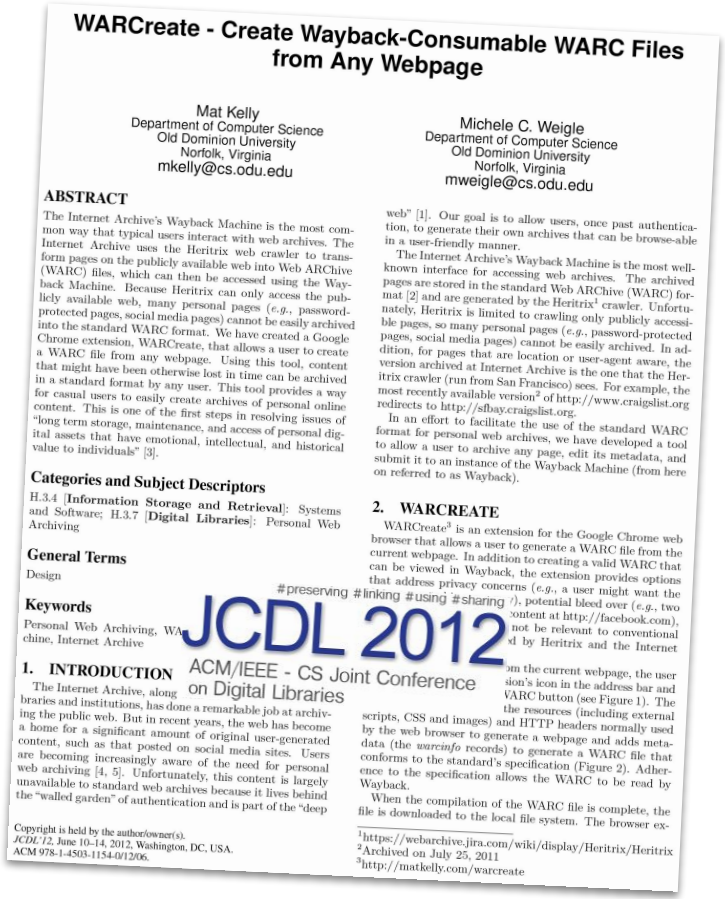
Michael L. Nelson (Member)

Yaohang Li (Member)

Site-Agnostic Preservation

- Preserve everything you see!
- Created files that adhere to standard ISO28500 (**W**eb **ARC**hive) format
- Enable individuals to preserve any Web page from their browser

`github.com/machawk1/warcreate`



WARCreate - Create Wayback-Consumable WARC Files from Any Webpage

Mat Kelly
Department of Computer Science
Old Dominion University
Norfolk, Virginia
mkelly@cs.odu.edu

Michele C. Weigle
Department of Computer Science
Old Dominion University
Norfolk, Virginia
mweigle@cs.odu.edu

ABSTRACT

The Internet Archive's Wayback Machine is the most common way that typical users interact with web archives. The Internet Archive uses the Heritrix web crawler to transform pages on the publicly available web into Web ARChive (WARC) files, which can then be accessed using the Wayback Machine. Because Heritrix can only access the publicly available web, many personal pages (*e.g.*, password-protected pages, social media pages) cannot be easily archived into the standard WARC format. We have created a Google Chrome extension, WARCreate, that allows a user to create a WARC file from any webpage. Using this tool, content that might have been otherwise lost in time can be archived in a standard format by any user. This tool provides a way for casual users to easily create archives of personal online content. This is one of the first steps in resolving issues of "long term storage, maintenance, and access of personal digital assets that have emotional, intellectual, and historical value to individuals" [3].

Categories and Subject Descriptors

H.3.4 [**Information Storage and Retrieval**]: Systems and Software; H.3.7 [**Digital Libraries**]: Personal Web Archiving

General Terms

Design

Keywords

Personal Web Archiving, WA
chine, Internet Archive

#preserving #linking #using #sharing

JCDL 2012
ACM/IEEE - CS Joint Conference on Digital Libraries

1. INTRODUCTION

The Internet Archive, along
braries and institutions, has done a remarkable job at archiving the public web. But in recent years, the web has become a home for a significant amount of original user-generated content, such as that posted on social media sites. Users are becoming increasingly aware of the need for personal web archiving [4, 5]. Unfortunately, this content is largely unavailable to standard web archives because it lives behind the "walled garden" of authentication and is part of the "deep web" [1]. Our goal is to allow users, once past authentication, to generate their own archives that can be browse-able in a user-friendly manner.

The Internet Archive's Wayback Machine is the most well-known interface for accessing web archives. The archived pages are stored in the standard Web ARChive (WARC) format [2] and are generated by the Heritrix[1] crawler. Unfortunately, Heritrix is limited to crawling only publicly accessible pages, so many personal pages (*e.g.*, password-protected pages, social media pages) cannot be easily archived. In addition, for pages that are location or user-agent aware, the version archived at Internet Archive is the one that the Heritrix crawler (run from San Francisco) sees. For example, the most recently available version[2] of http://www.craigslist.org redirects to http://sfbay.craigslist.org.

In an effort to facilitate the use of the standard WARC format for personal web archives, we have developed a tool to allow a user to archive any page, edit its metadata, and submit it to an instance of the Wayback Machine (from here on referred to as Wayback).

2. WARCREATE

WARCreate[3] is an extension for the Google Chrome web browser that allows a user to generate a WARC file from the current webpage. In addition to creating a valid WARC that can be viewed in Wayback, the extension provides options that address privacy concerns (*e.g.*, a user might want the potential bleed over (*e.g.*, two content at http://facebook.com), not be relevant to conventional d by Heritrix and the Internet om the current webpage, the user sion's icon in the address bar and VARC button (see Figure 1). The the resources (including external scripts, CSS and images) and HTTP headers normally used by the web browser to generate a webpage and adds metadata (the *warcinfo* records) to generate a WARC file that conforms to the standard's specification (Figure 2). Adherence to the specification allows the WARC to be read by Wayback.

When the compilation of the WARC file is complete, the file is downloaded to the local file system. The browser ex-

JCDL'12, June 10–14, 2012, Washington, DC, USA.
ACM 978-1-4503-1154-0/12/06.

[1] https://webarchive.jira.com/wiki/display/Heritrix/Heritrix
[2] Archived on July 25, 2011
[3] http://matkelly.com/warcreate

Mat Kelly and Michele C. Weigle, "WARCreate - Create Wayback-Consumable WARC Files from Any Webpage," In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries (JCDL)*. June 2012

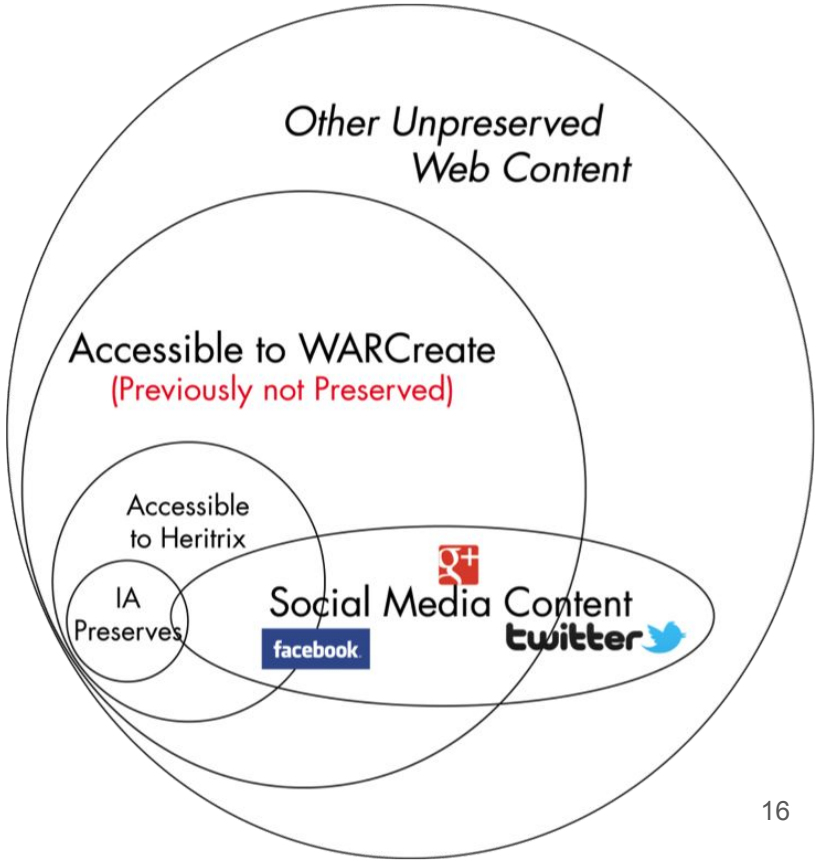
WARCreate *for Google Chrome*

- Facilitate preservation through familiar viewport (the browser)
- Extension for Google Chrome
  - Predated WebExtensions standard API
- Easy usage:
  - One-click, current webpage → WARC
- Acts as a “buffer” until commanded to create WARC

`github.com/machawk1/warcreate`

Archiving the Previously Unarchivable

- Target audience are for users that won't go to CLI
- Leveraging browser medium was novel and facilitated consistency



Initial limitations

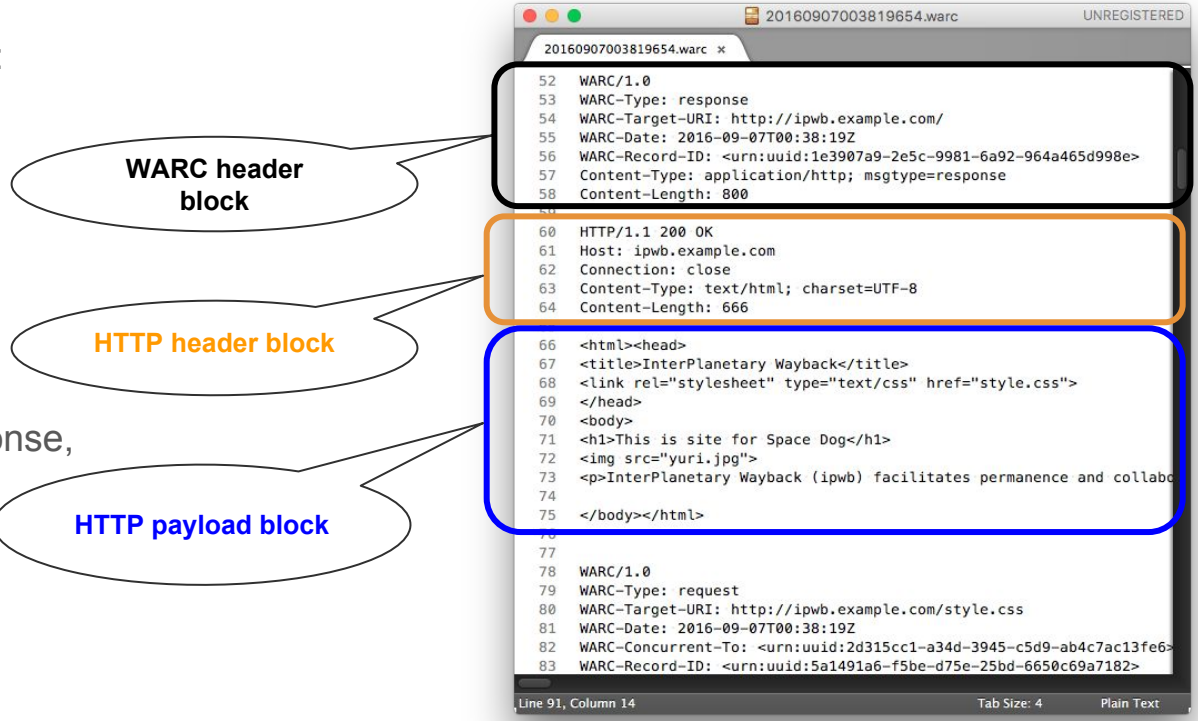
- Interacting with the File System was Limited
  - This was pre-HTML 5 File API
- Initial idea was to have Server-Side replay to also mitigate file limitations
  - This spun off “Web Archiving Integration Layer”
- As File APIs evolved, a “Server” was no longer needed for WARC generation

High Level Overview of WARC format

Concatenated records consisting of:

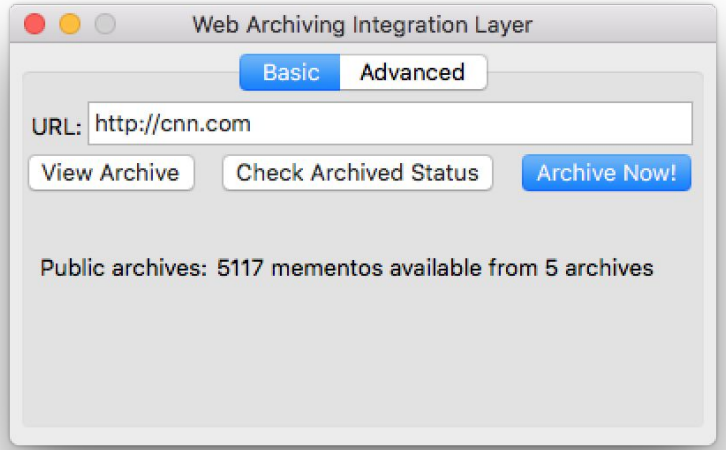
- WARC Headers
- WARC Payload
  - HTTP Headers
  - HTTP Payload

Information on HTTP request, response, DNS, metadata, general resources



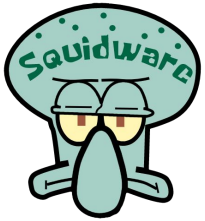
Web Archiving Integration Layer (WAIL)

- Written in Python, compiled to native application
  - initially OS X, Windows, and Linux
- Bundled and preconfigured “Institutional Grade” archiving tools
  - Heritrix (archival grade Web crawler)
  - OpenWayback (Web archive replay system)
- Again, simple interfaces to facilitate usage



Software that uses WARCs

Writers/Crawlers



...

Readers/Replay Engines

pywb

...

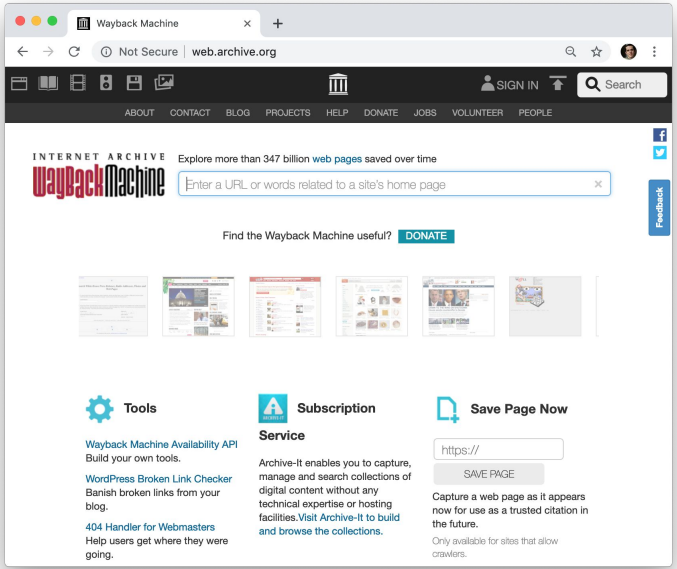
IA's Web Archives, stored in WARCs, use same tools

INTERNET ARCHIVE

WaybackMachine

OpenWayback

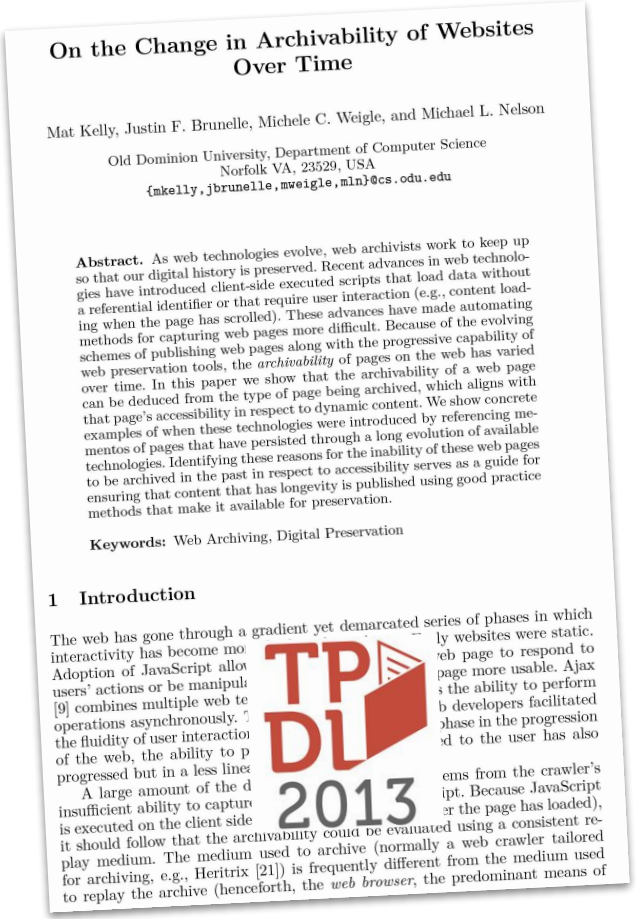
HERITRIX



Studying the Archived Web Beyond Tools and Formats

Study of Archiving Difficulties

- An initial examination of large Web archives
  - cf.live Web
- Which things are hard to preserve?



On the Change in Archivability of Websites Over Time

Mat Kelly, Justin F. Brunelle, Michele C. Weigle, and Michael L. Nelson

Old Dominion University, Department of Computer Science
Norfolk VA, 23529, USA
{mkelly,jbrunelle,mweigle,mln}@cs.odu.edu

Abstract. As web technologies evolve, web archivists work to keep up so that our digital history is preserved. Recent advances in web technologies have introduced client-side executed scripts that load data without a referential identifier or that require user interaction (e.g., content loading when the page has scrolled). These advances have made automating methods for capturing web pages more difficult. Because of the evolving schemes of publishing web pages along with the progressive capability of web preservation tools, the *archivability* of pages on the web has varied over time. In this paper we show that the archivability of a web page can be deduced from the type of page being archived, which aligns with that page's accessibility in respect to dynamic content. We show concrete examples of when these technologies were introduced by referencing mementos of pages that have persisted through a long evolution of available technologies. Identifying these reasons for the inability of these web pages to be archived in the past in respect to accessibility serves as a guide for ensuring that content that has longevity is published using good practice methods that make it available for preservation.

Keywords: Web Archiving, Digital Preservation

1 Introduction

The web has gone through a gradient yet demarcated series of phases in which interactivity has become mo… …ly websites were static. Adoption of JavaScript allo… …eb page to respond to users' actions or be manipul… …page more usable. Ajax [9] combines multiple web te… …s the ability to perform operations asynchronously. … …b developers facilitated the fluidity of user interactio… …hase in the progression of the web, the ability to p… …d to the user has also progressed but in a less line…

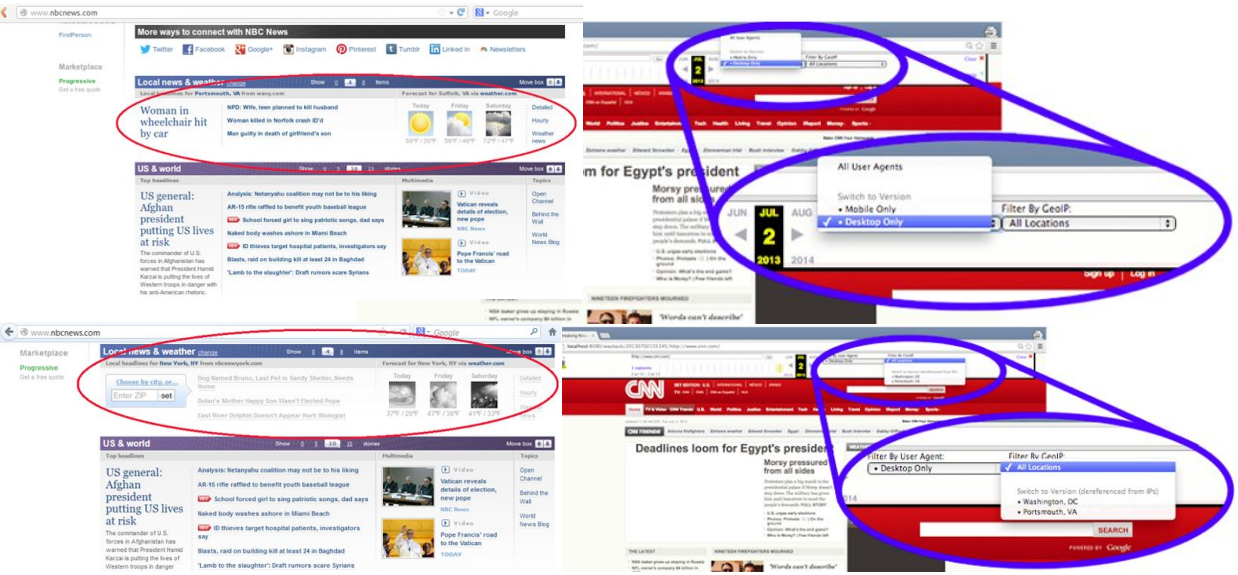
A large amount of the d… …ems from the crawler's insufficient ability to captur… …ipt. Because JavaScript is executed on the client side … …r the page has loaded), it should follow that the archivability could be evaluated using a consistent replay medium. The medium used to archive (normally a web crawler tailored for archiving, e.g., Heritrix [21]) is frequently different from the medium used to replay the archive (henceforth, the *web browser*, the predominant means of

TPDL 2013

Mat Kelly, Justin F. Brunelle, Michele C. Weigle, and Michael L. Nelson, "On the Change in Archivability of Websites Over Time," In Proceedings of the International Conference on Theory and Practice of Digital Libraries (TPDL). September 2013, pp. 35-47

Personalized Pages in the Archives

- An initial examination of large Web archives cf. live Web
- Some preserved things are personalized

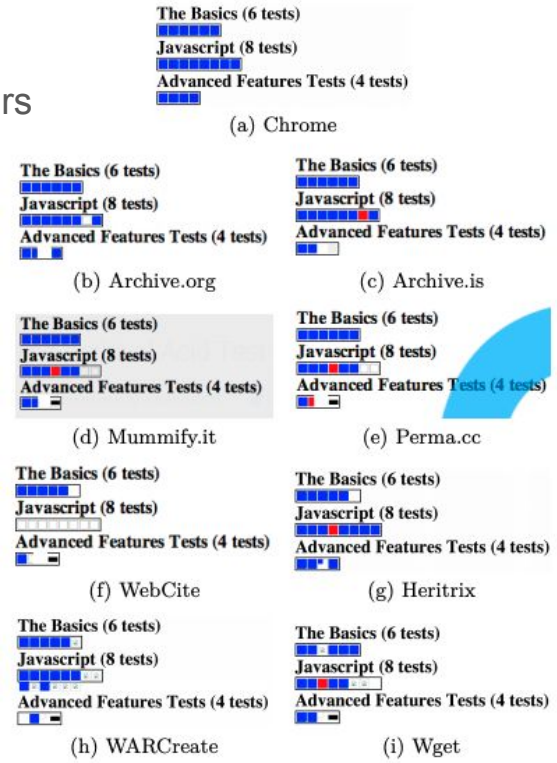


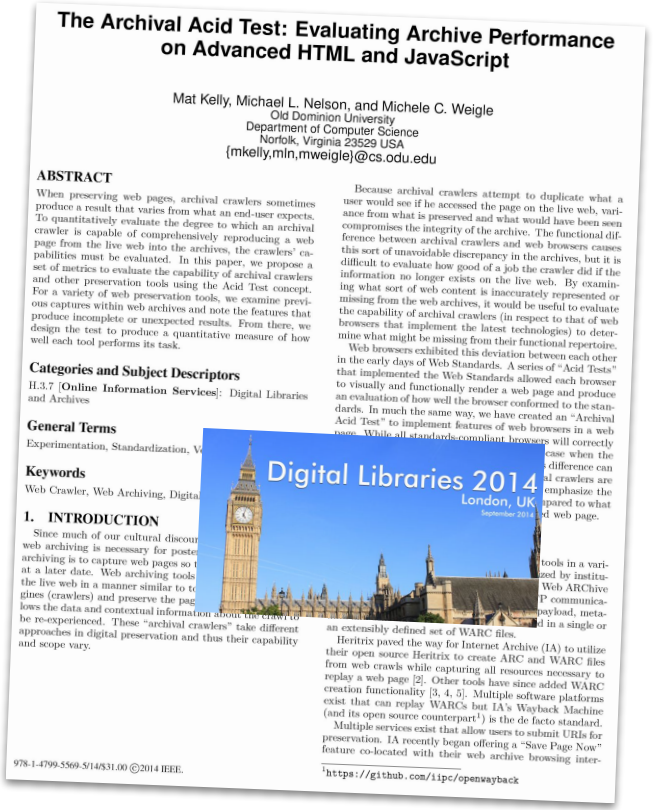
Mat Kelly, Justin F. Brunelle, Michele C. Weigle and Michael L. Nelson, "*A Method for Identifying Personalized Representations in the Archives*," D-Lib Magazine, 19(11/12), 2013.

Existing Tools' Capabilities

Punchline:

- Preservation tools lag in capability cf. Web browsers
- How well do archiving tools perform?





The Archival Acid Test: Evaluating Archive Performance on Advanced HTML and JavaScript

Mat Kelly, Michael L. Nelson, and Michele C. Weigle
Old Dominion University
Department of Computer Science
Norfolk, Virginia 23529 USA
{mkelly,mln,mweigle}@cs.odu.edu

ABSTRACT

When preserving web pages, archival crawlers sometimes produce a result that varies from what an end-user expects. To quantitatively evaluate the degree to which an archival crawler is capable of comprehensively reproducing a web page from the live web into the archives, the crawlers' capabilities must be evaluated. In this paper, we propose a set of metrics to evaluate the capability of archival crawlers and other preservation tools using the Acid Test concept. For a variety of web preservation tools, we examine previous captures within web archives and note the features that produce incomplete or unexpected results. From there, we design the test to produce a quantitative measure of how well each tool performs its task.

Because archival crawlers attempt to duplicate what a user would see if he accessed the page on the live web, variance from what is preserved and what would have been seen compromises the integrity of the archive. The functional difference between archival crawlers and web browsers causes this sort of unavoidable discrepancy in the archives, but it is difficult to evaluate how good of a job the crawler did if the information no longer exists on the live web. By examining what sort of web content is inaccurately represented or missing from the web archives, it would be useful to evaluate the capability of archival crawlers (in respect to that of web browsers that implement the latest technologies) to determine what might be missing from their functional repertoire.

Web browsers exhibited this deviation between each other in the early days of Web Standards. A series of "Acid Tests" that implemented the Web Standards allowed each browser to visually and functionally render a web page and produce an evaluation of how well the browser conformed to the standards. In much the same way, we have created an "Archival Acid Test" to implement features of web browsers in a web page. While all standards-compliant browsers will correctly

Categories and Subject Descriptors

H.3.7 [Online Information Services]: Digital Libraries and Archives

General Terms

Experimentation, Standardization, V

Keywords

Web Crawler, Web Archiving, Digita

I. INTRODUCTION

Since much of our cultural discour
web archiving is necessary for poste
archiving is to capture web pages so t
at a later date. Web archiving tools
the live web in a manner similar to t
gines (crawlers) and preserve the pag
lows the data and contextual information about the crawling
be re-experienced. These "archival crawlers" take different
approaches in digital preservation and thus their capability
and scope vary.

an extensibly defined set of WARC files.

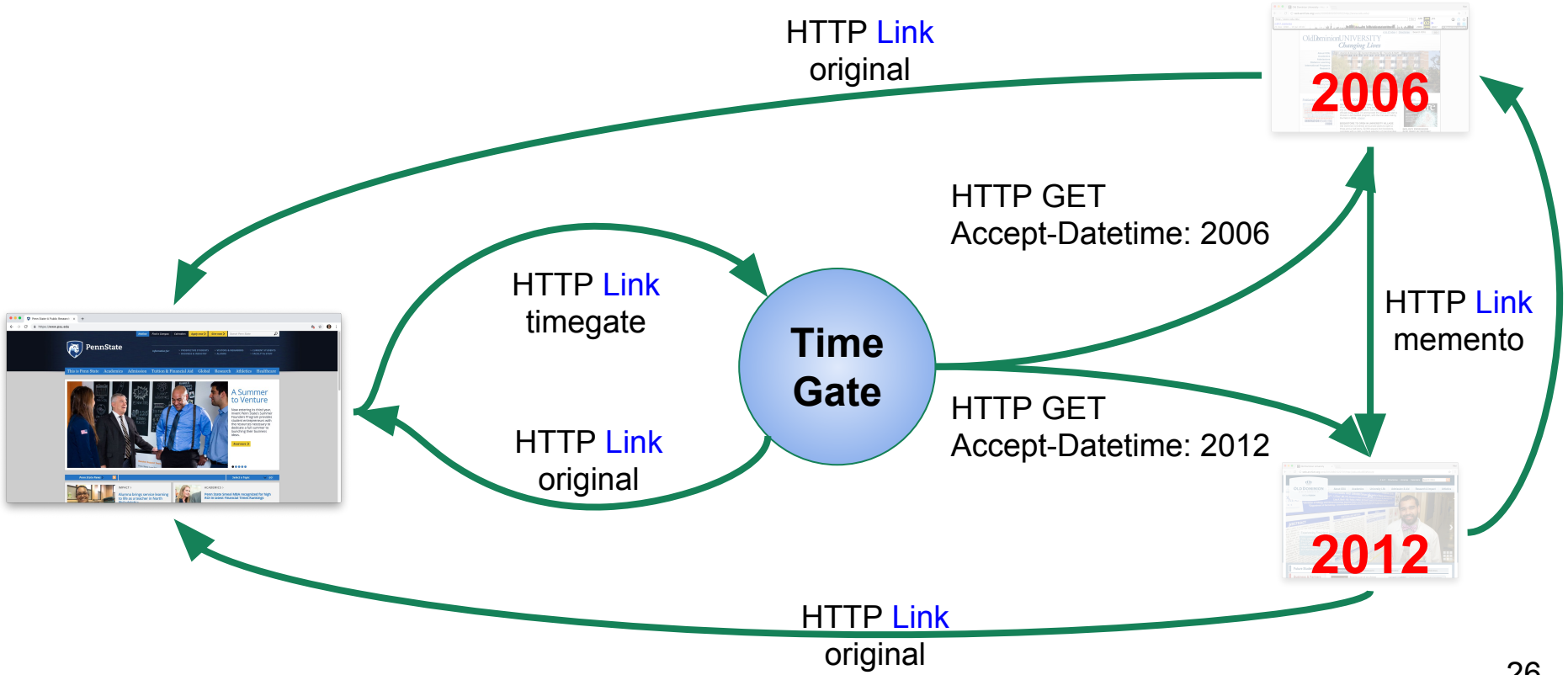
Heritrix paved the way for Internet Archive (IA) to utilize their open source Heritrix to create ARC and WARC files from web crawls while capturing all resources necessary to replay a web page [2]. Other tools have since added WARC creation functionality [3, 4, 5]. Multiple software platforms exist that can replay WARCs but IA's Wayback Machine (and its open source counterpart[1]) is the de facto standard.

Multiple services exist that allow users to submit URIs for preservation. IA recently began offering a "Save Page Now" feature co-located with their web archive browsing inter-

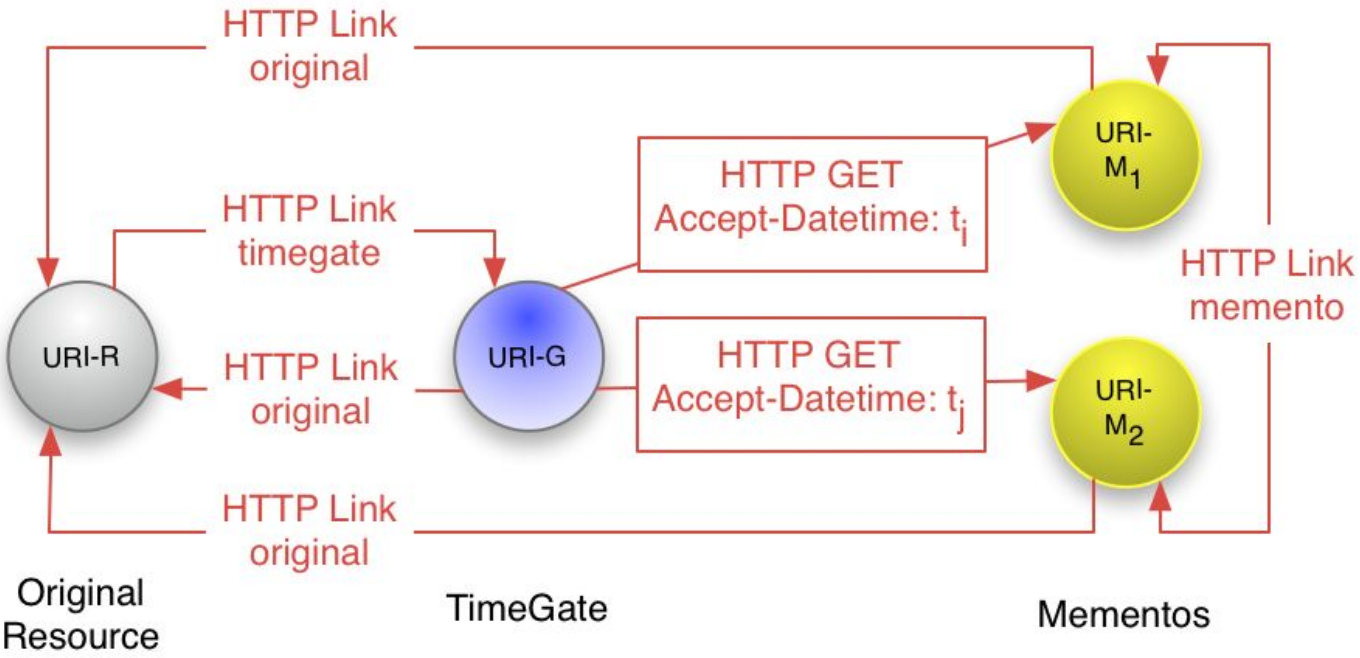
978-1-4799-5569-5/14/$31.00 ©2014 IEEE.

[1]https://github.com/iipc/openwayback

Representations can be Linked in time



Memento provides relations

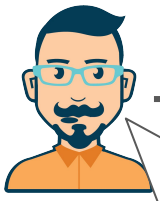


Memento Guide: Introduction. http://www.mementoweb.org/guide/quick-intro/, January 2015.

* H. Van de Sompel et al. *HTTP Framework for Time-Based Access to Resource States – Memento.* IETF RFC 7089, December 2013.

Memento Request Example

Memento Request Example



Request `psu.edu` at
Sept 11, 2001 at 9am EST

HTTP Request

- Accept-Datetime: Wed, 02 Aug 2017 23:15:00 GMT
- GET: http://web.archive.org/web/http://www.psu.edu

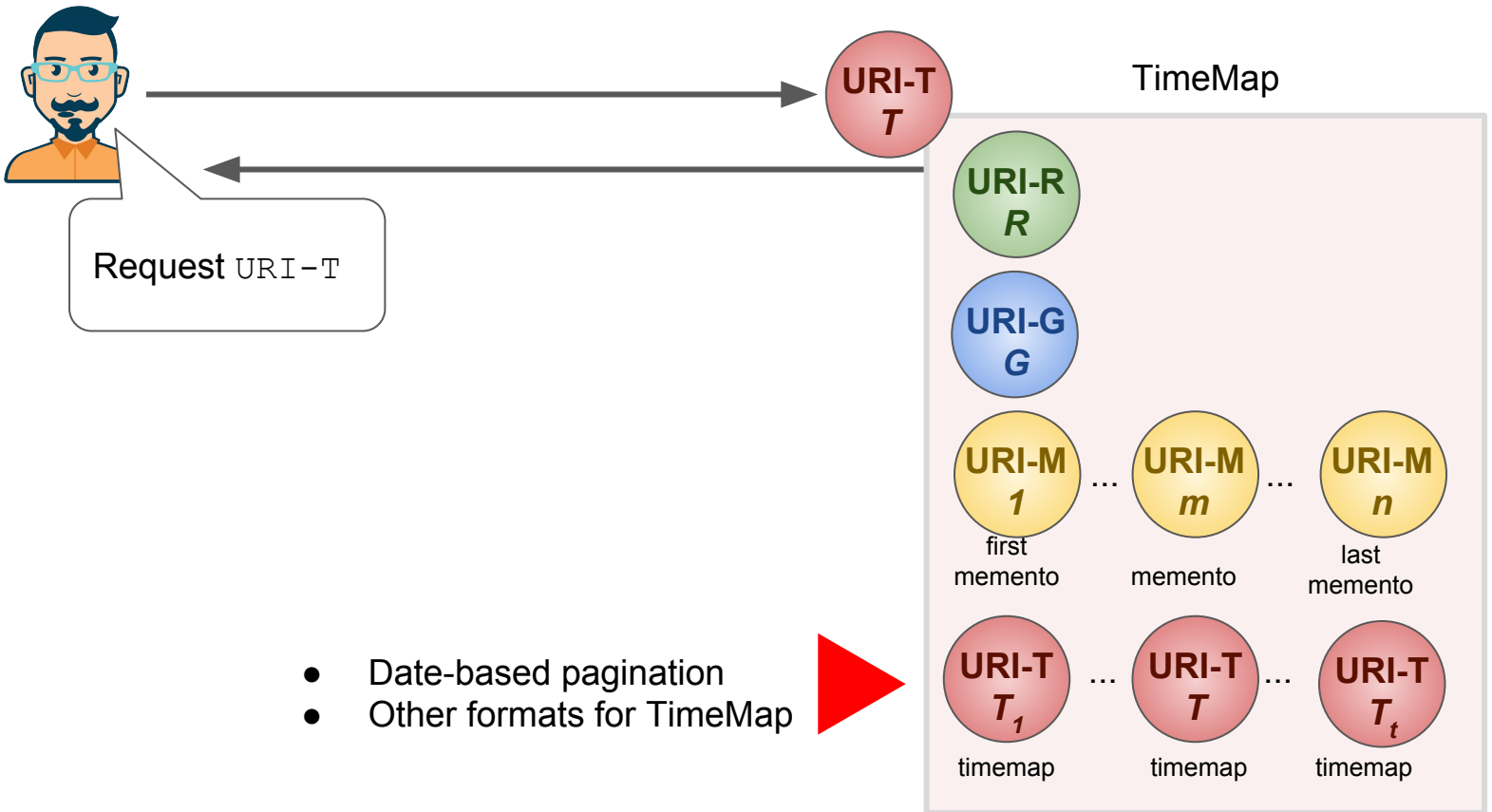
URI-G
G

HTTP Response (302)

- Memento-Datetime: Wed, 02 Aug 2017 23:18:04 GMT
- Location: http://web.archive.org/web/20170802231804/http://www.psu.edu
- Link:

URI-T *T* — timemap
URI-R *R* — original
URI-G *G* — timegate
URI-M *M* — memento

Background: Dereferencing a TimeMap at URI-T



- Date-based pagination
- Other formats for TimeMap

Memento “Damage” Metric

- Not all missing resource are created equal
- Multiple studies establishing metric
- Evaluated through Mechanical Turk

Not All Mementos Are Created Equal: Measuring The Impact Of Missing Resources

Justin F. Brunelle, Mat Kelly, Hany SalahEldeen, Michele C. Weigle, and Michael L. Nelson
Old Dominion University
Department of Computer Science
Norfolk, Virginia, 23529
{jbrunelle, mkelly, hany, mweigle, mln}@cs.odu.edu

ABSTRACT

Categories and Subject Descriptors

General Terms

Keywords

Digital Libraries 201

Int J Digit Libr (2015) 16:283–301
DOI 10.1007/s00799-015-0150-6

Not all mementos are created equal: measuring the impact of missing resources

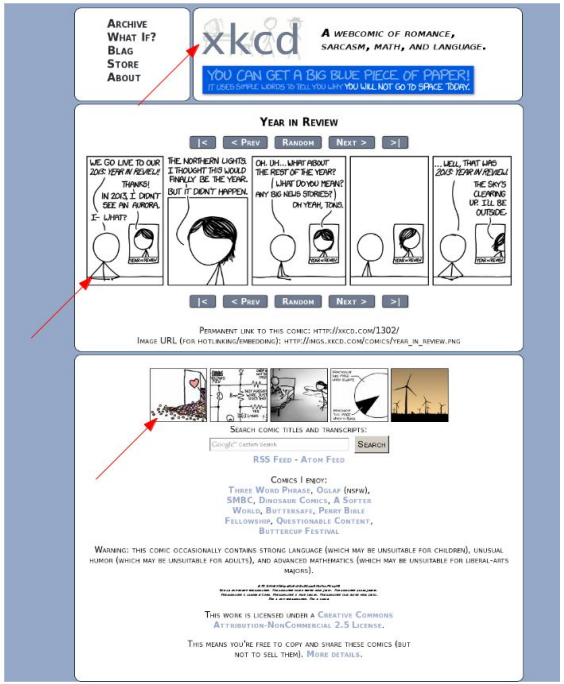
Justin F. Brunelle[1] · Mat Kelly[1] · Hany SalahEldeen[1] · Michele C. Weigle[1] · Michael L. Nelson[1]

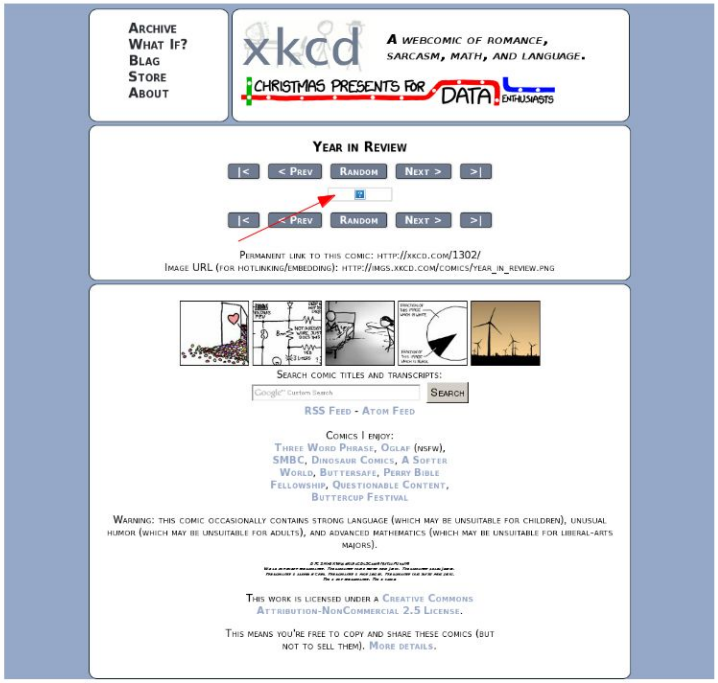
Received: 3 December 2014 / Revised: 22 April 2015 / Accepted: 22 April 2015 / Published online: 6 May 2015

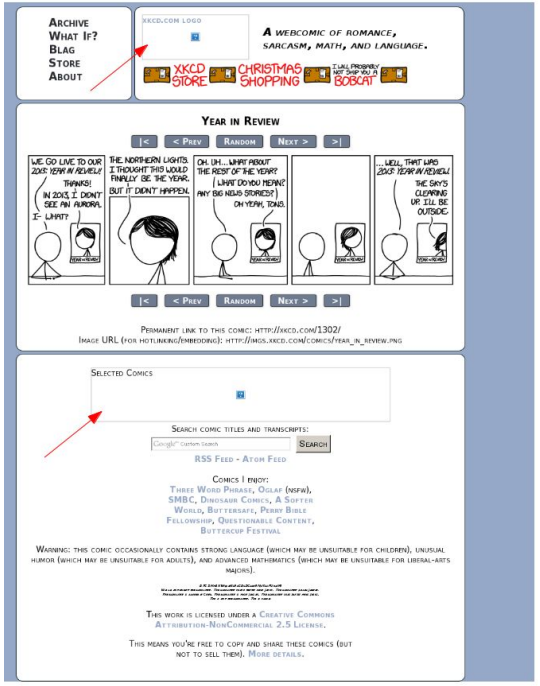
Abstract

INTERNATIONAL JOURNAL ON Digital Libraries

Impact of Missing Resources







m_0 m_1 m_2

Damage in the eyes of Mechanical Turkers

- m_0: live Web
- m_1: comic removed
- m_2: two logo images removed

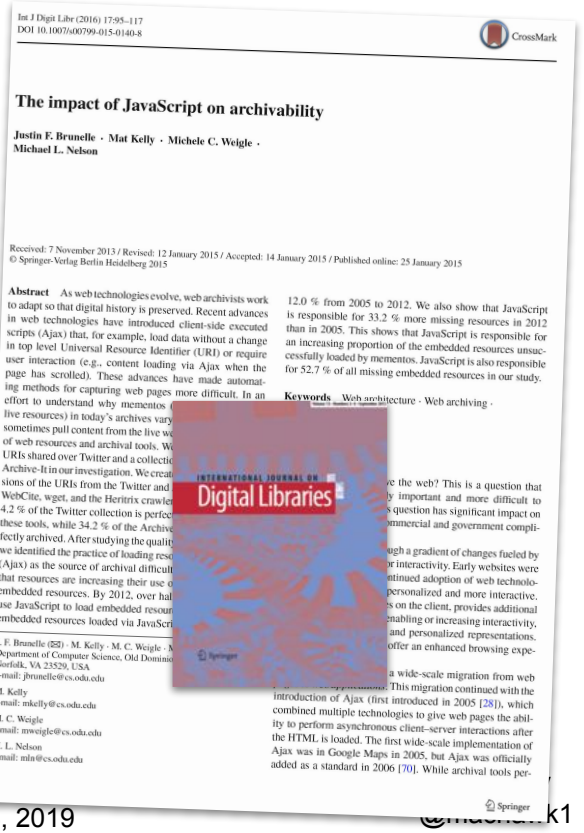
The turkers selected m_0 as the preferred memento 81% of the time, and more consistently for larger ΔM_m values.

ΔM_m	Splits						
	5-0	4-1	3-2	2-3	1-4	0-5	Total
1.0							0.00
0.9							0.00
0.8	4						0.07
0.7							0.00
0.6							0.00
0.5	1	1					0.04
0.4							0.00
0.3	15	5					0.36
0.2	2						0.04
0.1	5	4	4	2		1	0.29
0.0	5	3	1	3			0.22
Total	0.58	0.23	0.09	0.09	0.00	0.02	1.0

Justin F. Brunelle, Mat Kelly, Hany SalahEldeen, Michele C. Weigle, and Michael L. Nelson, “Not All Mementos Are Created Equal: Measuring the Impact of Missing Resources,” International Journal on Digital Libraries (IJDL), 16(3), pp. 283-301.

Impact of JavaScript on Archivability

- Missing JavaScript has big ramifications
- Content Complexity (CC) measure
- URIs shared over Twitter and from Archive-It collection
- Evaluated WebCitation, wget, and the Heritrix
- 4.2% of the Twitter collection is perfectly archived by all of these tools
- 34.2% of the Archive-It collection is perfectly archived.



Int J Digit Libr (2016) 17:95–117
DOI 10.1007/s00799-015-0140-8

CrossMark

The impact of JavaScript on archivability

Justin F. Brunelle · Mat Kelly · Michele C. Weigle · Michael L. Nelson

Received: 7 November 2013 / Revised: 12 January 2015 / Accepted: 14 January 2015 / Published online: 25 January 2015

Abstract As web technologies evolve, web archivists work to adapt so that digital history is preserved. Recent advances in web technologies have introduced client-side executed scripts (Ajax) that, for example, load data without a change in top level Universal Resource Identifier (URI) or require user interaction (e.g., content loading via Ajax when the page has scrolled). These advances have made automating methods for capturing web pages more difficult. In an effort to understand why mementos (
live resources) in today's archives vary
sometimes pull content from the live we
of web resources and archival tools. We
URIs shared over Twitter and a collectio
Archive-It in our investigation. We creat
sions of the URIs from the Twitter and
WebCite, wget, and the Heritrix crawler
4.2 % of the Twitter collection is perfec
these tools, while 34.2 % of the Archive
fectly archived. After studying the qualit
we identified the practice of loading reso
(Ajax) as the source of archival difficult
that resources are increasing their use o
embedded resources. By 2012, over hal
use JavaScript to load embedded resour
embedded resources loaded via JavaScri
12.0 % from 2005 to 2012. We also show that JavaScript is responsible for 33.2 % more missing resources in 2012 than in 2005. This shows that JavaScript is responsible for an increasing proportion of the embedded resources unsuccessfully loaded by mementos. JavaScript is also responsible for 52.7 % of all missing embedded resources in our study.

Keywords Web architecture · Web archiving ·

J. F. Brunelle (✉) · M. Kelly · M. C. Weigle · M
Department of Computer Science, Old Dominio
Norfolk, VA 23529, USA
e-mail: jbrunelle@cs.odu.edu

M. Kelly
e-mail: mkelly@cs.odu.edu

M. C. Weigle
e-mail: mweigle@cs.odu.edu

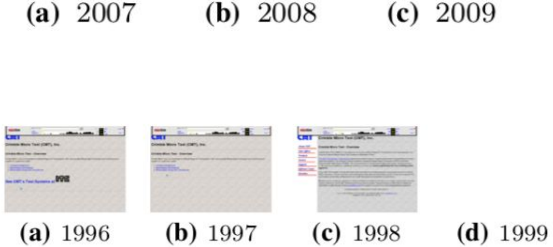
M. L. Nelson
e-mail: mln@cs.odu.edu

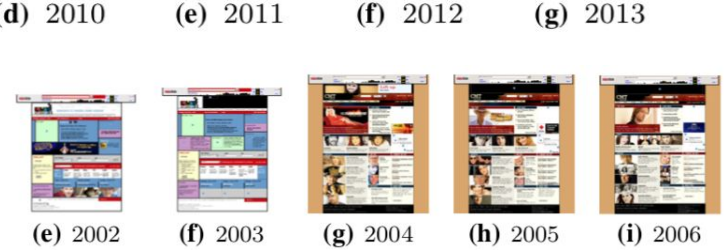
INTERNATIONAL JOURNAL ON Digital Libraries

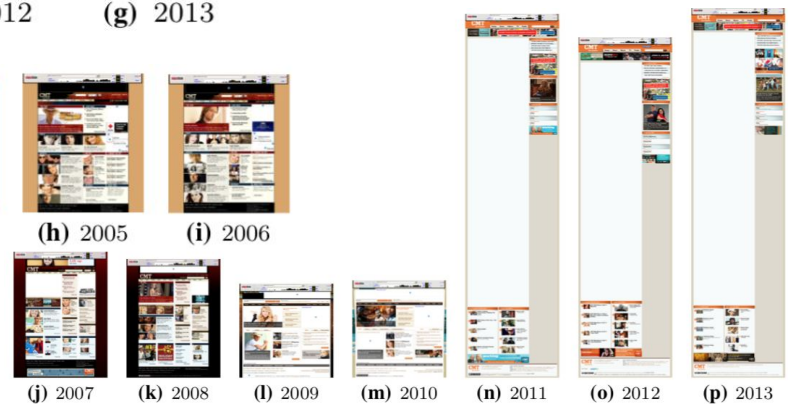
Springer

Missing resources a direct result of JS usage

www.doc.alabama.gov
perfectly archived in time

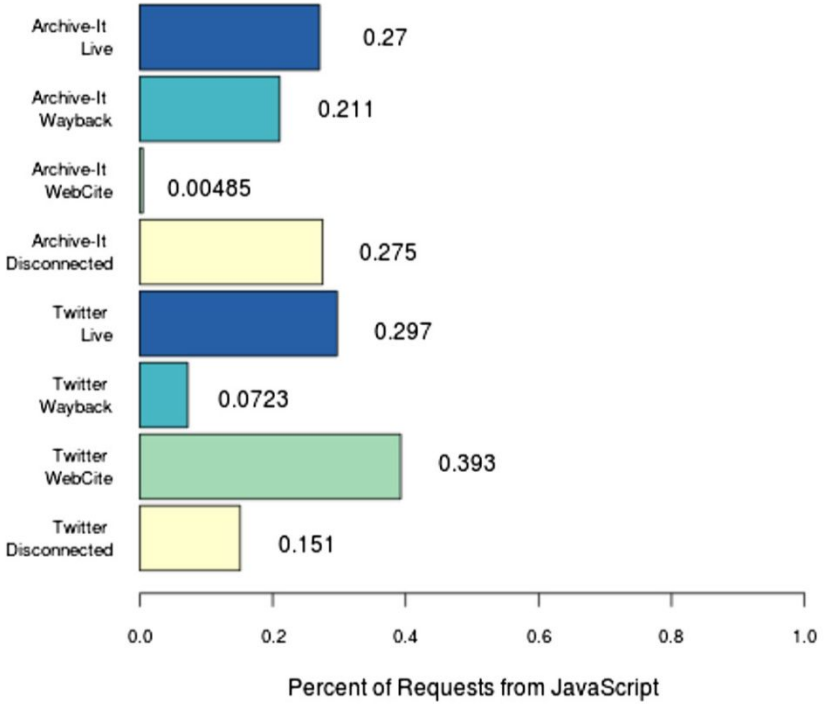


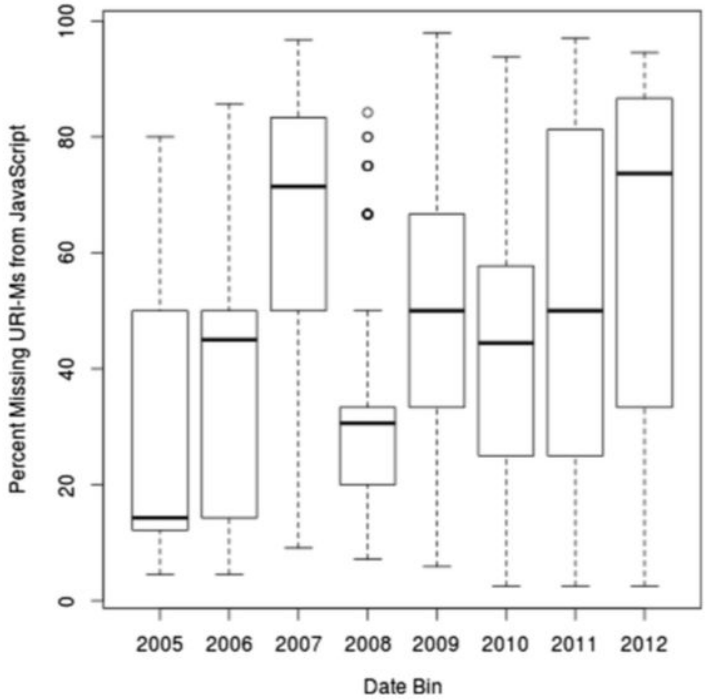




cmt.com
Impact of JS evident in missing resources

Problem *was* getting worse

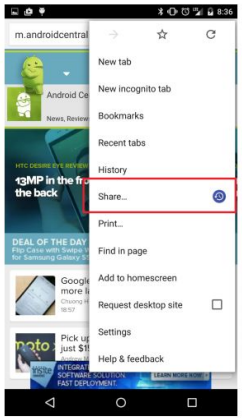


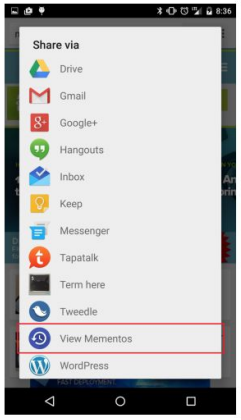


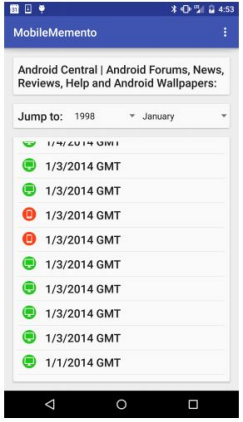
Justin F. Brunelle, Mat Kelly, Michele C. Weigle and Michael L. Nelson, “The Impact of JavaScript on Archivability,” International Journal on Digital Libraries (IJDL), 17(2), pp. 95-117.

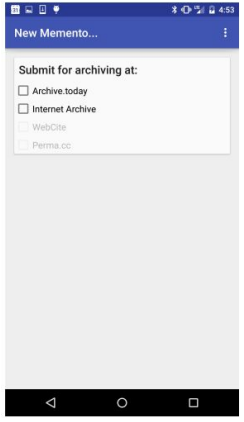
Mobile Mink

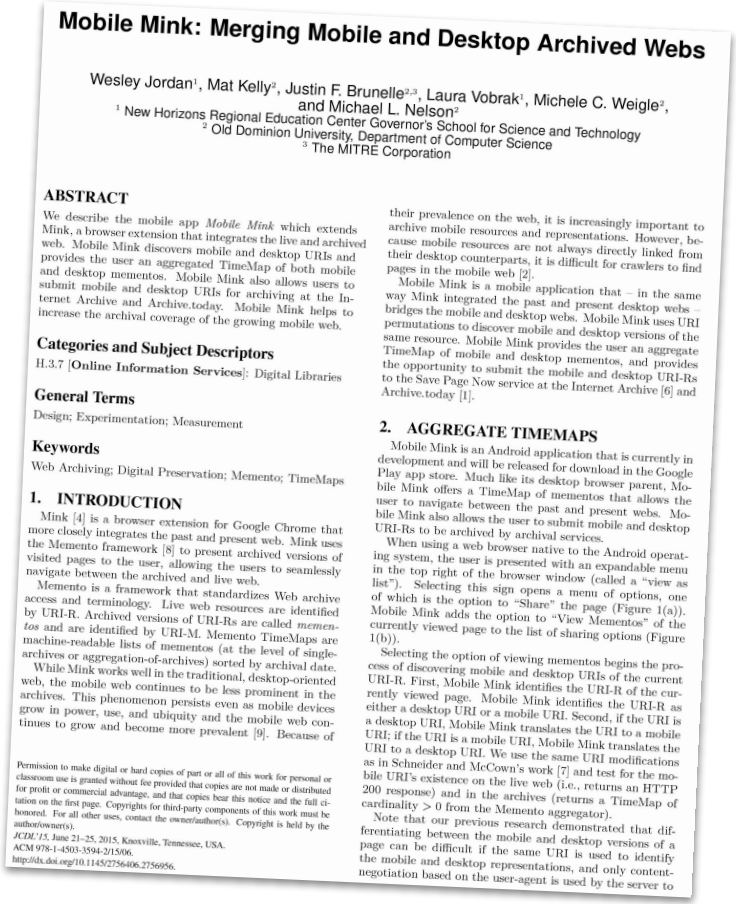
- Android Application
  - Few web archiving offerings in this realm
- Leveraged Native “Share” feature for archival lookup
- Identified and associated archived mobile representations











Mobile Mink: Merging Mobile and Desktop Archived Webs

Wesley Jordan[1], Mat Kelly[2], Justin F. Brunelle[2,3], Laura Vobrak[1], Michele C. Weigle[2], and Michael L. Nelson[2]
[1] New Horizons Regional Education Center Governor's School for Science and Technology
[2] Old Dominion University, Department of Computer Science
[3] The MITRE Corporation

ABSTRACT

We describe the mobile app *Mobile Mink* which extends Mink, a browser extension that integrates the live and archived web. Mobile Mink discovers mobile and desktop URIs and provides the user an aggregated TimeMap of both mobile and desktop mementos. Mobile Mink also allows users to submit mobile and desktop URIs for archiving at the Internet Archive and Archive.today. Mobile Mink helps to increase the archival coverage of the growing mobile web.

Categories and Subject Descriptors

H.3.7 [**Online Information Services**]: Digital Libraries

General Terms

Design; Experimentation; Measurement

Keywords

Web Archiving; Digital Preservation; Memento; TimeMaps

1. INTRODUCTION

Mink [4] is a browser extension for Google Chrome that more closely integrates the past and present web. Mink uses the Memento framework [8] to present archived versions of visited pages to the user, allowing the users to seamlessly navigate between the archived and live web.

Memento is a framework that standardizes Web archive access and terminology. Live web resources are identified by URI-R. Archived versions of URI-Rs are called *mementos* and are identified by URI-M. Memento TimeMaps are machine-readable lists of mementos (at the level of single-archives or aggregation-of-archives) sorted by archival date.

While Mink works well in the traditional, desktop-oriented web, the mobile web continues to be less prominent in the archives. This phenomenon persists even as mobile devices grow in power, use, and ubiquity and the mobile web continues to grow and become more prevalent [9]. Because of their prevalence on the web, it is increasingly important to archive mobile resources and representations. However, because mobile resources are not always directly linked from their desktop counterparts, it is difficult for crawlers to find pages in the mobile web [2].

Mobile Mink is a mobile application that – in the same way Mink integrated the past and present desktop webs – bridges the mobile and desktop webs. Mobile Mink uses URI permutations to discover mobile and desktop versions of the same resource. Mobile Mink provides the user an aggregate TimeMap of mobile and desktop mementos, and provides the opportunity to submit the mobile and desktop URI-Rs to the Save Page Now service at the Internet Archive [6] and Archive.today [1].

2. AGGREGATE TIMEMAPS

Mobile Mink is an Android application that is currently in development and will be released for download in the Google Play app store. Much like its desktop browser parent, Mobile Mink offers a TimeMap of mementos that allows the user to navigate between the past and present webs. Mobile Mink also allows the user to submit mobile and desktop URI-Rs to be archived by archival services.

When using a web browser native to the Android operating system, the user is presented with an expandable menu in the top right of the browser window (called a “view as list”). Selecting this sign opens a menu of options, one of which is the option to “Share” the page (Figure 1(a)). Mobile Mink adds the option to “View Mementos” of the currently viewed page to the list of sharing options (Figure 1(b)).

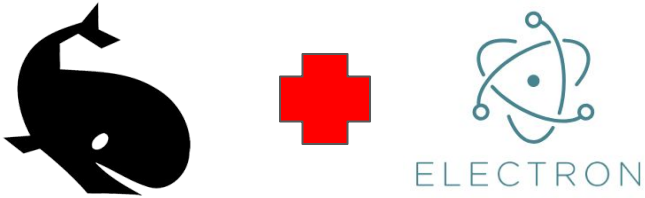
Selecting the option of viewing mementos begins the process of discovering mobile and desktop URIs of the current URI-R. First, Mobile Mink identifies the URI-R of the currently viewed page. Mobile Mink identifies the URI-R as either a desktop URI or a mobile URI. Second, if the URI is a desktop URI, Mobile Mink translates the URI to a mobile URI; if the URI is a mobile URI, Mobile Mink translates the URI to a desktop URI. We use the same URI modifications as in Schneider and McCown’s work [7] and test for the mobile URI’s existence on the live web (i.e., returns an HTTP 200 response) and in the archives (returns a TimeMap of cardinality > 0 from the Memento aggregator).

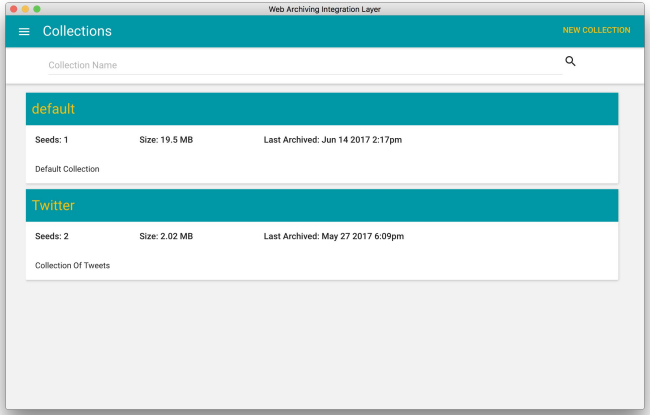
Note that our previous research demonstrated that differentiating between the mobile and desktop versions of a page can be difficult if the same URI is used to identify the mobile and desktop representations, and only content-negotiation based on the user-agent is used by the server to

JCDL'15, June 21–25, 2015, Knoxville, Tennessee, USA.
ACM 978-1-4503-3594-2/15/06.
http://dx.doi.org/10.1145/2756406.2756956.

WAIL reimagined

- Archive from the desktop with higher fidelity than conventional archiving tools





WAIL: Collection-Based Personal Web Archiving

John A. Berlin, Mat Kelly, Michael L. Nelson, Michele C. Weigle
Old Dominion University, Department of Computer Science, Norfolk VA, 23529, USA
{jberlin,mkelly,mln,mweigle}@cs.odu.edu

ABSTRACT

Web Archiving Integration Layer (WAIL) is a desktop application written in Python that integrates Heritrix and OpenWayback. In this work we recreate and extend WAIL from the ground up to facilitate collection-based personal Web archiving. Our new iteration of the software, WAIL-Electron, leverages native Web technologies (e.g., JavaScript, Chromium) using Electron to open new potential for Web archiving by individuals in a stand-alone cross-platform native application. By replacing OpenWayback with PyWb, we provide a novel means for personal Web archivists to curate collections of their captures from their own personal computer rather than relying on an external archival Web service. As extended features we also provide the ability for a user to monitor and automatically archive Twitter users' feeds, even those requiring authentication, as well as provide a reference implementation for integrating a browser-based preservation tool into an OS native application.

KEYWORDS

Personal Web Archiving

ACM Reference format:
John A. Berlin, Mat Kelly, Michael L. Nelson, Michele C. Weigle. 2017. WAIL: Collection-Based Personal Web Archiving. In *Proceedings of Joint Conference on Digital Libraries, Toronto, Ontario, Canada, June 2017 (JCDL '17)*, 2 pages.
DOI: 10.XXX/XXXX

1 INTRODUCTION

Subscription-based Web archiving services like Archive-it allow users with limited technical knowledge to create and replay personalized collections of Web archives. Archive-It provides its users with a simple interface to create collections and to launch on-demand archival crawl. Similar to Archive-It is Webrecorder[1], which allows any user to register for the service and provides them with the ability to create and manage personalized collections of Web archives. But unlike Archive-It, Webrecorder requires its users to drive the preservation process or upload content for replay, only providing its users up to five gigabytes of storage. Individuals that wish to freely (*gratis* and *libre*) archive Web pages without arbitrary restrictions beyond the limitations of their personal computers using institutional grade tools must setup an archival Web crawler (e.g., Heritrix) and replay system (e.g., Wayback), time consuming and technical tasks potentially beyond the individual's skill level. Even if a user is able to successfully set up these tools, they must also configure the crawls via Heritrix and come up with their own means of associating the Web archives to each other for collection-based replay via Wayback.

We have developed a tool that provides users with access to both Heritrix and Wayback while providing an interoperable mechanism for personal collection-based Web archiving from their personal computers. [illegible] and add to these collections through the software taking care of the details in managing collections, crawls, and replay. We have integrated a native Chromium[2] browser (the core of Google's Chrome Web browser) into the archival process in order to surface content specific to sites like Twitter for more accurate and comprehensive preservation.

2 WAIL

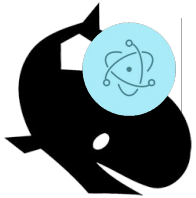
WAIL-Electron is a[n] [illegible]-based reimplementation of the original WAIL application [illegible] WAIL-Electron (from here on referred to as WAIL) allows users to create and manage personalized collections of Web archives from their personal computers. When a user first starts the application WAIL provides them with a default collection and the means to create additional collections straight away from the collection screen (Figure 1). The collection view displays an overview of the collections WAIL is currently managing and information about them. This information includes the number [illegible] in the collection along with the collection's size [illegible] updated. A user may easily create a new collection by clicking the "New Collection" button.

Doing so displays a dialog (Figure 2), prompting the user for a collection name, title, and description. These values are propagated to the WAIL interface and are viewable when replaying the collection through Wayback. When viewing a collection, WAIL displays

Figure 1: Collections screen

[1]https://webrecorder.io/
[2]https://www.chromium.org/
[3]http://electron.atom.io/

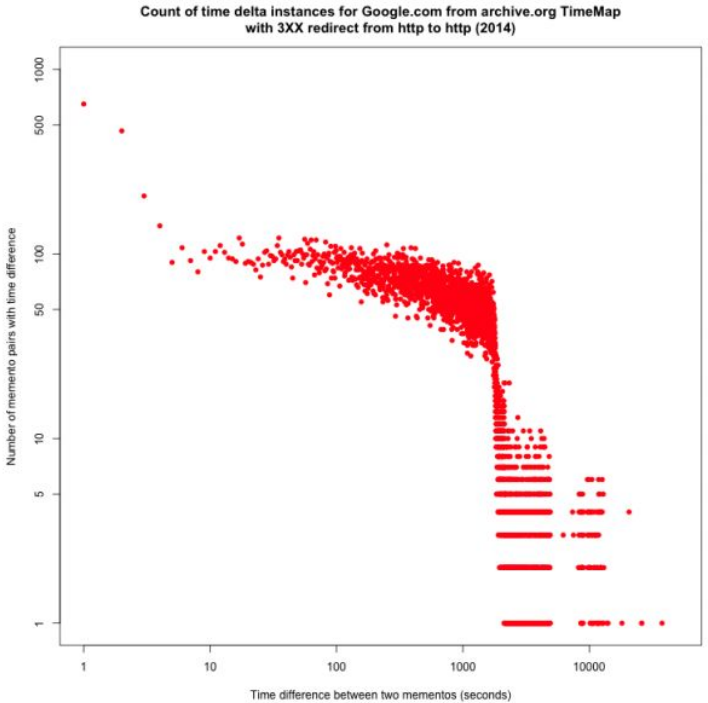
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
JCDL'17, Toronto, Ontario, Canada
© 2017 Copyright held by the owner/author(s). XXX-YYYY-ZZ-AAA/BB/CC…$15.00
DOI: 10.XXX/XXXX



Code	Python	HTML, JavaScript, Electron
Archival organization	Single Collection	Multiple Collections
User interface	System Native	Material
Archival Crawler	Heritrix	Heritrix, node-warc
Archival Replay	OpenWayback	pywb
Release	macOS, Windows	macOS, Windows, Linux
Source	github.com/machawk1/wail	github.com/n0tan3rd/wail

Conicalization's effects over time

URI coalescence considered harmful for archives



Impact of URI Canonicalization on Memento Count

Mat Kelly, Lulwah M. Alkwai, Sawood Alam, Michael L. Nelson, and Michele C. Weigle
Old Dominion University
Department of Computer Science
Norfolk, Virginia, USA
{mkelly,lalkwai,salam,mln,mweigle}@cs.odu.edu

Herbert Van de Sompel
Los Alamos National Laboratory
Los Alamos, New Mexico, USA
herbertv@lanl.gov

ABSTRACT

Memento TimeMaps [5] list identifiers for archival web captures (URI-Ms). When some URI-Ms are dereferenced, they redirect to a different URI-M instead of a unique representation at the datetime. This suggests that confidently obtaining an accurate count quantifying the number of non-forwarding captures for an Original Resource URI (URI-R) is not possible using a TimeMap alone and that the magnitude of a TimeMap is not equivalent to the number of representations it identifies. This work represents an abbreviated version of the full technical report describing this phenomena in depth [3]. For google.com we found that 84.9% of the URI-Ms in a TimeMap result in an HTTP redirect when dereferenced. The full study applies this technique to seven other URI-Rs of large Web sites and 13 academic institutions. Using a ratio metric for the number of URI-Ms without redirects those requiring a redirect when dereferenced, five of large web sites' and two of the thirteen academ tions' TimeMaps had a ratio of less than one, more than half of the URI-Ms in these Ti redirects when dereferenced.

year	M_{TM}	M_{RC}	DI
2006	735	483	1.917
2007	1,055	842	3.953
2008	1,376	894	1.855
2009	6,074	4,335	2.493
2010	9,326	6,530	2.335
2011	20,634	9,279	0.817
2012	102,533	16,240	0.188
2013	228,405	25,203	0.124
2014	164,865	22,738	0.160
2015	17,978	11,286	1.686
2016	139,520	5,805	0.043

Table 1: Google over time (abbreviated), bucketed by year, based on IA mementos extracted from the TimeMap. M_{TM} is the memento count based solely on the data in the TimeMap, M_{RC} is the count based on exclusion of redirects when dereferencing mementos. DI is the ratio of non-redirecting mementos to redirecting mementos.

1 INTRODUCTION

Web archives return TimeMaps with a HTTP transactions observed at archi have generally been used as a con resentations of a URI-R pres TimeMaps may include URI-Ms for a redirections, and errors [2]. For exampl for http://vimeo.com produce an HTT URI-M is in the TimeMap that returns a TimeMaps do not explicitly return a "co cate the number of mementos listed in the produce a non-redirecting HTTP status code w enced. The heuristic of determining how many cap represented by URI-Ms in a TimeMap cannot be complet without dereferencing.

Redirection in a Web archive can be attributed to a variety of canonicalization rules [3]. Preserving and replaying these redirects allows an archive to accurately reproduce the HTTP transactions that would have occurred when the URI being accessed resided on the live Web. Because of the potential for redirection, the heuristic of counting URI-Ms with relation values of "memento" is an inaccurate means of determining the number of unique representations inferred from a TimeMap. We further emphasize the distinction per the Memento specification that the identifiers for mementos

2 BACKGROUND AND RELATED WORK

URI canonicalization associates differently formatted URIs ref to the same resource. As URI schemes from a Web site change over time, canonicalization is critical for retaining a cohesive, comprehensive listing of the mementos available for a Web page.

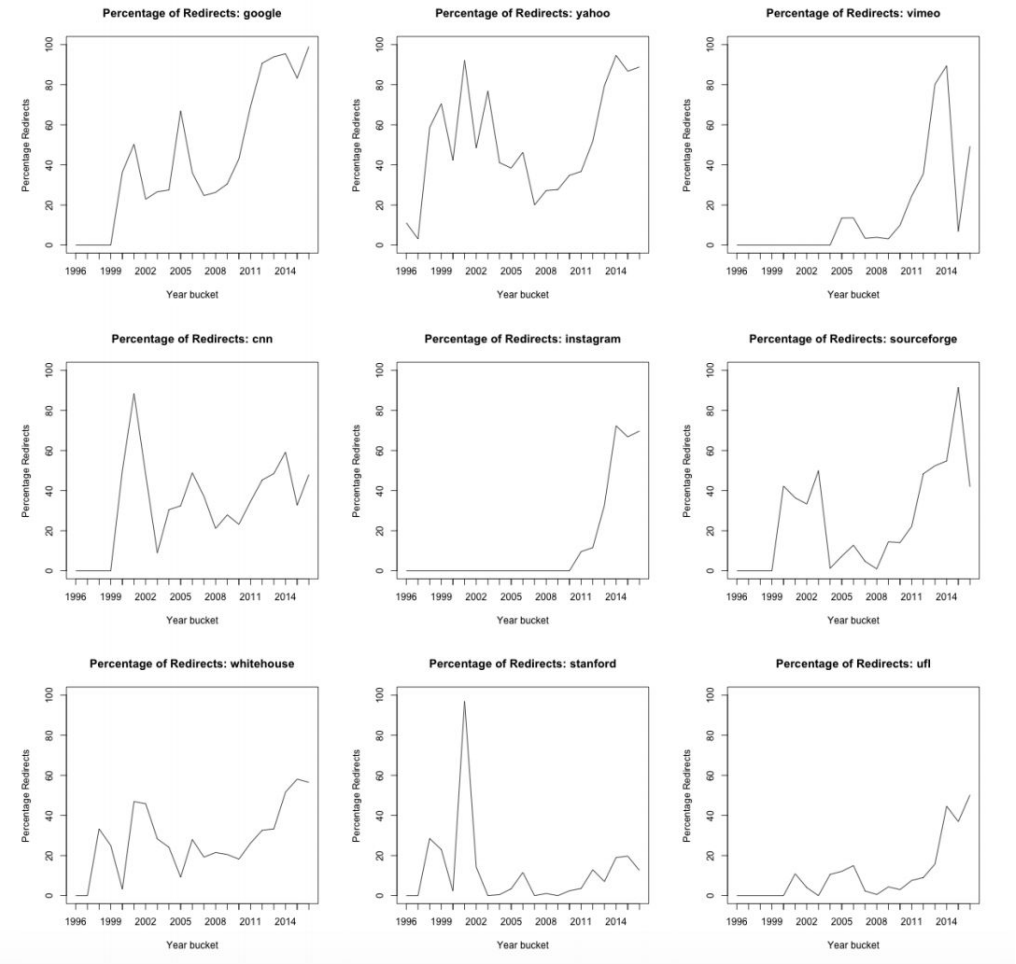
AlSum et al. [1] analyzed memento redirection patterns relating to HTTP redirects to supply the user with the correct memento when a redirect is encountered in the archives. They introduced the notion of "URI stability" to give a quantitative measure of the presence of HTTP 3XX status codes that result when URI-Ms in TimeMaps are dereferenced.

Many URI-Ms are actually redirects

host	% 3XX	% 200	M_{TM}	DI
google	84.89	15.11	695,525	0.178
yahoo	88.16	11.83	418,896	0.134
sourceforge	73.34	26.63	31,408	0.363
instagram	67.32	32.65	55,228	0.485
vimeo	57.04	42.94	199,262	0.752
cnn	49.97	50.01	87,148	1.001
wikipedia	44.62	55.19	25,973	1.240
whitehouse	44.57	55.24	26,006	1.243

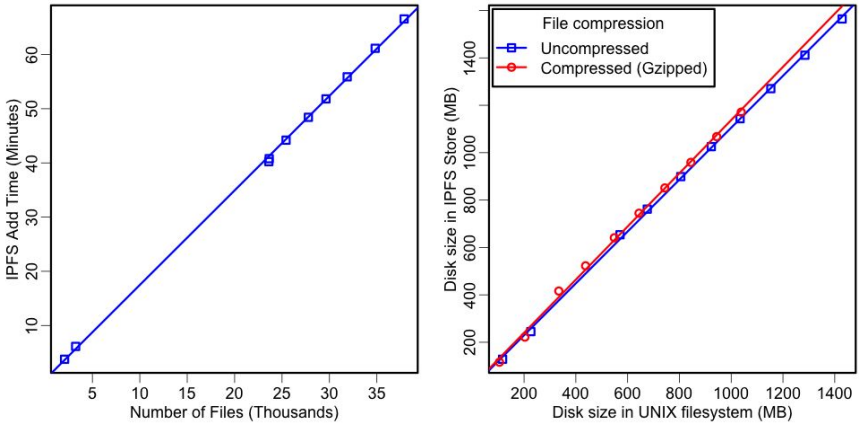
host	% 3XX	% 200	M_{TM}	DI
stanford	62.14	37.84	19,309	0.609
princeton	60.10	39.88	9,355	0.663
columbia	48.01	51.88	9,882	1.082
harvard	33.91	65.96	7,699	1.948
caltech	33.13	66.86	5,474	2.017
mit	26.57	73.24	6,379	2.763
gatech	26.03	73.94	3,907	2.841
ufl	24.76	75.23	4,927	3.038
vt	23.07	76.92	4,061	3.334
lsu	15.06	84.93	2,974	5.638
nsu	13.82	86.00	1,208	6.233
odu	9.727	90.27	1,727	9.279
tcc	5.429	94.57	884	17.41

Mat Kelly, Lulwah M. Alkwai, Michael L. Nelson, Michele C. Weigle, and Herbert Van de Sompel, “Impact of URI Canonicalization on Memento Count,” Technical Report arXiv:1703.03302, 2017.


Percentage of Redirects: google
Percentage of Redirects: yahoo
Percentage of Redirects: vimeo
Percentage of Redirects: cnn
Percentage of Redirects: instagram
Percentage of Redirects: sourceforge
Percentage of Redirects: whitehouse
Percentage of Redirects: stanford
Percentage of Redirects: ufl
Percentage Redirects
Year bucket
0
20
40
60
80
100
1996
1999
2002
2005
2008
2011
2014

InterPlanetary Wayback (ipwb)

- Personal archives are more resilient when propagated.
- How much does it cost to have resilient personal archives?

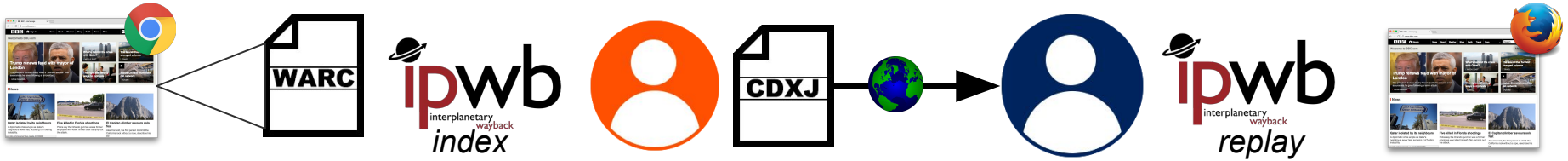


ipwb

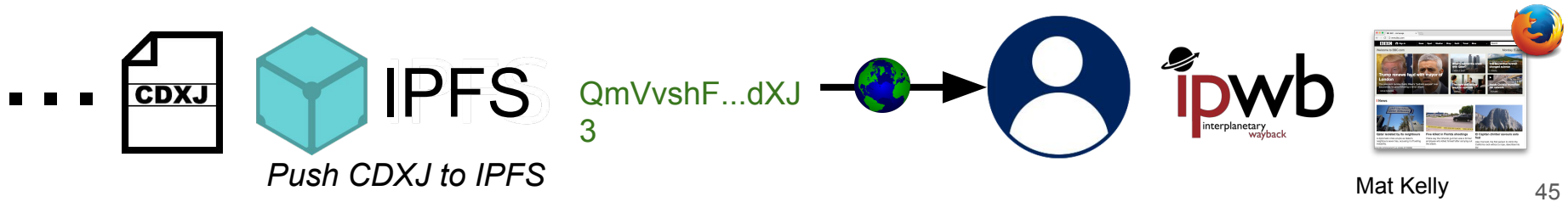
- Persistence of archived web data dependent on resilience of organization and availability of data
- Remove massive redundancy in web archive files of exact duplicate content
- Determine feasibility of pushing WARCs into IPFS

ipwb Base Dynamics

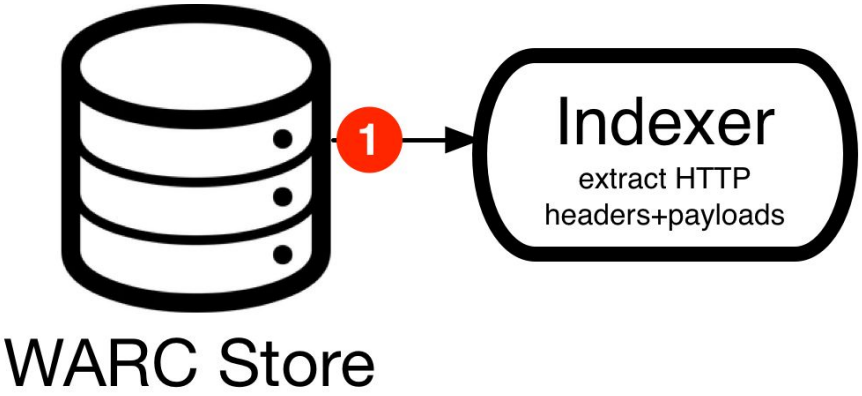
- IPWB CDXJs may be transferred for our users' replay

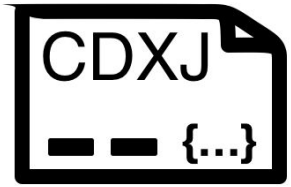


- CDXJ-by-hash recursive fetch/replay
  - Share hash of CDXJ then `$ ipwb replay hash` to replicate experience



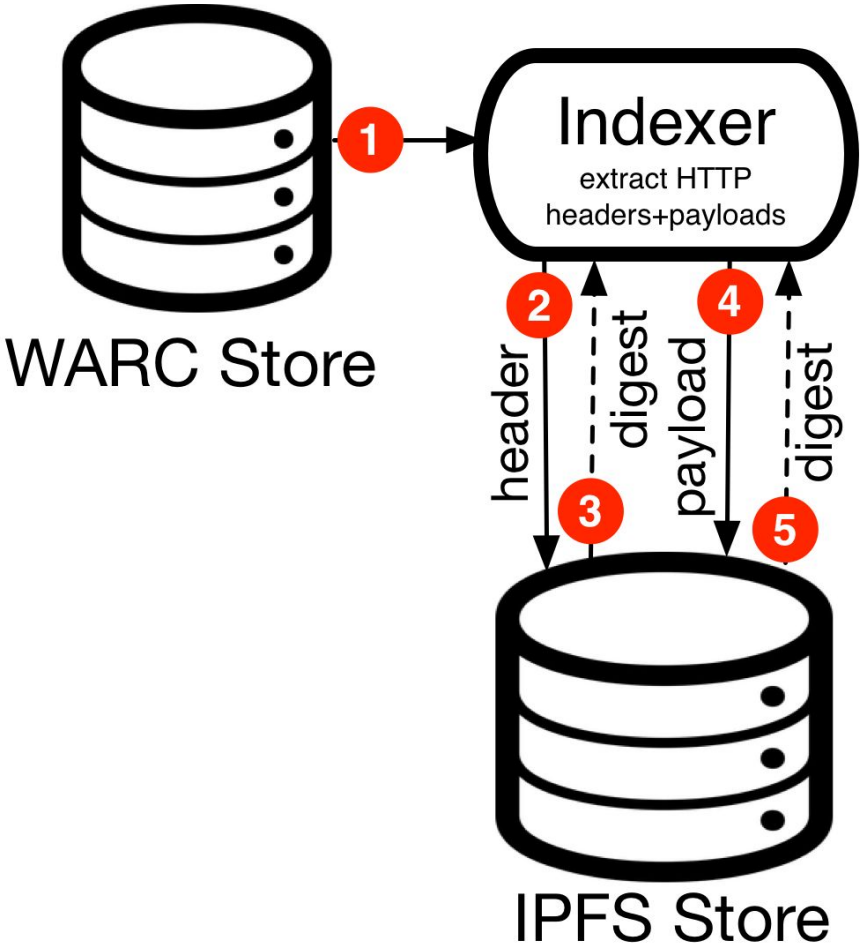
Push CDXJ to IPFS

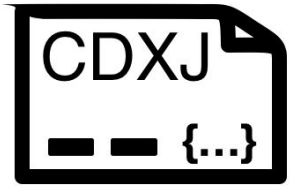

1
Indexer
extract HTTP
headers+payloads
WARC Store


CDXJ
{...}

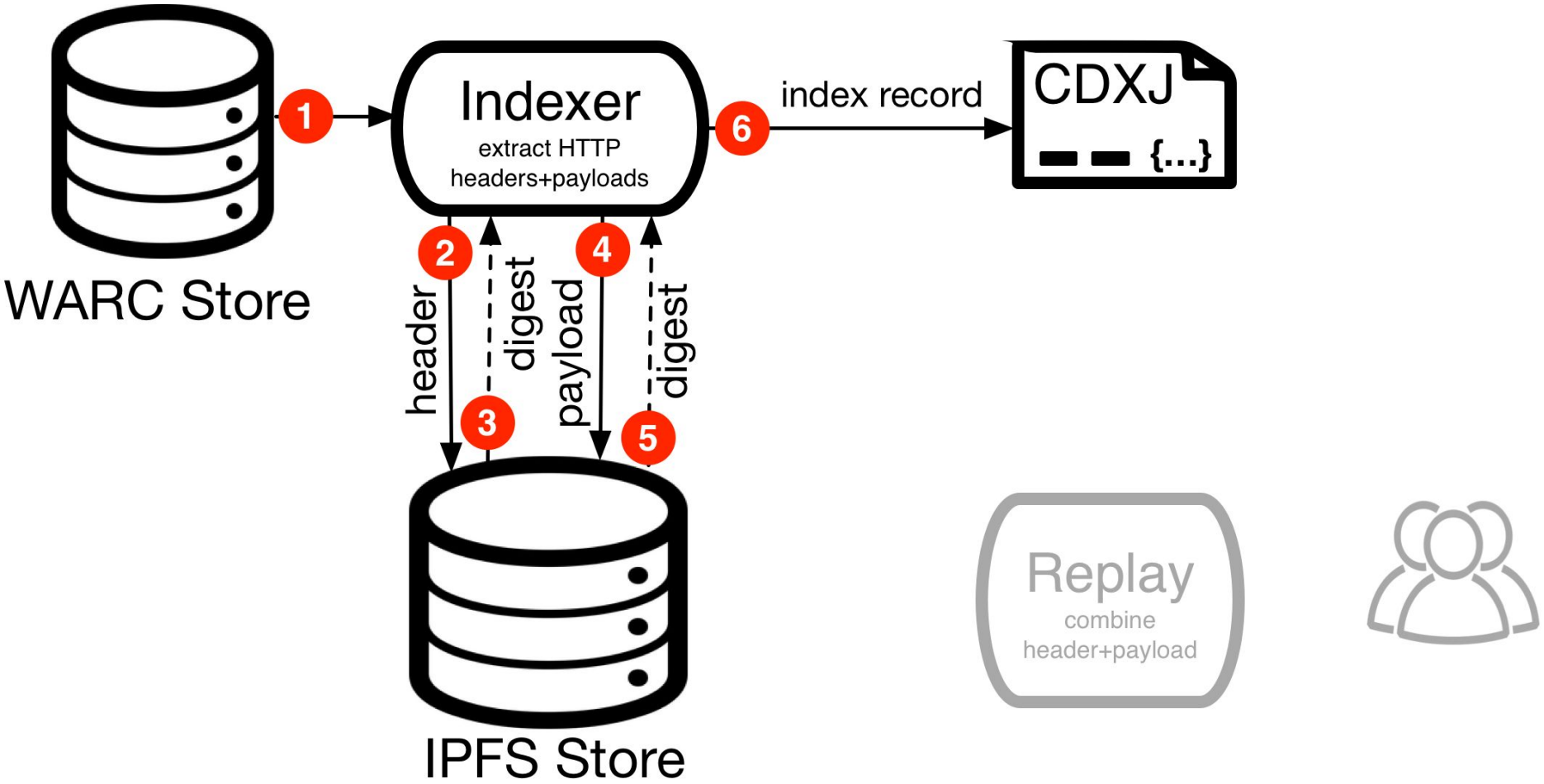

IPFS Store


Replay
combine
header+payload


Indexer
extract HTTP
headers+payloads
1
2
3
4
5
header
digest
payload
digest
WARC Store
IPFS Store


CDXJ
{...}

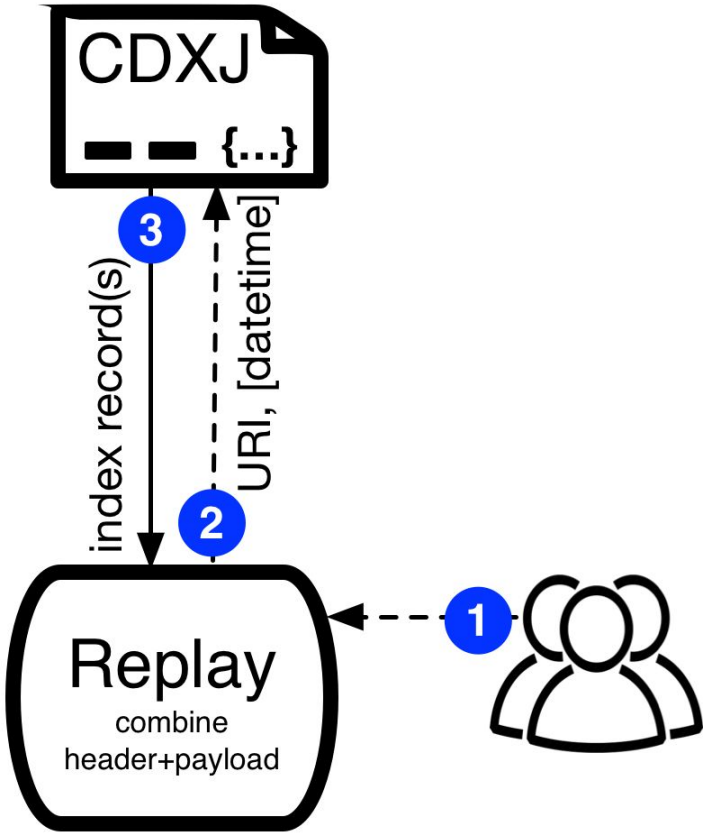

Replay
combine
header+payload

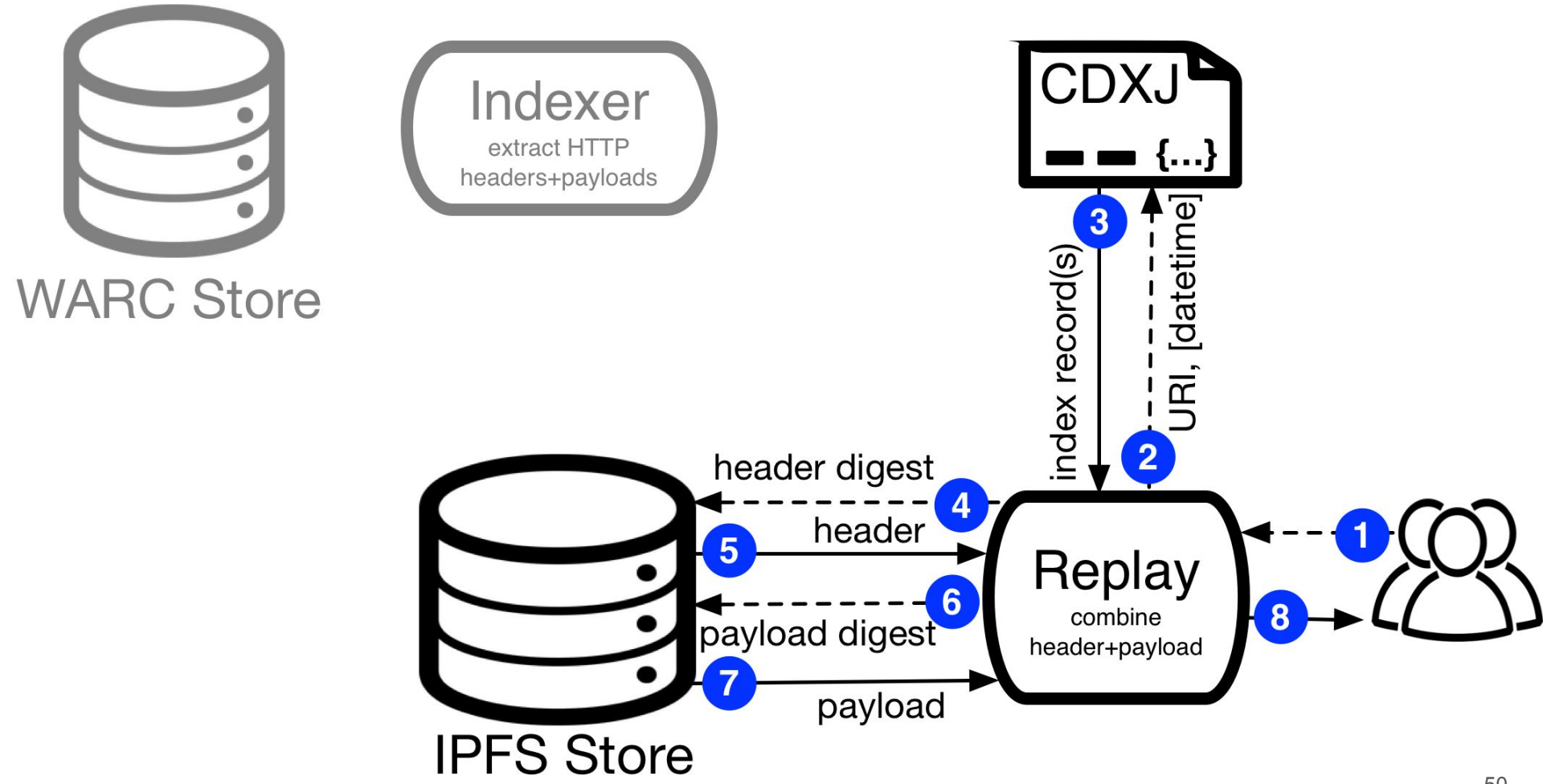

WARC Store
1
Indexer
extract HTTP
headers+payloads
6
index record
CDXJ
{...}
2
header
3
digest
4
payload
5
digest
IPFS Store
Replay
combine
header+payload

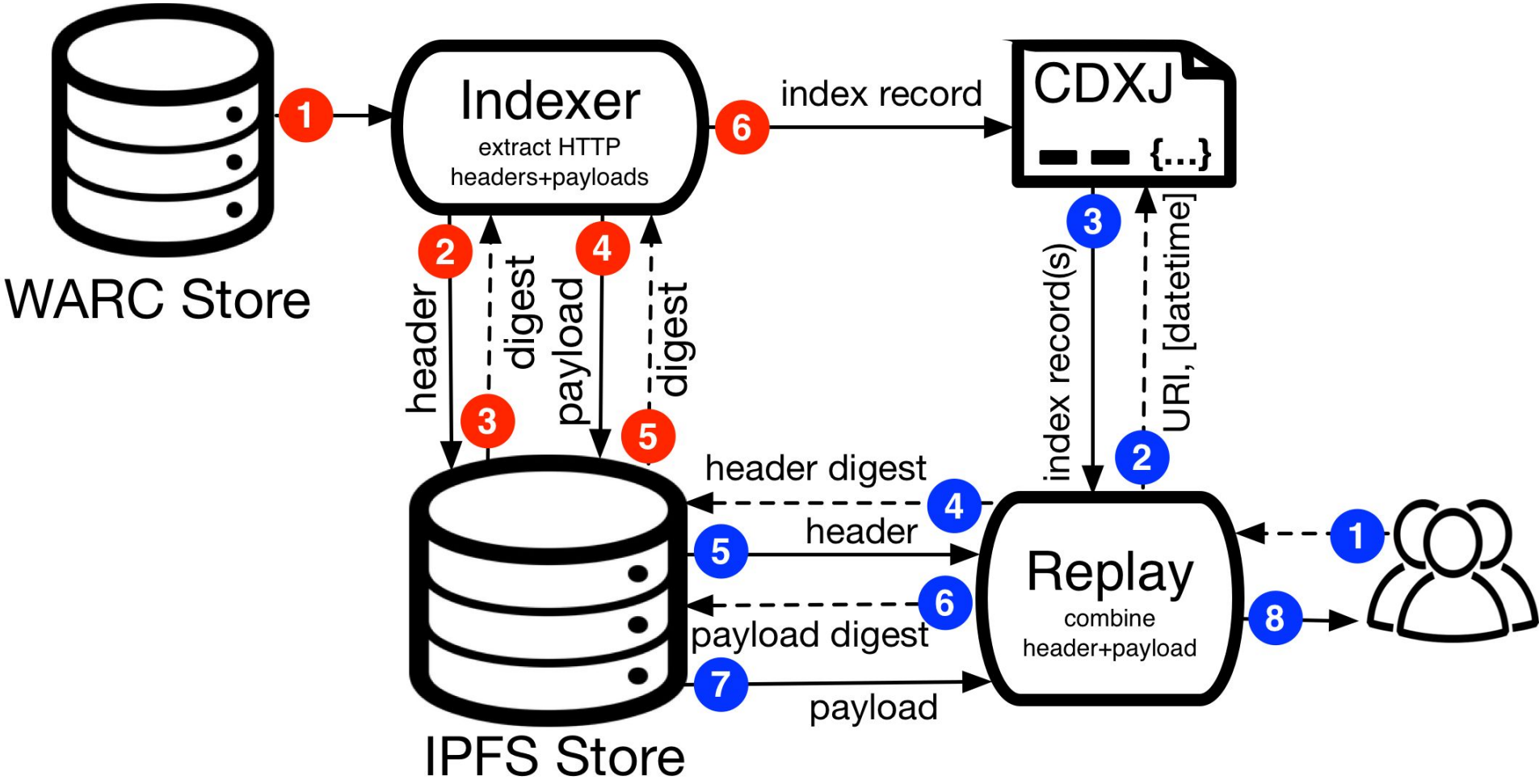

WARC Store


Indexer
extract HTTP
headers+payloads

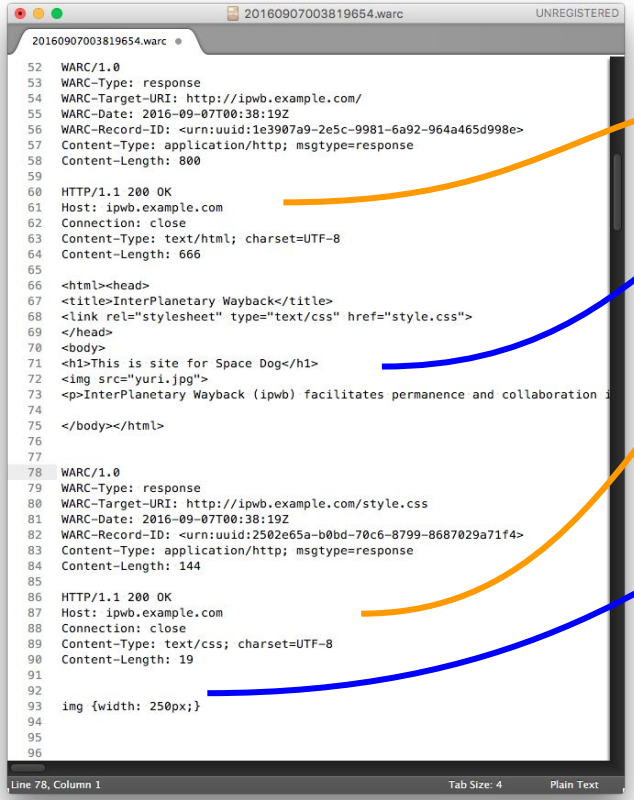

IPFS Store


CDXJ
{...}
3
index record(s)
URI, [datetime]
2
Replay
combine
header+payload
1


WARC Store
Indexer
extract HTTP
headers+payloads
CDXJ
{...}
3
index record(s)
URI, [datetime]
2
header digest
4
header
5
6
payload digest
7
payload
IPFS Store
Replay
combine
header+payload
1
8


WARC Store
1
Indexer
extract HTTP
headers+payloads
6
index record
CDXJ
{...}
2
header
3
digest
4
payload
5
digest
3
index record(s)
2
URI, [datetime]
header digest
4
5
header
6
payload digest
7
payload
IPFS Store
Replay
combine
header+payload
1
8

WARC Record extraction to CDXJ



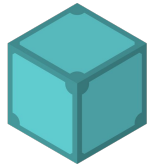
ipwb.example.com)/ 20160905022013 {"locator": "urn:ipfs/QmcN9eWwRF73dZj5BgT4x8jeEcFrxurX1hot8QwCbMi9PB/Qmczh9YnB4H1ptPeqxcaTZA4aMmuNUswTLTWzXntvbp9sL", "mime_type": "text/html", "status_code": "200"}
ipwb.example.com)/style.css 20160905022013 {"locator": "urn:ipfs/QmU1k71bT6ibZBSdxBL35cQXwovTih8cTB4CXfrjyMfZxE/QmbvUAo9U31wSdvARjvbPeVBTAwCjN1kyPhQ4ho3n8TAZo", "mime_type": "text/css", "status_code": "200"}
ipwb.example.com)/ipwb.png 20160905022013 {"locator": "urn:ipfs/QmTjfMxFGvbP4nwFoq3tNYDPW6gC99i5njrqsXSw6QRvHa/QmYMKZbnk53kuPJirahJHGevCCy2afLyePRdX38TukFUwd", "mime_type": "image/png", "status_code": "200"}
ipwb.example.com)/fileduration.png 20160905022013 {"locator": "urn:ipfs/QmaCj6LNngxwqxaLmfp1xCyxcwDt2Uzqf8gCG6bVyQppYC/QmdgtMcGprTF8bqv7ytgMwtoi5BhRxfuvBjD6Vj2U7ohz1", "mime_type": "image/png", "status_code": "200"}
ipwb.example.com)/filesize.png 20160905022013 {"locator": "urn:ipfs/QmNPjrSVY31oGDooMiA18ZDNHfkLnEg3j5gRj1dFdrqmS4/Qmb4heB8PU58nkWt6w5tBgMfpeLTKuU7iuxg9tFdoPsF1B", "mime_type": "image/png", "status_code": "200"}

IPFS multihashes in IPWB CDXJs

com,example)/index.html 20170301192639 {"locator": "urn:ipfs/QmPdyY6Pm66iWtGpTc7PqK11hvsnYSKMVL57G69RiNjGcm/QmNZ6mKSSAXAmXEocQj5gT4y4kdcr5D2C173ubWJ6PSKEZ", "mime_type": "text/html", "status_code": "200"}
com,example)/images/frog.png 20170301192639 {"locator": "urn:ipfs/QmUeko8zM7Xanwz6F9GtRH4rLAi4Poj3EMECGsci3BRQfs/QmPhMnX74cwqx2xgj9d3N3gTra8CzafXwSbUwU8xagMfqR", "mime_type": "image/png", "status_code": "200"}

Content Addressing

http://foo.com/spaceDog.jpg

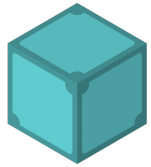


IPFS

QmZAD4xeeNeYF3TmwWgBXypLKTiCGwGRMXHW7MtheWKtw4

===

http://example.org/yuri.jpg



QmZAD4xeeNeYF3TmwWgBXypLKTiCGwGRMXHW7MtheWKtw4

IPFS

$ ipfs cat QmZAD4xeeNeYF3TmwWgBXypLKTiCGwGRMXHW7MtheWKtw4 > doge.jpg

Methodology - IPWB WARC indexing

- `warc-response` record body extracted into temp files
  - HTTP header and entity body (payload) separated
  - Response metadata (e.g., datetime) retained
- temp files pushed into IPFS via locally running daemon
  - Two IPFS hashes (for header and payload) returned
- CDXJ record created representing `warc-response` contents
  - Contains URI-R, archived HTTP status, encoded IPFS hashes

Methodology - Replaying Archives

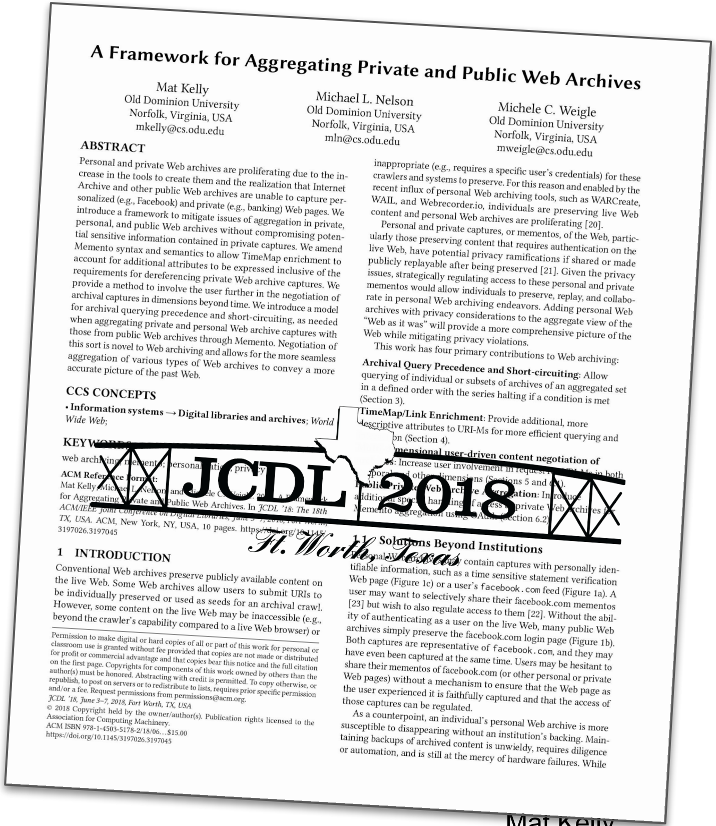
- Extension of pywb API to read CDXJ files
- On encountering IPFS URN, fetch `warc-response` temp files from IPFS using local daemon
  - This may occur on a separate machine using a separate daemon
- With WARC contents fetched, replay contents using pywb where the locator value in the CDXJ is used to dereference the temp files pulled from IPFS

CDXJ in ipwb

```
SURT_URI DATETIME {
        "id": "WARC-Record-ID",
        "url": "ORIGINAL_URI",
        "status": "3-DIGIT_HTTP_STATUS",
        "mime": "Content-Type",
        "locator": "urn:ipfs/HEADER_DIGEST/PAYLOAD_DIGEST"
}
```

A Framework for Aggregating Private and Public Web Archives

- Mitigate outstanding issues in Web archiving beyond public scope.
- High-level of dissertation topic
- Introduced the “Mementity Framework”



A Framework for Aggregating Private and Public Web Archives

Mat Kelly
Old Dominion University
Norfolk, Virginia, USA
mkelly@cs.odu.edu

Michael L. Nelson
Old Dominion University
Norfolk, Virginia, USA
mln@cs.odu.edu

Michele C. Weigle
Old Dominion University
Norfolk, Virginia, USA
mweigle@cs.odu.edu

ABSTRACT

Personal and private Web archives are proliferating due to the increase in the tools to create them and the realization that Internet Archive and other public Web archives are unable to capture personalized (e.g., Facebook) and private (e.g., banking) Web pages. We introduce a framework to mitigate issues of aggregation in private, personal, and public Web archives without compromising potential sensitive information contained in private captures. We amend Memento syntax and semantics to allow TimeMap enrichment to account for additional attributes to be expressed inclusive of the requirements for dereferencing private Web archive captures. We provide a method to involve the user further in the negotiation of archival captures in dimensions beyond time. We introduce a model for archival querying precedence and short-circuiting, as needed when aggregating private and personal Web archive captures with those from public Web archives through Memento. Negotiation of this sort is novel to Web archiving and allows for the more seamless aggregation of various types of Web archives to convey a more accurate picture of the past Web.

CCS CONCEPTS

• Information systems → Digital libraries and archives; World Wide Web;

KEYWORDS

web archiving, memento, personal, privacy

ACM Reference Format:

Mat Kelly, Michael L. Nelson, and Michele C. Weigle. 2018. A Framework for Aggregating Private and Public Web Archives. In JCDL '18: The 18th ACM/IEEE Joint Conference on Digital Libraries, June 3–7, 2018, Fort Worth, TX, USA. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3197026.3197045

1 INTRODUCTION

Conventional Web archives preserve publicly available content on the live Web. Some Web archives allow users to submit URIs to be individually preserved or used as seeds for an archival crawl. However, some content on the live Web may be inaccessible (e.g., beyond the crawler's capability compared to a live Web browser) or

inappropriate (e.g., requires a specific user's credentials) for these crawlers and systems to preserve. For this reason and enabled by the recent influx of personal Web archiving tools, such as WARCreate, WAIL, and Webrecorder.io, individuals are preserving live Web content and personal Web archives are proliferating [20].

Personal and private captures, or mementos, of the Web, particularly those preserving content that requires authentication on the live Web, have potential privacy ramifications if shared or made publicly replayable after being preserved [21]. Given the privacy issues, strategically regulating access to these personal and private mementos would allow individuals to preserve, replay, and collaborate in personal Web archiving endeavors. Adding personal Web archives with privacy considerations to the aggregate view of the “Web as it was” will provide a more comprehensive picture of the Web while mitigating privacy violations.

This work has four primary contributions to Web archiving:

Archival Query Precedence and Short-circuiting: Allow querying of individual or subsets of archives of an aggregated set in a defined order with the series halting if a condition is met (Section 3).

TimeMap/Link Enrichment: Provide additional, more descriptive attributes to URI-Ms for more efficient querying and ... (Section 4).

... user-driven content negotiation of ...: Increase user involvement in request ... in both ... other dimensions (Sections 5 and 6).

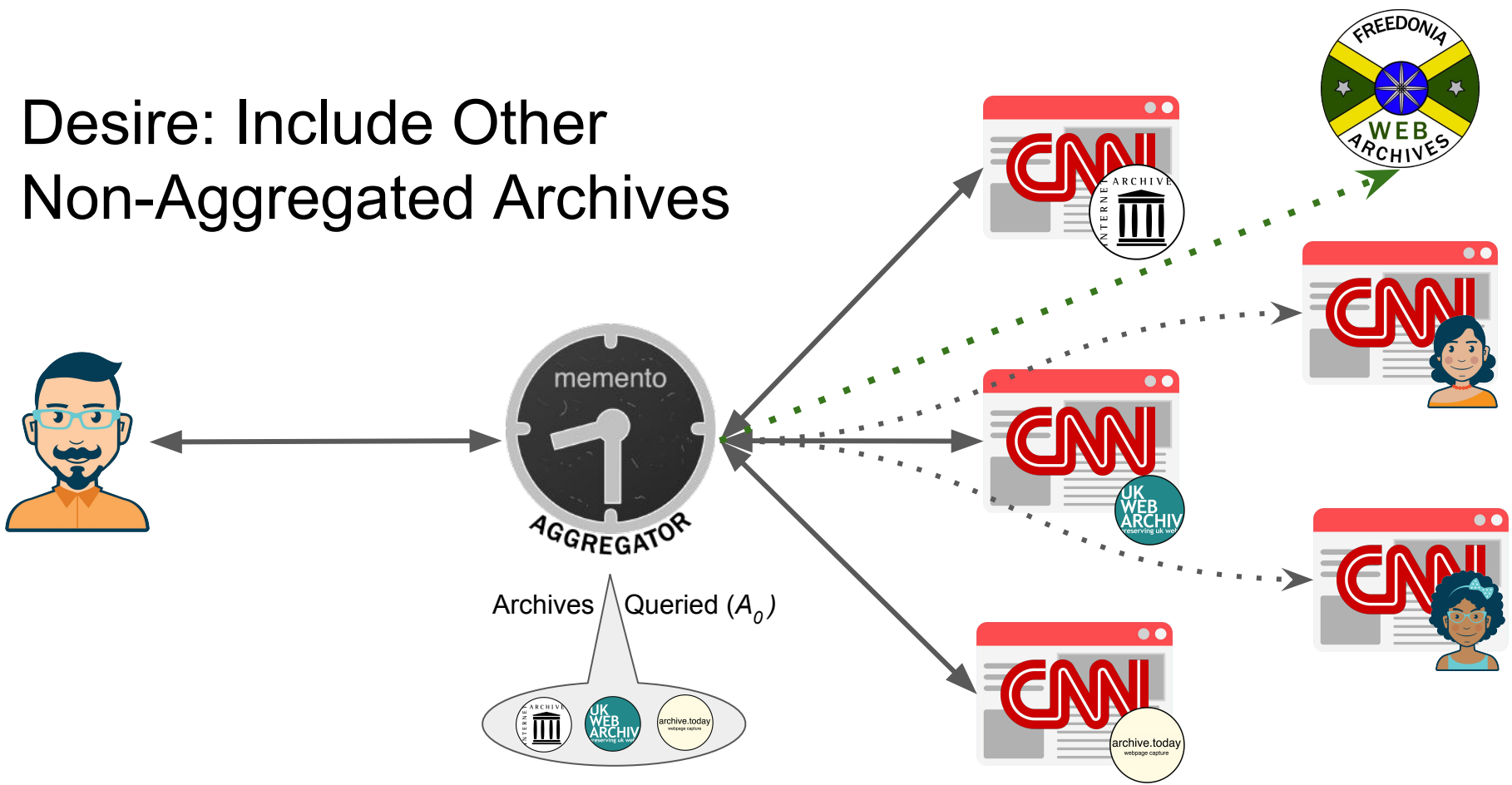
... Private Web Archive Aggregation: Introduce additional special handling ... private Web archives ... Memento aggregation using OAuth (Section 6.2).

1.1 Solutions Beyond Institutions

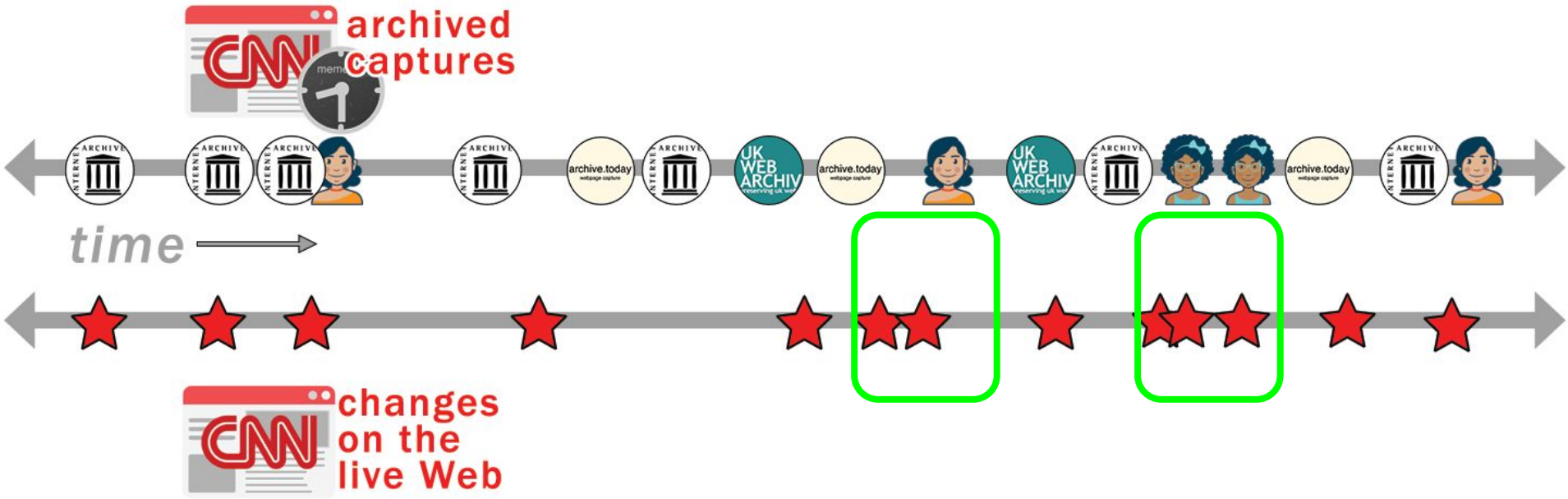
Personal Web archives may contain captures with personally identifiable information, such as a time sensitive statement verification Web page (Figure 1c) or a user's facebook.com feed (Figure 1a). A user may want to selectively share their facebook.com mementos [23] but wish to also regulate access to them [22]. Without the ability of authenticating as a user on the live Web, many public Web archives simply preserve the facebook.com login page (Figure 1b). Both captures are representative of facebook.com, and they may have even been captured at the same time. Users may be hesitant to share their mementos of facebook.com (or other personal or private Web pages) without a mechanism to ensure that the Web page as the user experienced it is faithfully captured and that the access of those captures can be regulated.

As a counterpoint, an individual's personal Web archive is more susceptible to disappearing without an institution's backing. Maintaining backups of archived content is unwieldy, requires diligence or automation, and is still at the mercy of hardware failures. While

Desire: Include Other Non-Aggregated Archives

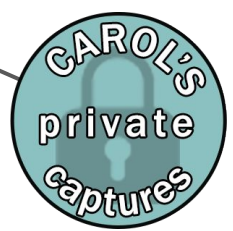


Archiving More Archives Provides a Better Picture of the Web



Query Precedence & Short-Circuiting

- **May give priority to archive relevancy.**
- **Series halt when threshold met.**



1

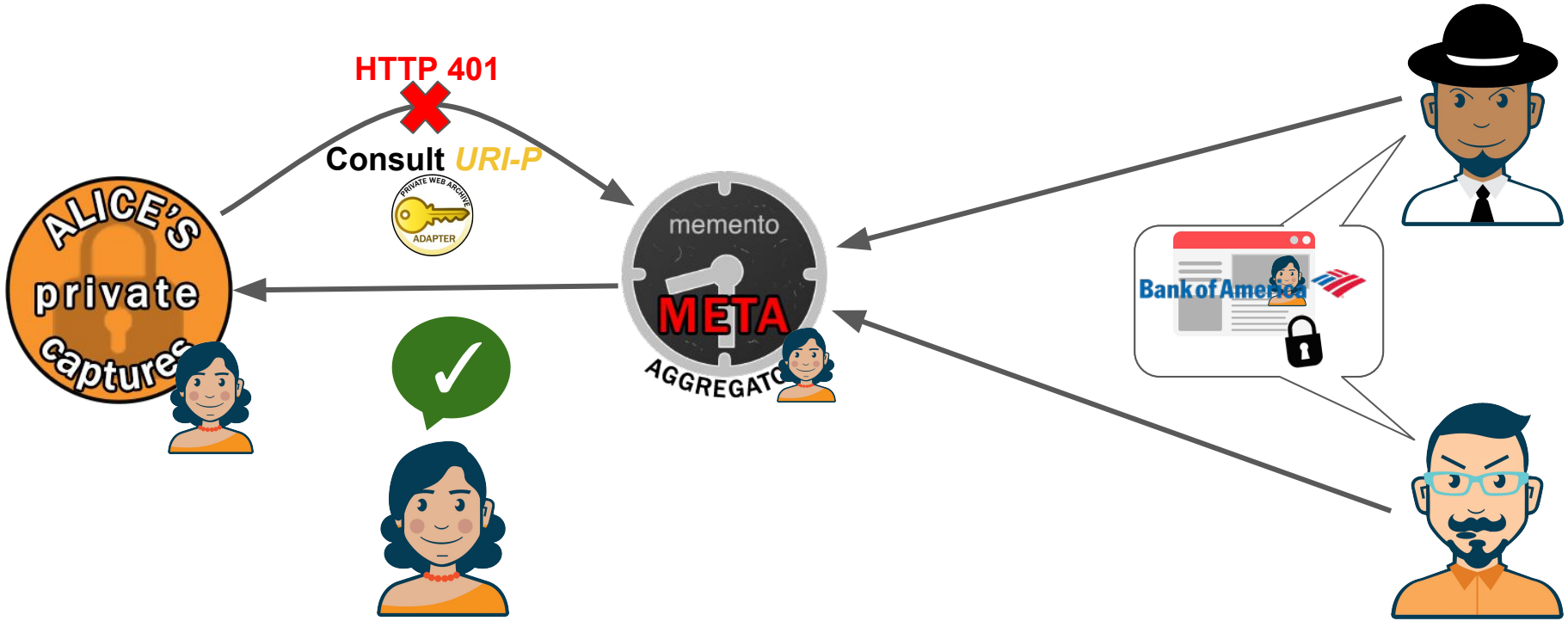
1

2

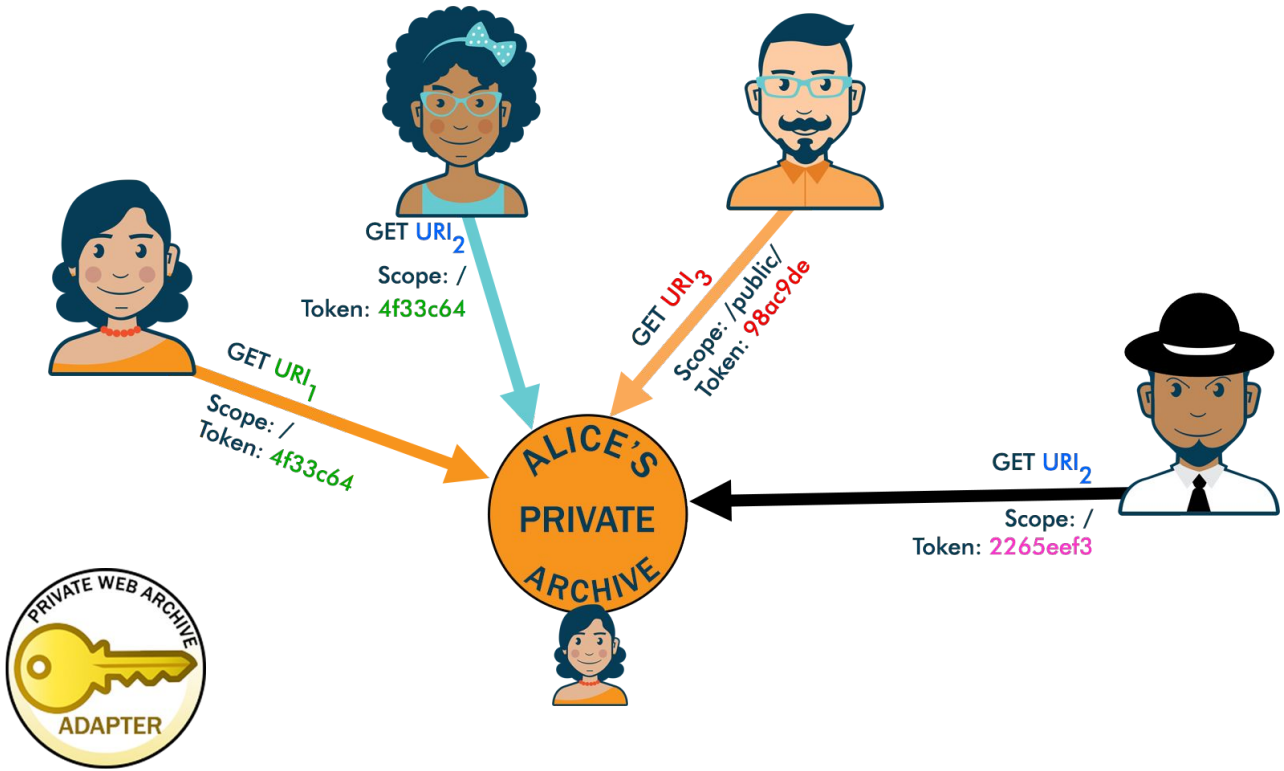
2

"Check private archives first. **Iff** you find no captures, only *then* check the public archives.

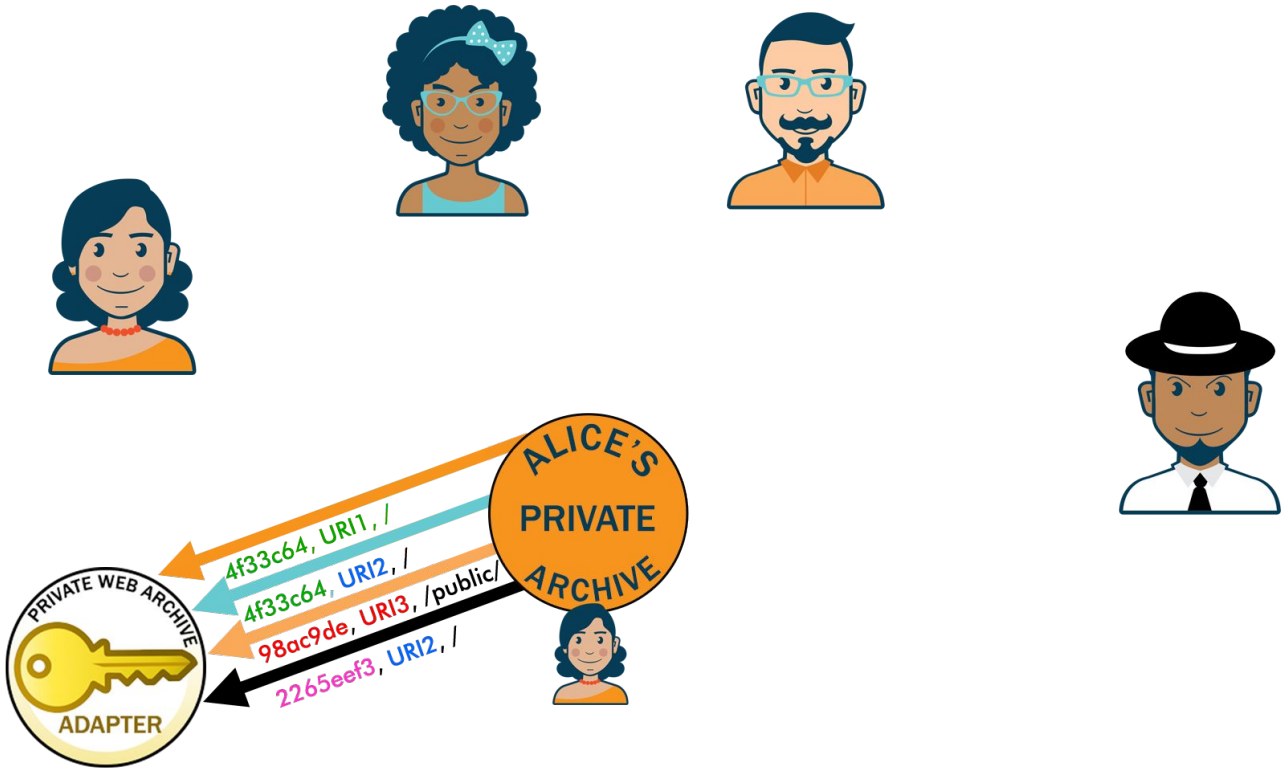
Aggregation with Access Regulation



OAuth2-based tokenization patterns



...with offloading of the procedure from the archives



Evaluation Through Implementation

extend

Extend for client-side archival specification

extend

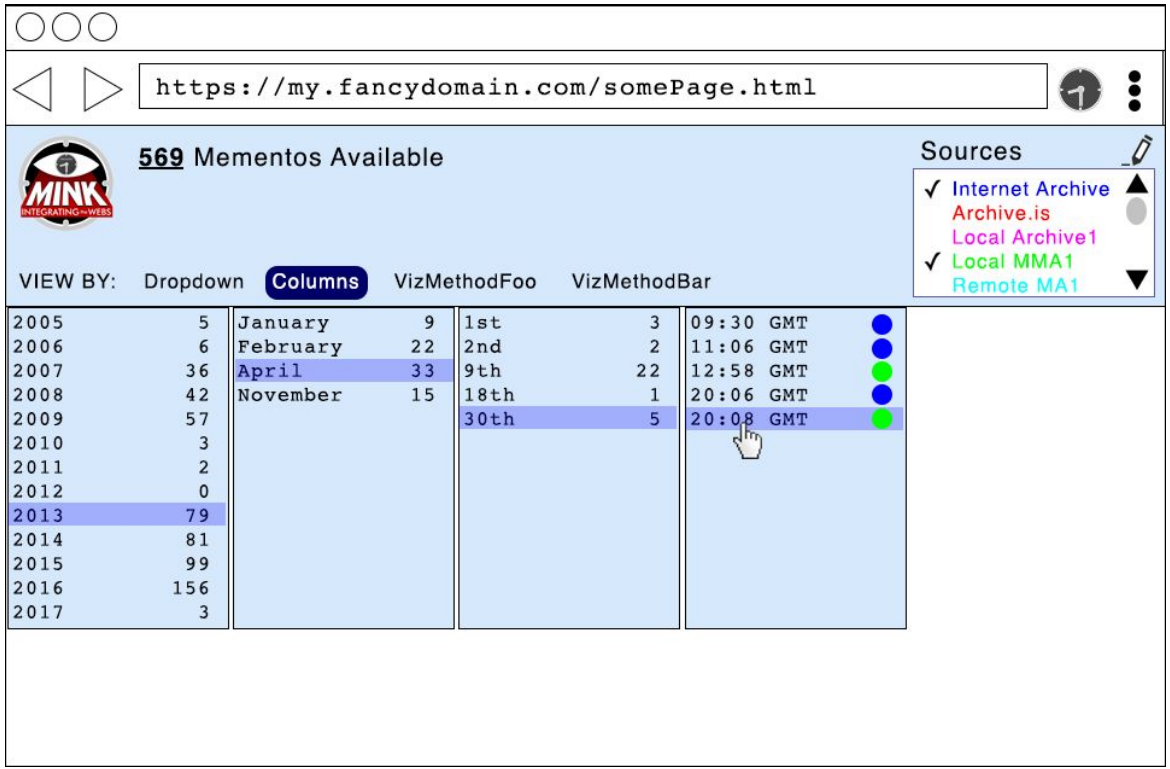
Exhibit features of an MMA

create

Regulate access to Private Web archives

create

Facilitate archival negotiation in more dimensions

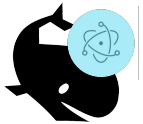


On Tool Building and Evaluation of the Archived Web

Open Source Web Archiving Tools

GitHub machachawk1/warcreate

machachawk1/wail



n0tan3rd/wail

machachawk1/mink

oduwsdl/ipwb

More details of studies:

- Measuring the Impact of Missing Resources (Conf, Journal)
- Impact of URI Canonicalization (Conf, arXiv)
- Archivability Over Time (Conf, arXiv)
- Impact of JS on Archivability (Journal)
- Personalization in Web Archives (article)

Slides available at http://matkelly.com/psu

Mat Kelly
https://www.cs.odu.edu/~mkelly/
February 13, 2019