

# A Framework for Aggregating Private and Public Web Archives

Mat Kelly

Old Dominion University  
Web Science & Digital Libraries Research Group

Department of Computer Science  
Norfolk, Virginia USA

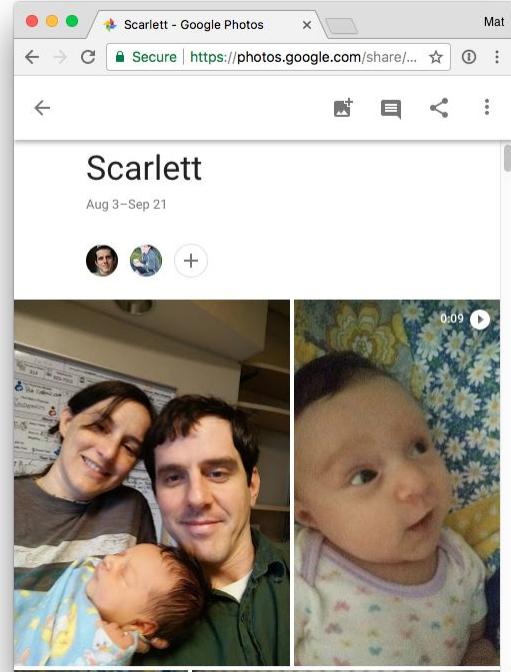
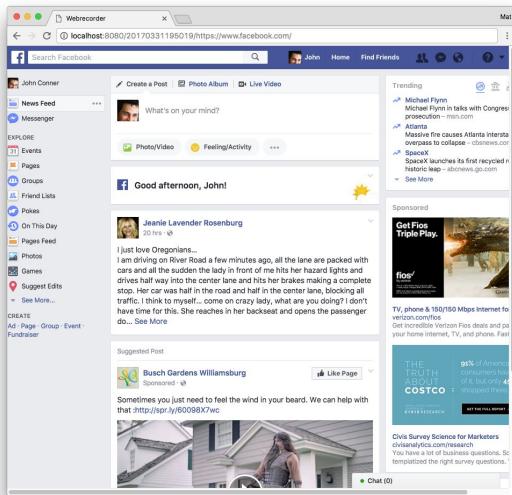
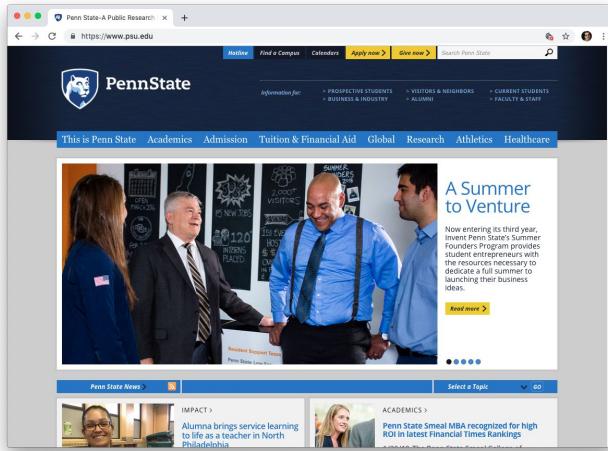
[mkelly@cs.odu.edu](mailto:mkelly@cs.odu.edu)



Seminar, Penn State University  
February 14, 2019



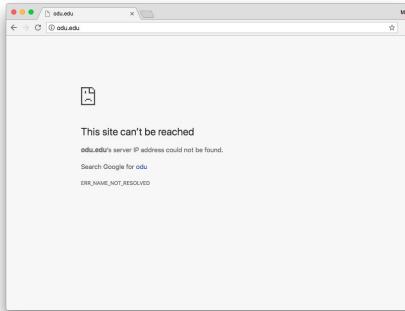
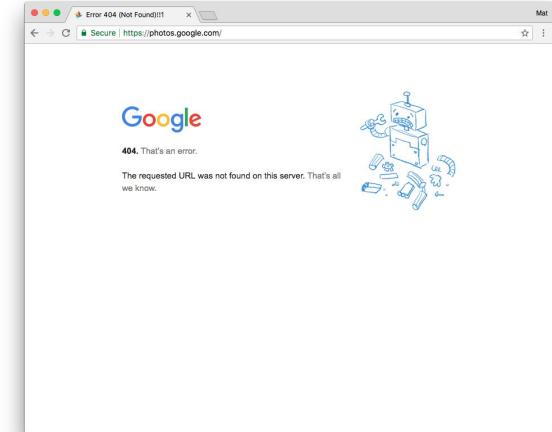
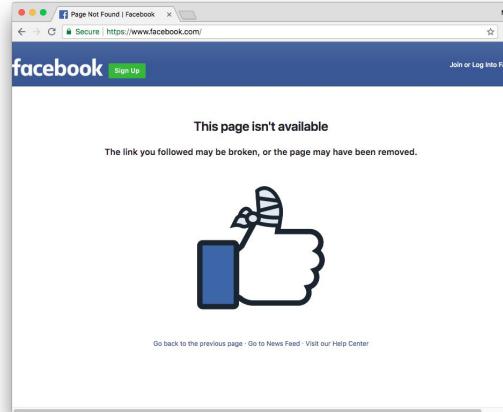
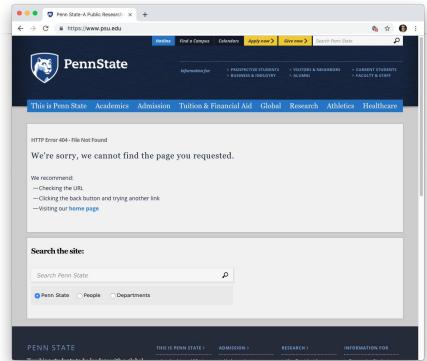
# The Web



A Framework for Aggregating Public and Private Web Archives  
February 14, 2019  
Mat Kelly



# The Web is Ephemeral

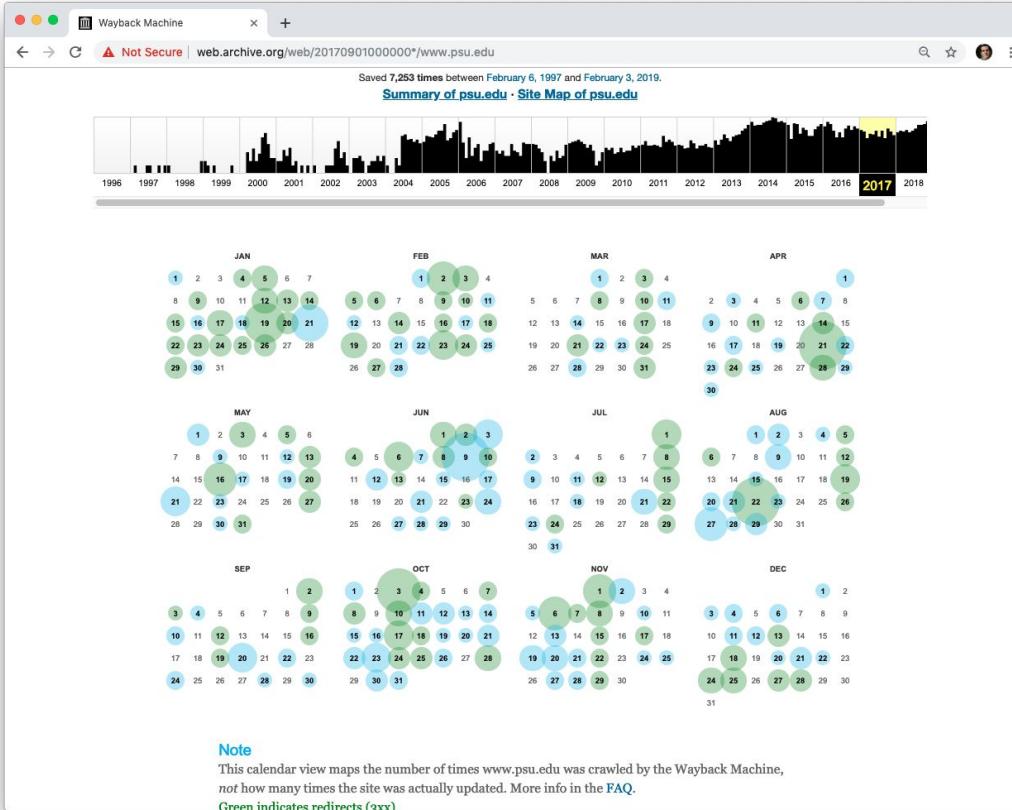


# Web Archives to the Rescue: Typical Access

The screenshot shows the Wayback Machine homepage. At the top, there's a navigation bar with links for ABOUT, CONTACT, BLOG, PROJECTS, HELP, DONATE, JOBS, VOLUNTEER, and PEOPLE. A search bar is prominently displayed. Below the header, the Internet Archive logo and the Wayback Machine logo are visible. A banner states "Explore more than 347 billion web pages saved over time". A search form contains the URL "www.psu.edu". To the right of the search form are social media sharing buttons for Facebook and Twitter, and a "Feedback" link. Below the search area, there's a section titled "Find the Wayback Machine useful? DONATE". A row of thumbnail images shows various captured web pages from different points in time. On the left side, there's a "Tools" section with links for "Wayback Machine Availability API", "WordPress Broken Link Checker", and "404 Handler for Webmasters". In the center, there's a "Subscription Service" section with a brief description and a "Save Page Now" button. The "Save Page Now" button has a URL input field containing "https://", a "SAVE PAGE" button, and a note about it being available for sites that allow crawlers. At the bottom, there are links for "FAQ | Contact Us | Beta Site Feedback | Terms of Service (Dec 31, 2014)". The footer contains the Internet Archive logo and a detailed description of the Wayback Machine's mission and other projects like Open Library and archive-it.org.

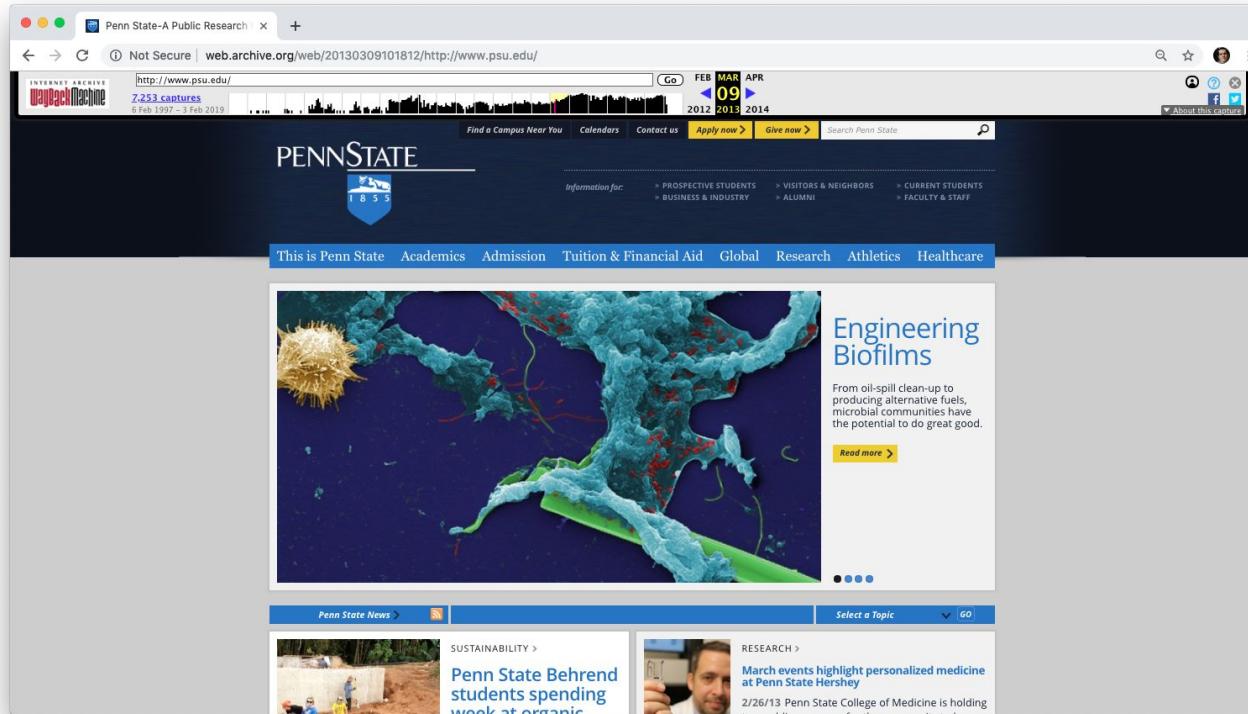
1. Go to `archive.org` in your browser
2. Enter the URL you want to see in the past in the form field
3. Submit your query

# Web Archives to the Rescue: Typical Access



4. Locate the capture on the calendar or histogram view
5. Select the year/capture for the day
6. Repeat until you find the closest date and time

# 7. Finally, view the capture

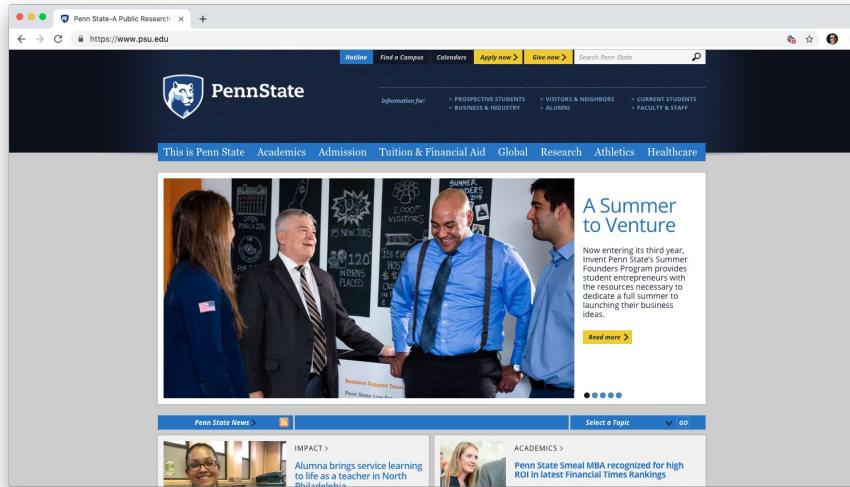


A Framework for Aggregating Public and Private Web Archives  
February 14, 2019  
Mat Kelly

from March 9, 2013



# Web Archiving - Live Web psu.edu



Now

A Framework for Aggregating Public and Private Web Archives  
February 14, 2019  
Mat Kelly



# Web Archiving - Archival Capture

A screenshot of the Penn State website homepage. The header features the Penn State logo and navigation links for Home, Find a Campus, Calendars, Apply now, Give now, and Search Penn State. Below the header, there's a main banner with four people and the text "A Summer to Venture". A sidebar on the right contains a news item about the Invent Penn State's Summer Founders Program. At the bottom, there are news cards for Penn State News, Impact, Academics, and Research.



A screenshot of the Penn State website as it appeared on March 9, 2013, according to the web archive interface. The header and navigation bar are identical to the live site. The main banner and sidebar content are preserved. The news cards at the bottom have been replaced by a large, detailed image of microorganisms under a microscope, with the text "Engineering Biofilms" overlaid. A timestamp in the top right corner indicates the capture date.

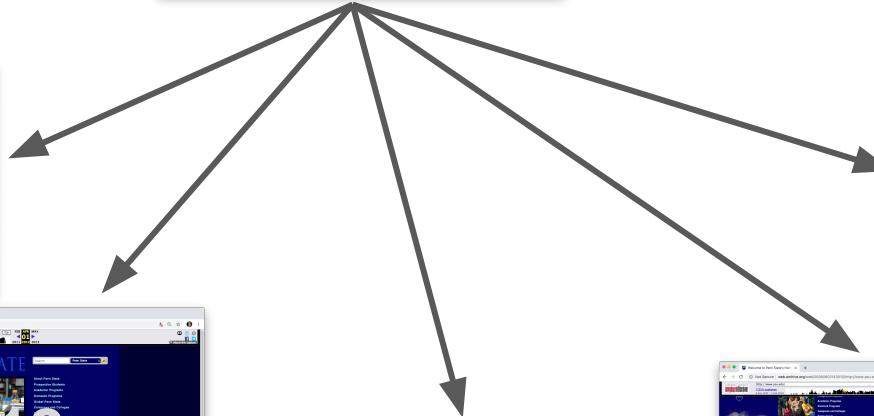
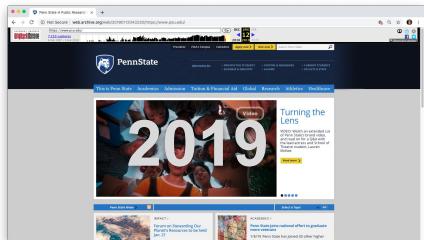
Now

March 9, 2013

# Web Archiving



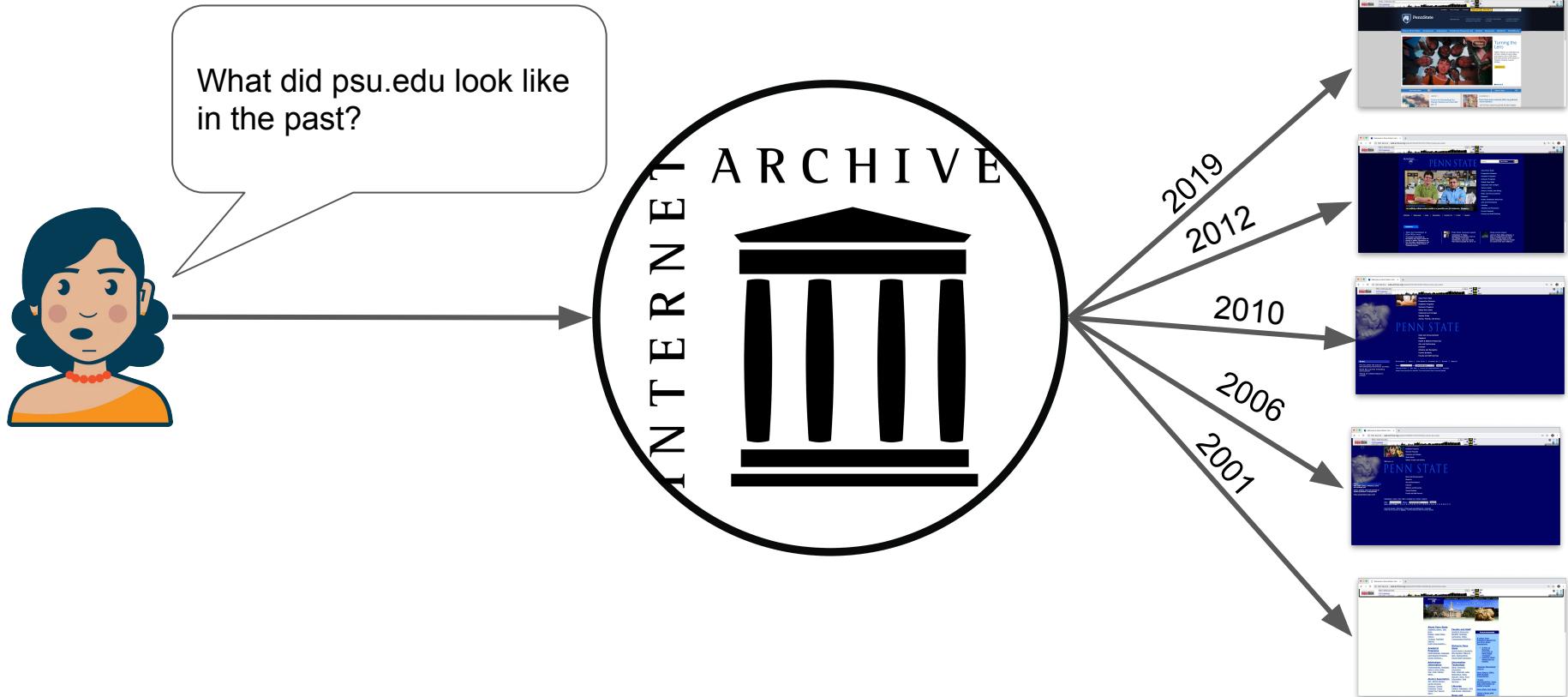
Associate live Web URIs



With their archived representations



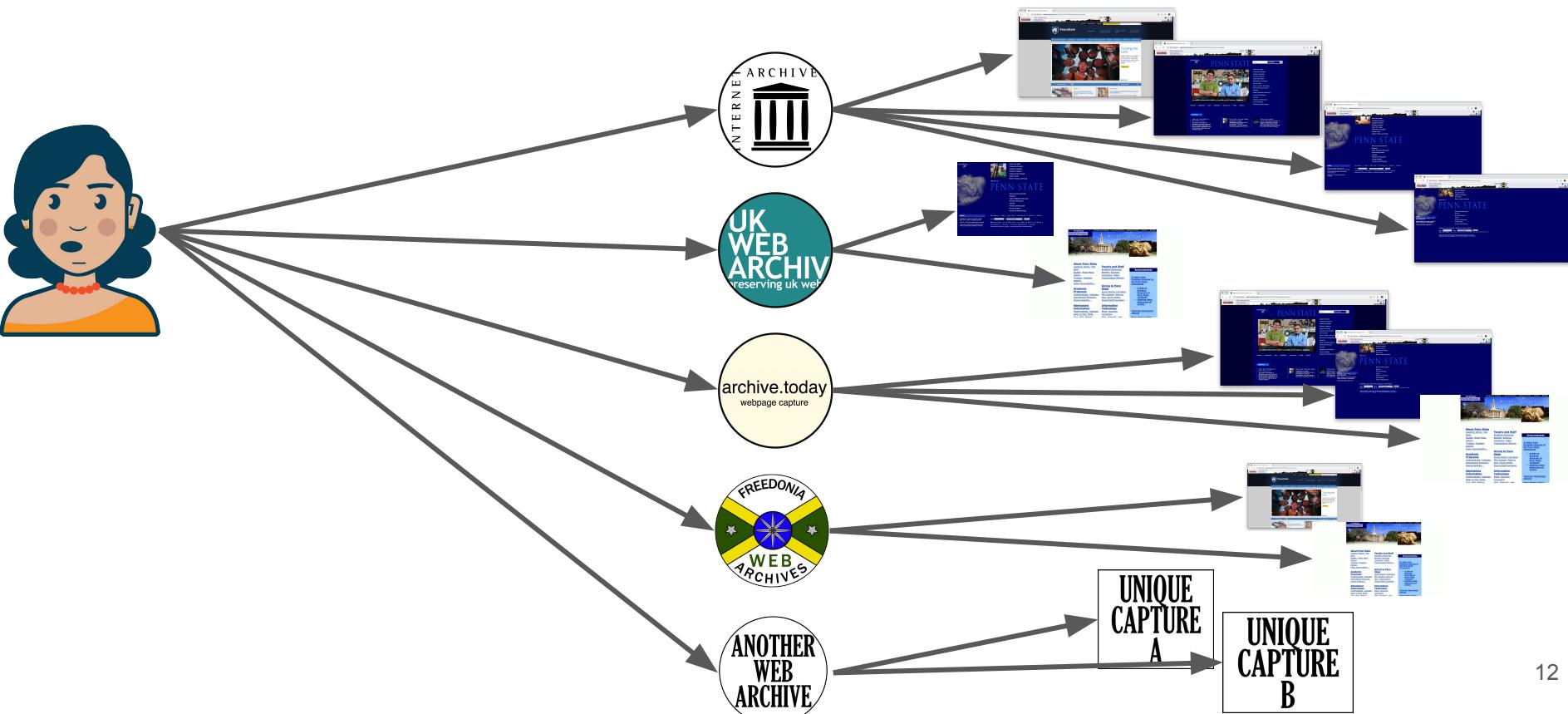
# Web Archives provides access to the Web that was



# Multiple archival efforts (3 of many)



# More archives produces a more comprehensive picture



# Even then, not everything is preserved

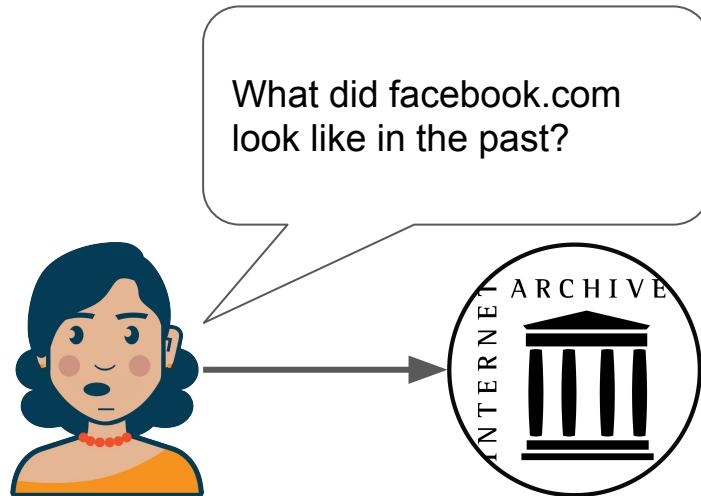
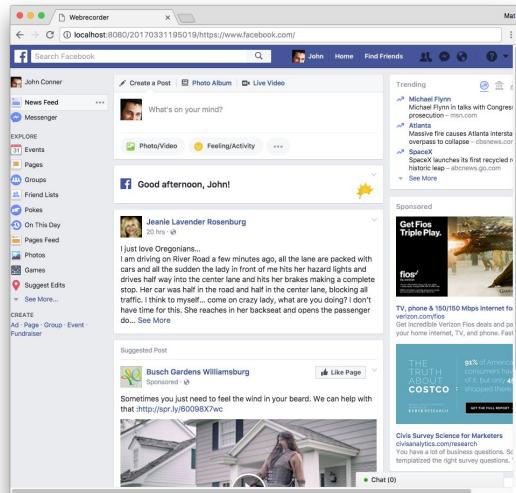


What did obscuresite.com look like in the past?

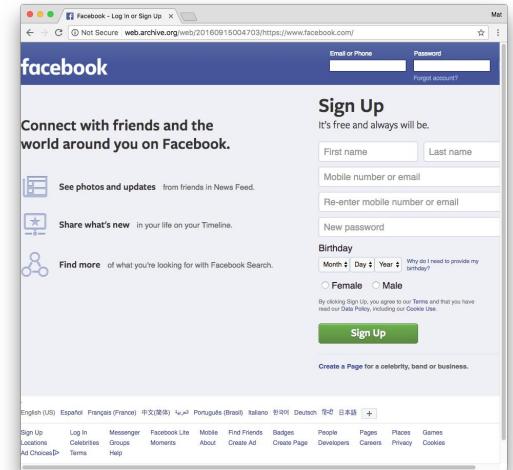


0 captures for  
obscuresite.com

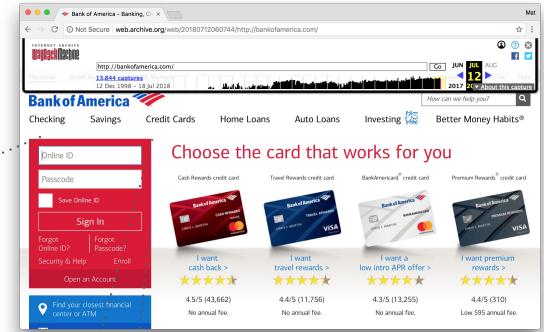
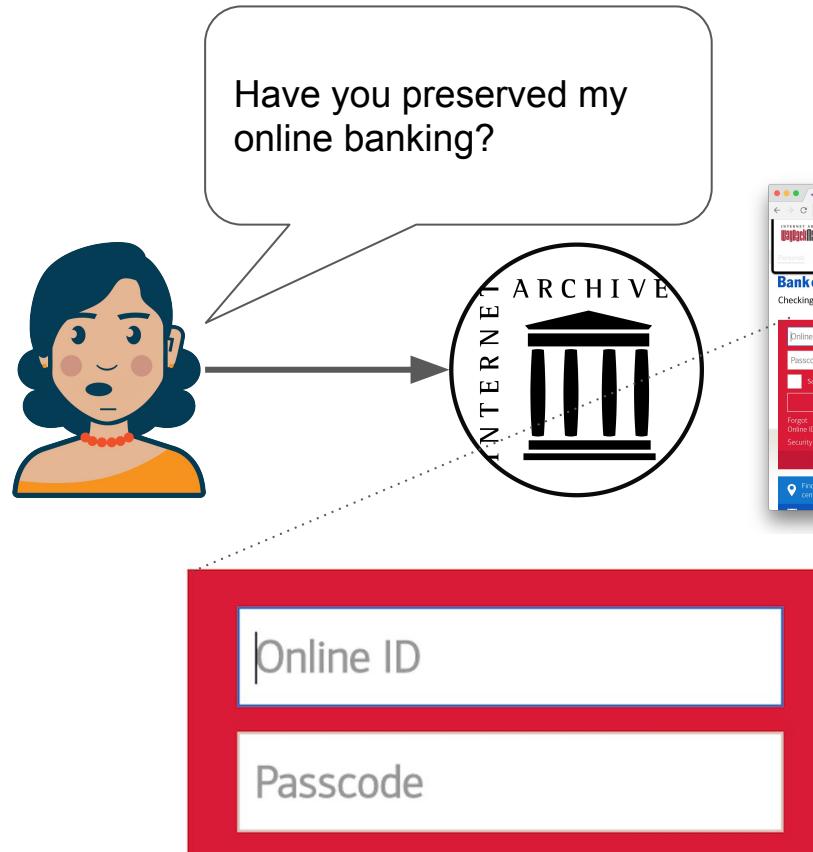
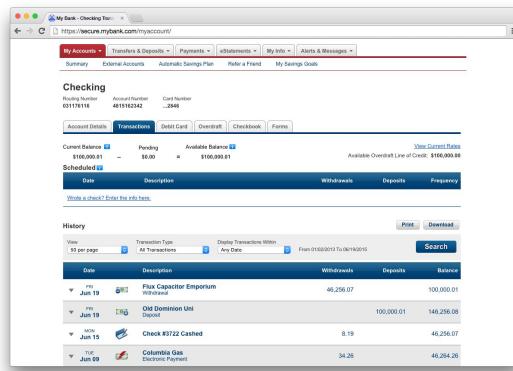
# User sees on live Web may not be what is captured



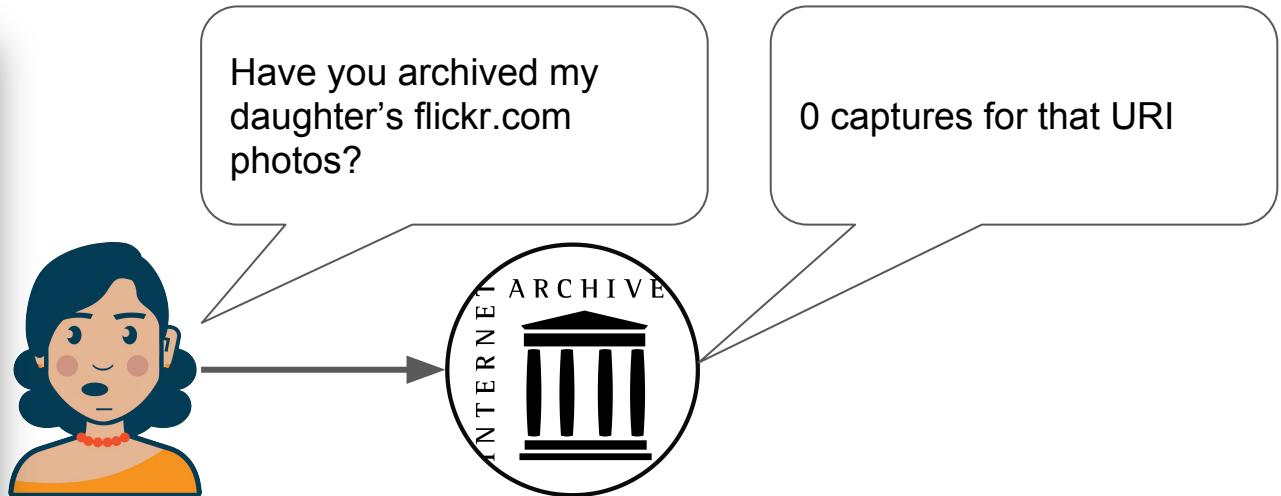
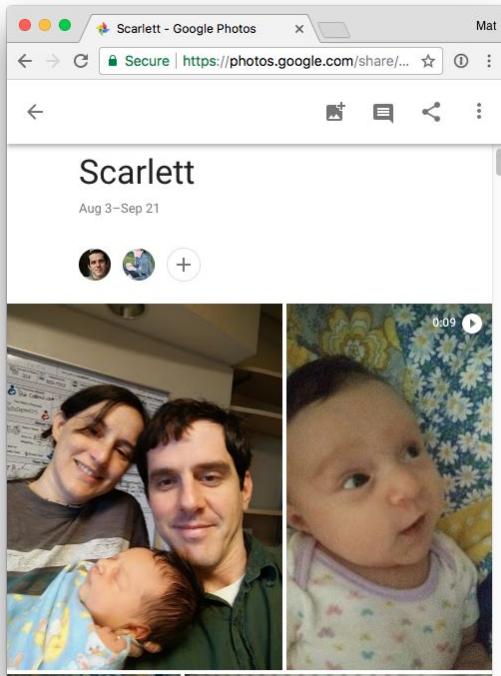
What did facebook.com look like in the past?



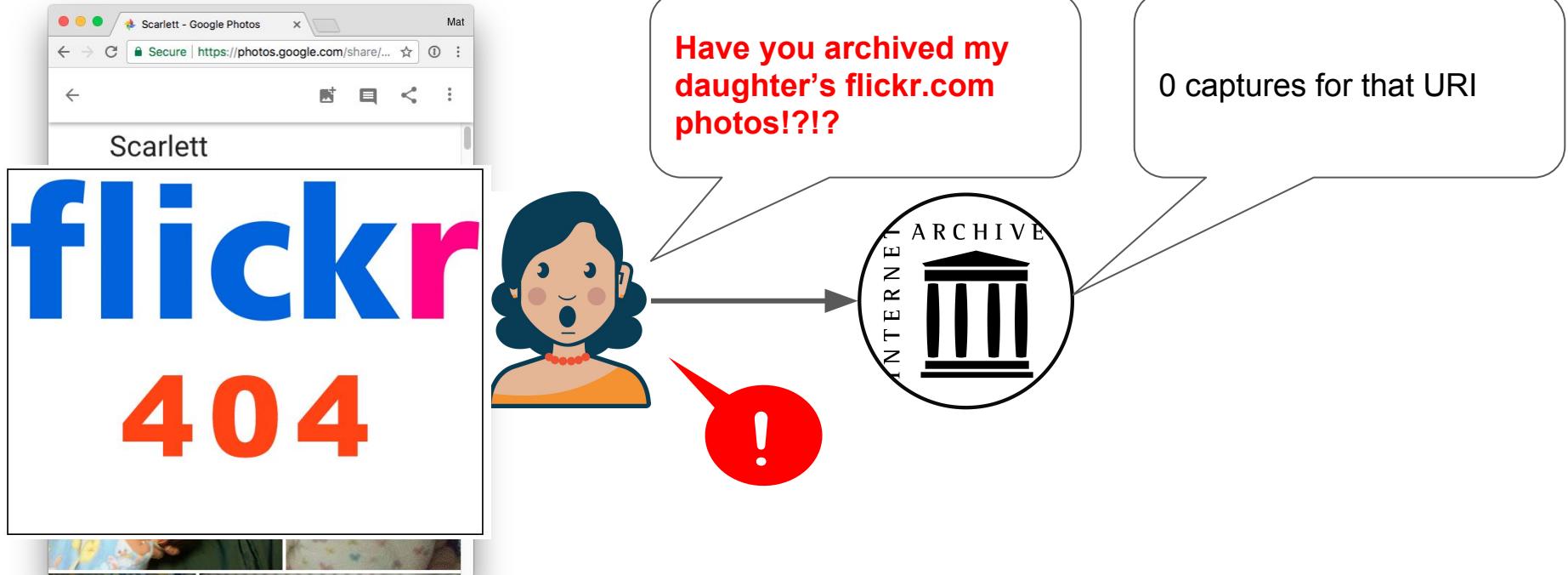
# ...And oftentimes that is for the best



# Other times, we may want our content archived

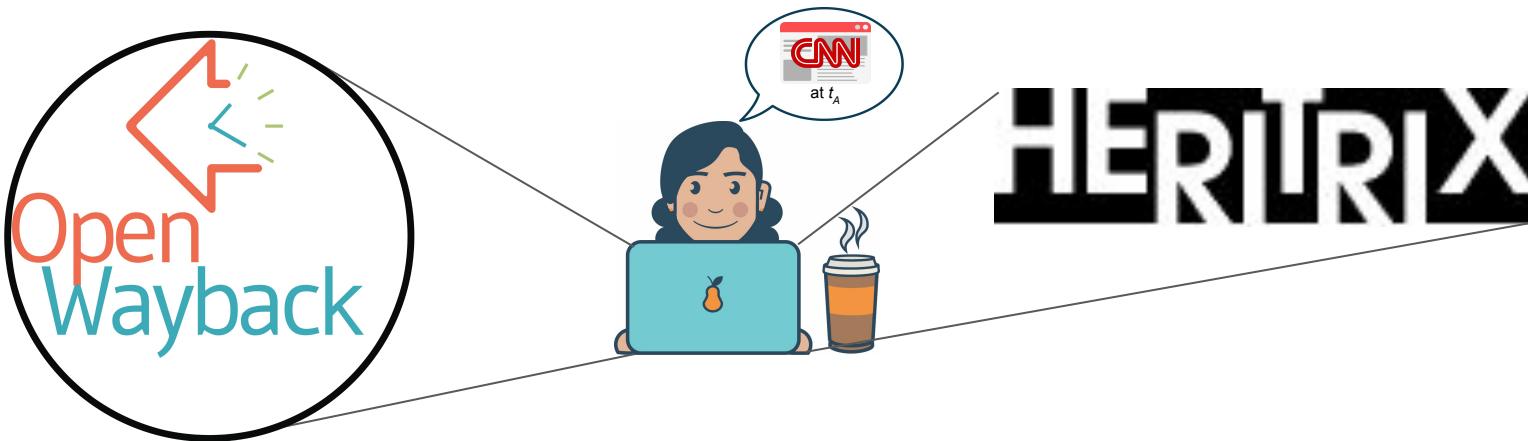


# ...especially when it has disappeared



# “Save this, but only for me.”

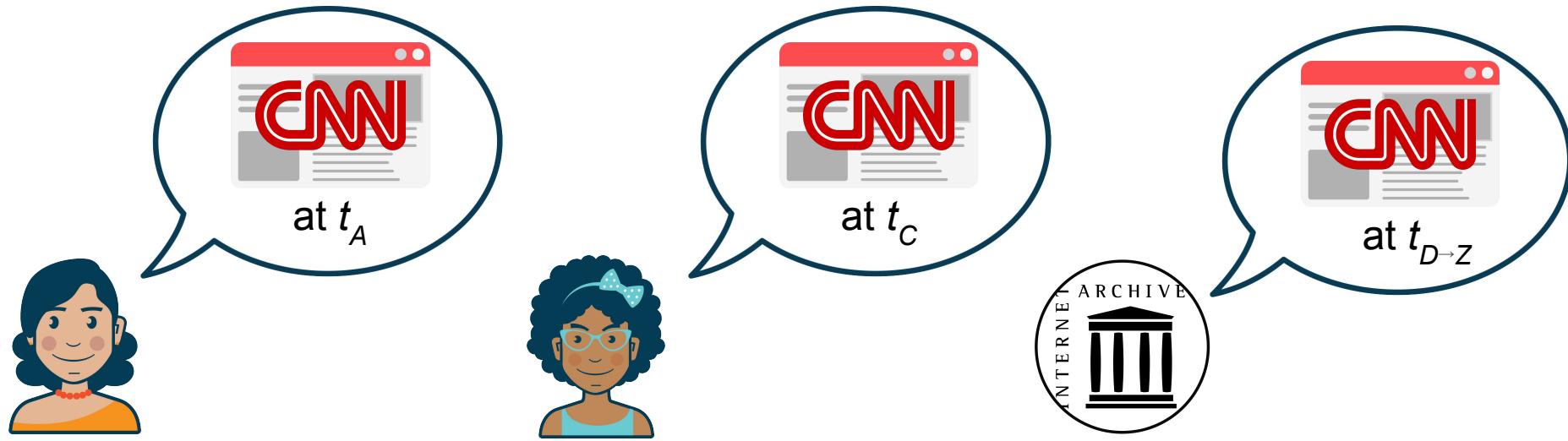
- **Screenshots** of Web pages are insufficient
  - Not interactive/representative, do not integrate, lose context otherwise provided in metadata
- Large-scale archives' tools are open source
- Individuals can archive, but there are still technical barriers



# Individuals, Too, Can Archive The Web



# Captures from Institutional and Personal Sources

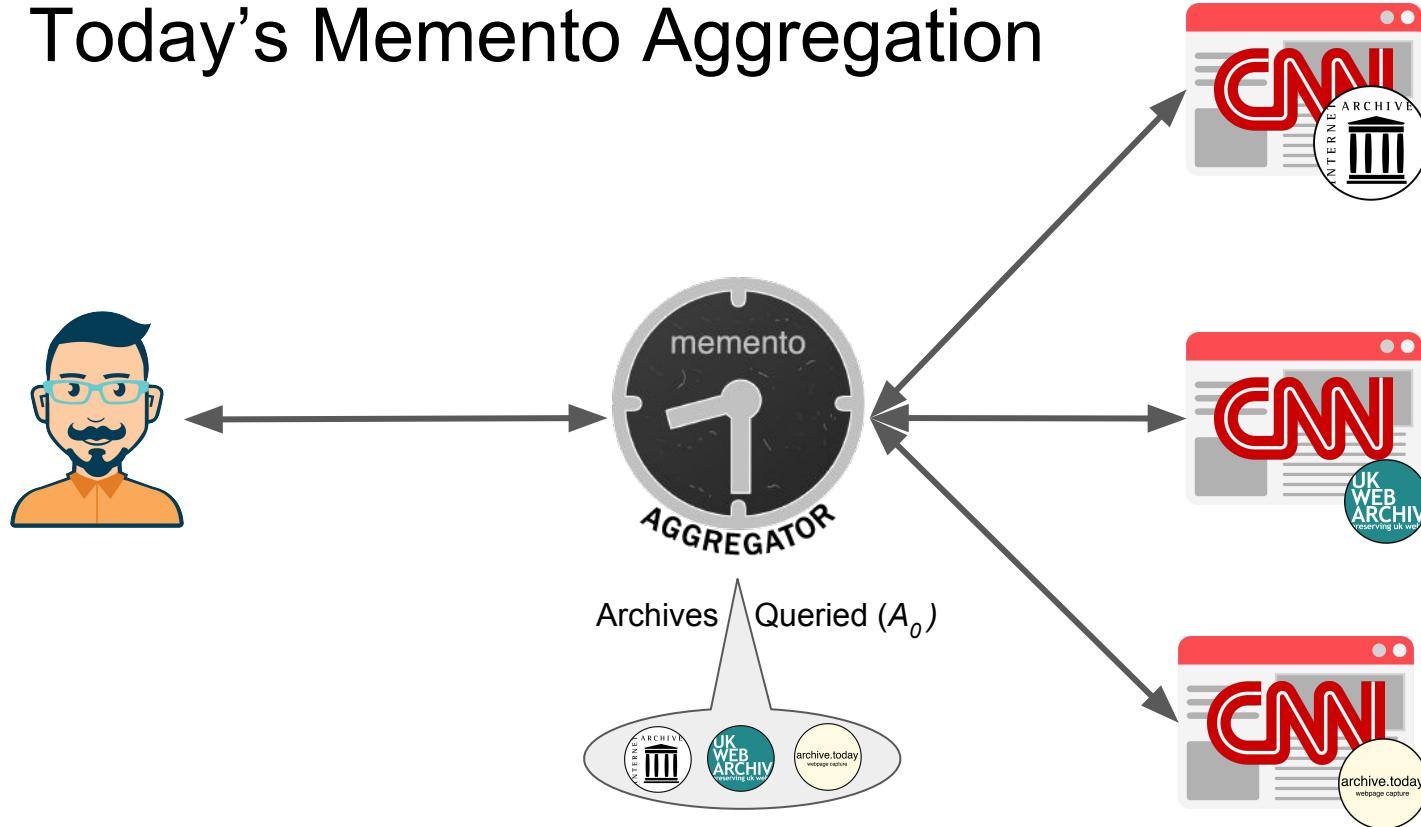


# Memento Facilitates this Aggregation

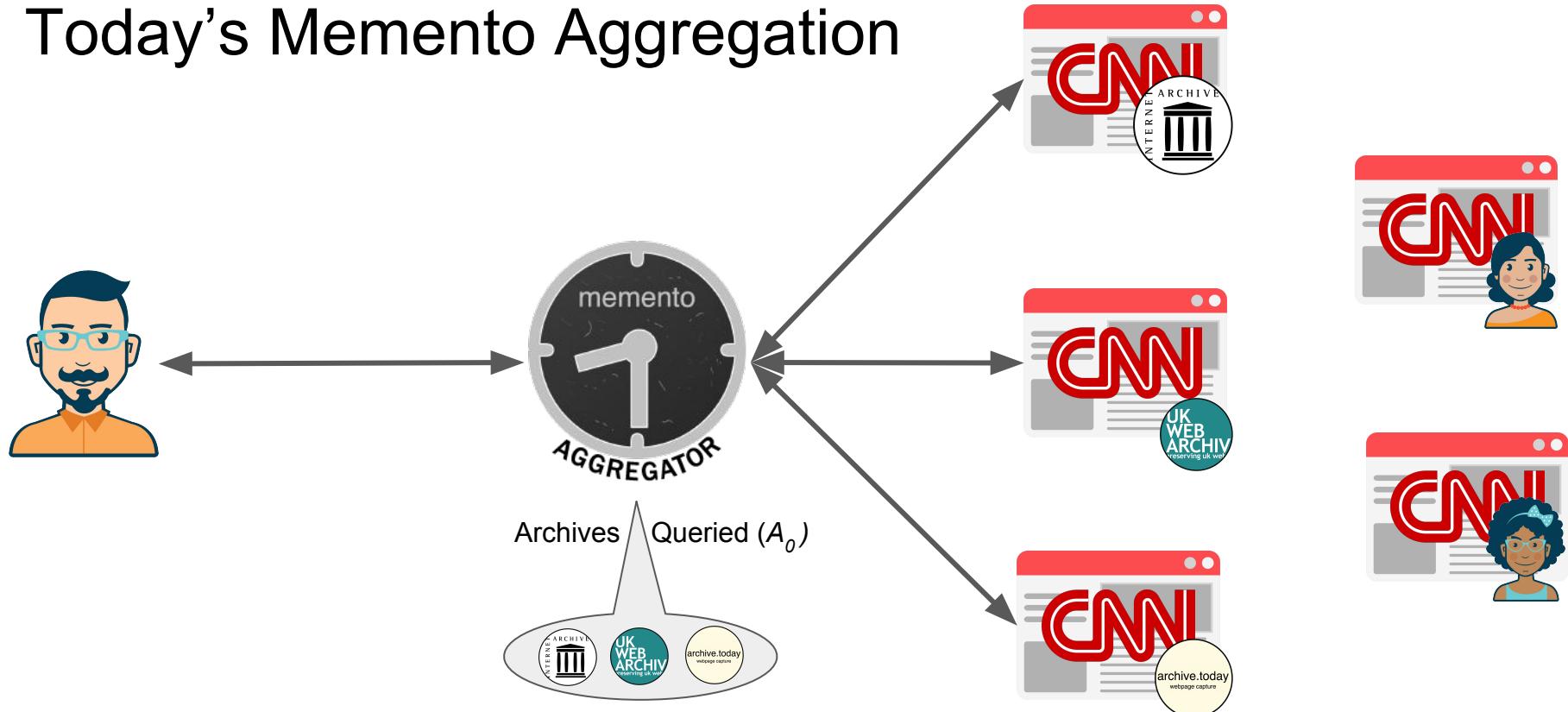
RFC7089



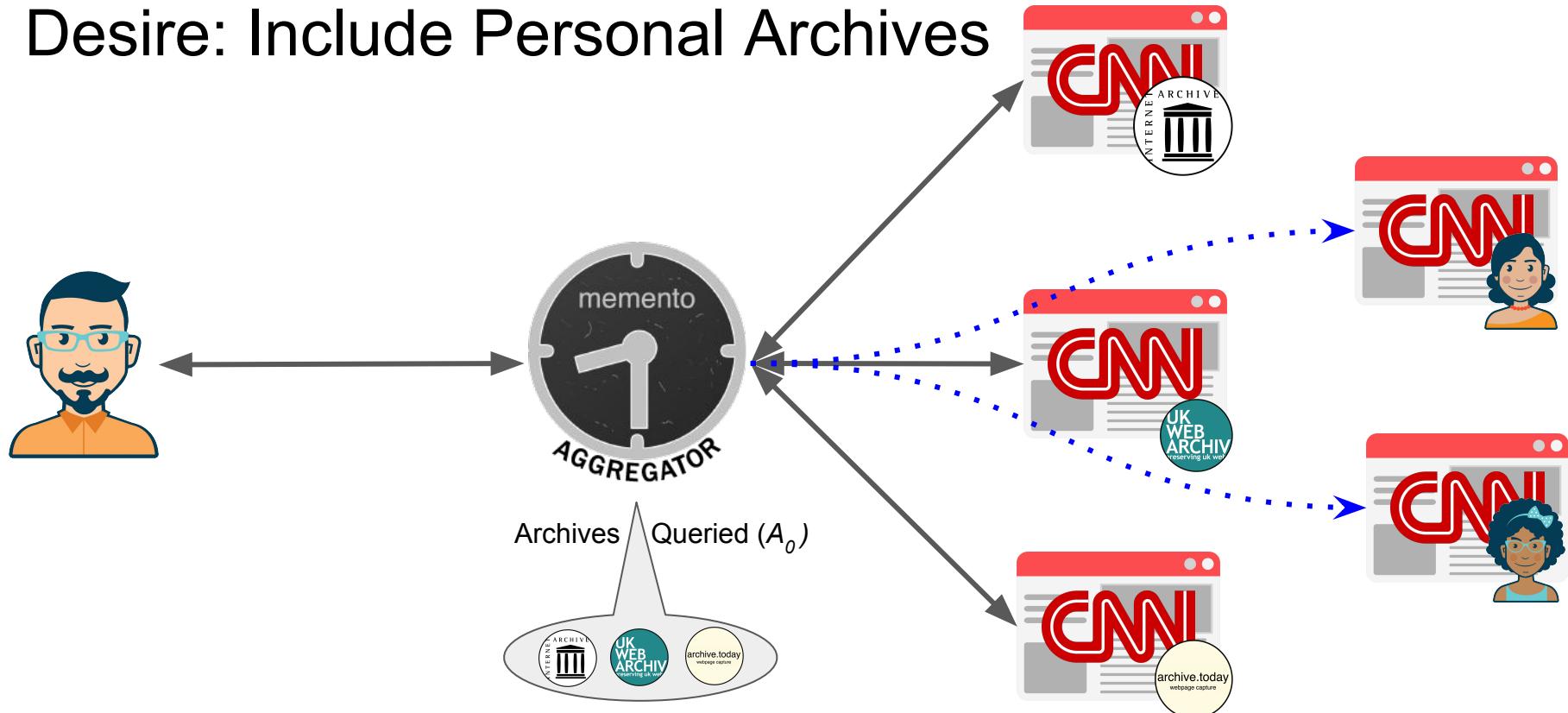
# Today's Memento Aggregation



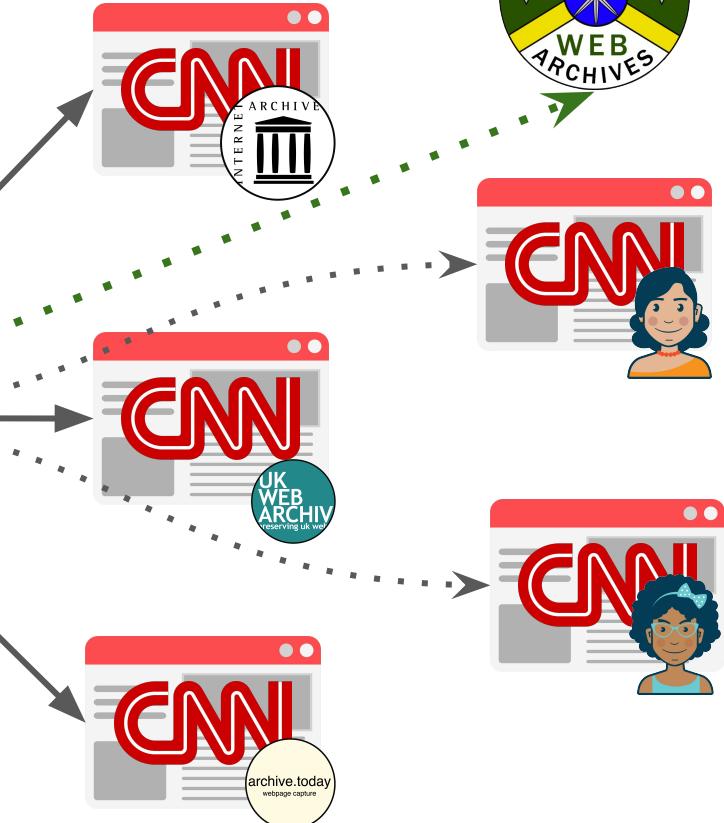
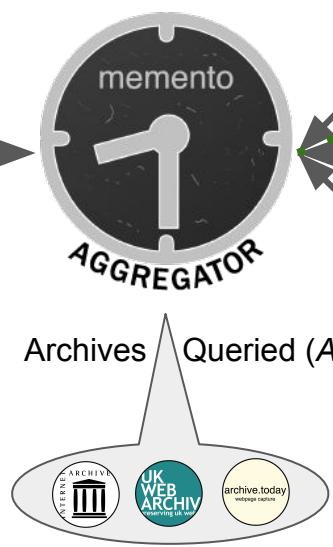
# Today's Memento Aggregation



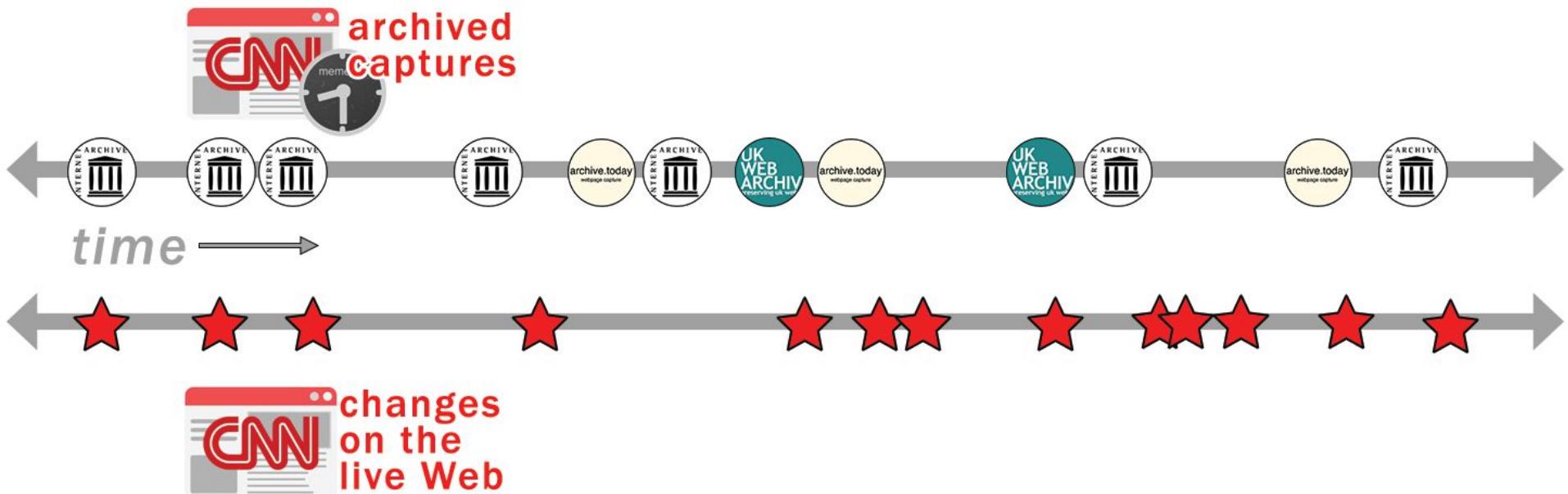
# Desire: Include Personal Archives



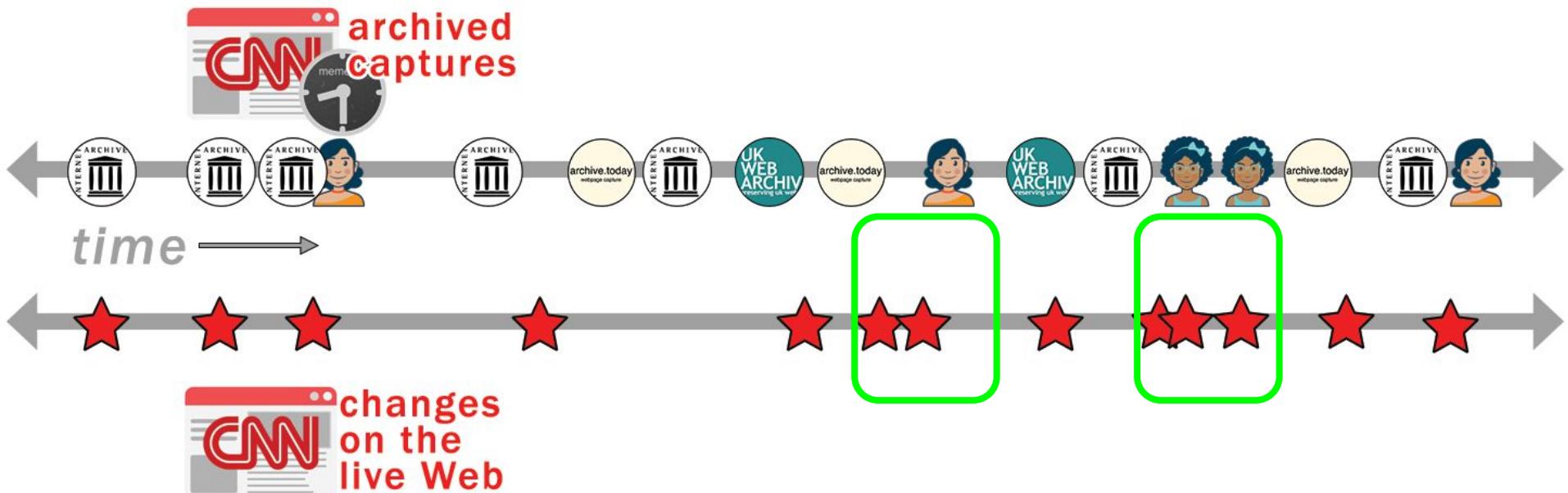
# Desire: Include Other Non-Aggregated Archives



# Rapidly Changing Pages May Not Be Comprehensively Captured



# Archiving More Archives Provides a Better Picture of the Web



# Research Questions

**RQ1:** What sort of content is difficult to capture and replay for preservation from the perspective of a Web browser?

**RQ2:** How do Web browser APIs compare in potential functionality to the capabilities of archival crawlers?

**RQ3:** What issues exist for capturing and replaying content behind authentication?

**RQ4:** How can content that was captured behind authentication signal to Web archive replay systems that it requires special handling?

**RQ5:** How can Memento aggregators indicate that private Web archive content requires special handling to be replayed, despite being aggregated with publicly available Web archive content?

**RQ6:** What kinds of access control do users who create private Web archives need to regulate access to their archives?

# Research Questions

**RQ1:** What sort of **content is difficult to capture** and replay for preservation from the perspective of a Web browser?

**RQ2:** How do **Web browser APIs compare** in potential functionality to the capabilities of archival crawlers?

**RQ3:** What issues exist for capturing and replaying **content behind authentication**?

**RQ4:** How can **content** that was captured behind authentication **signal** to Web archive replay systems that it **requires special handling**?

**RQ5:** How can Memento **aggregators indicate** that private Web archive content requires **special handling** to be replayed, despite being aggregated with publicly available Web archive content?

**RQ6:** What kinds of access control do users who create private Web archives need to **regulate access** to their archives?

# Outline

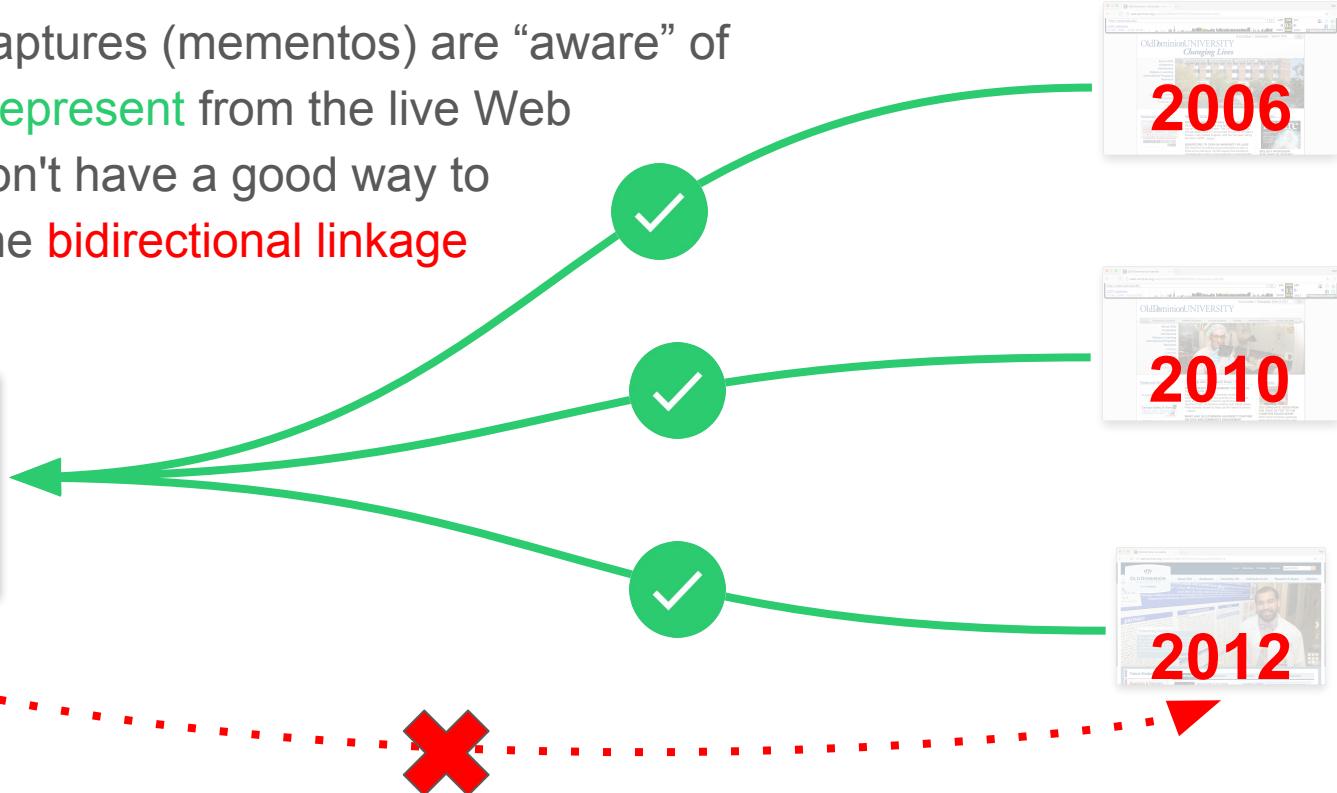
- **Introduction/Motivation**
- Background
- Preliminary Research
- Proposed Framework
- Evaluation Plan

# Outline

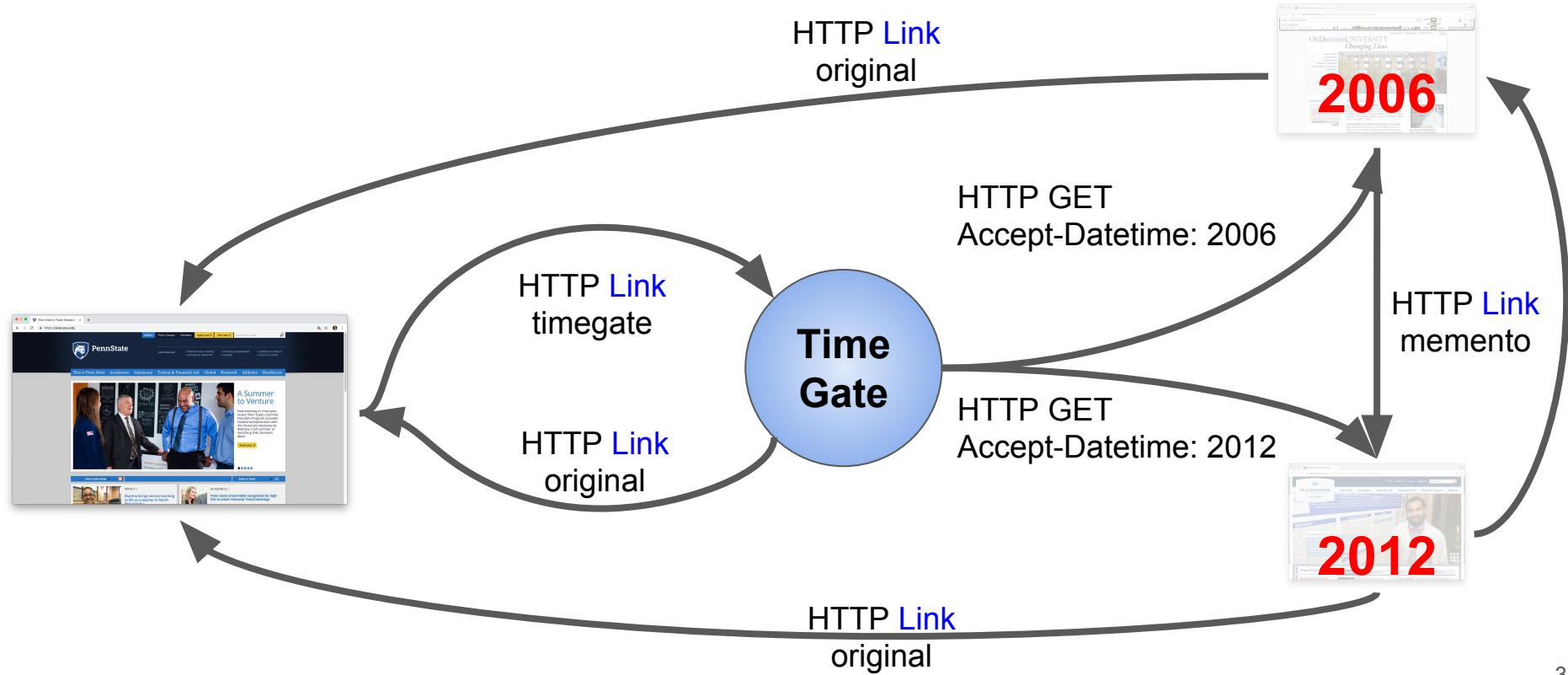
- Introduction/Motivation
- **Background**
- Preliminary Research
- Proposed Framework
- Evaluation Plan

# Needed Association of Live-to-Archived Web

- Archived captures (mementos) are “aware” of **what they represent** from the live Web
- ...but we don't have a good way to establish the **bidirectional linkage**

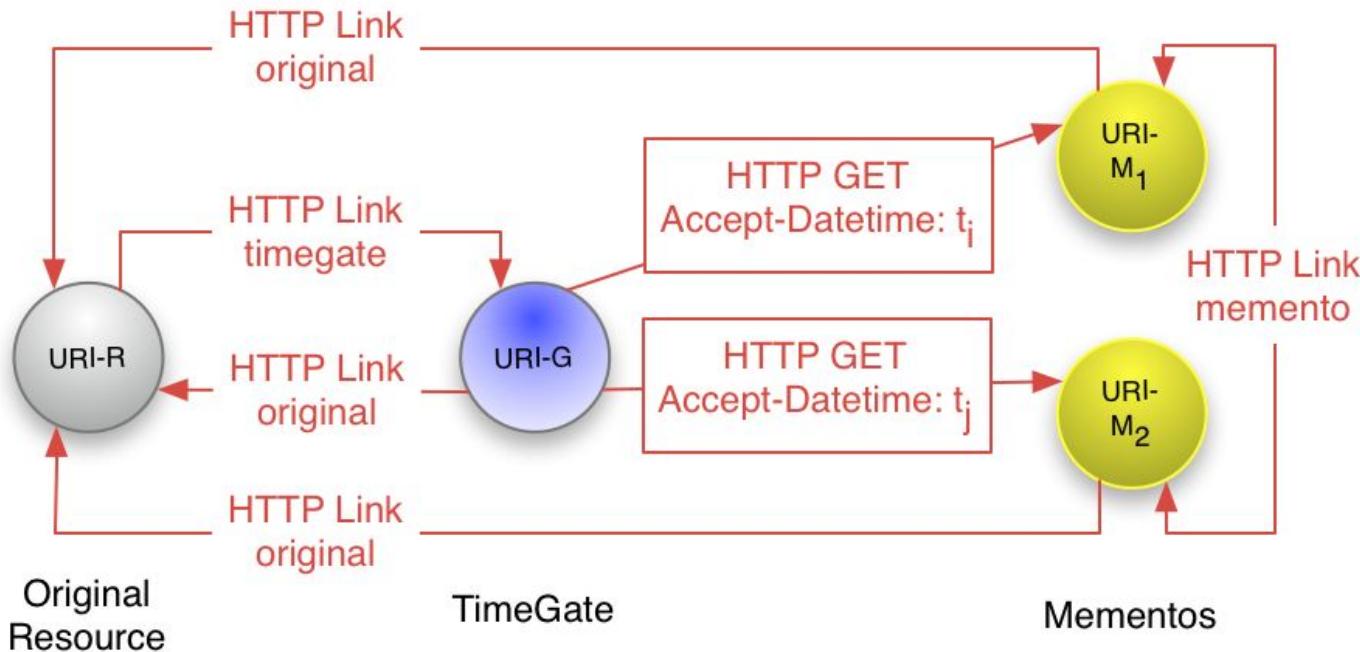


# Representations can be **Linked** in time





# Background: Memento

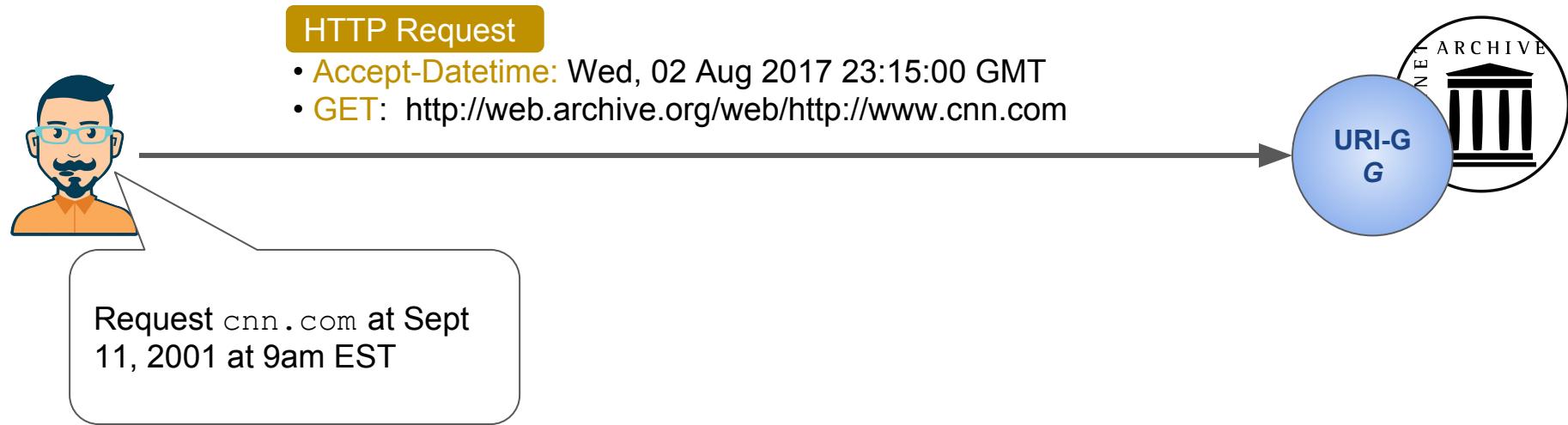


Memento Guide: Introduction. <http://www.mementoweb.org/guide/quick-intro/>, January 2015.

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

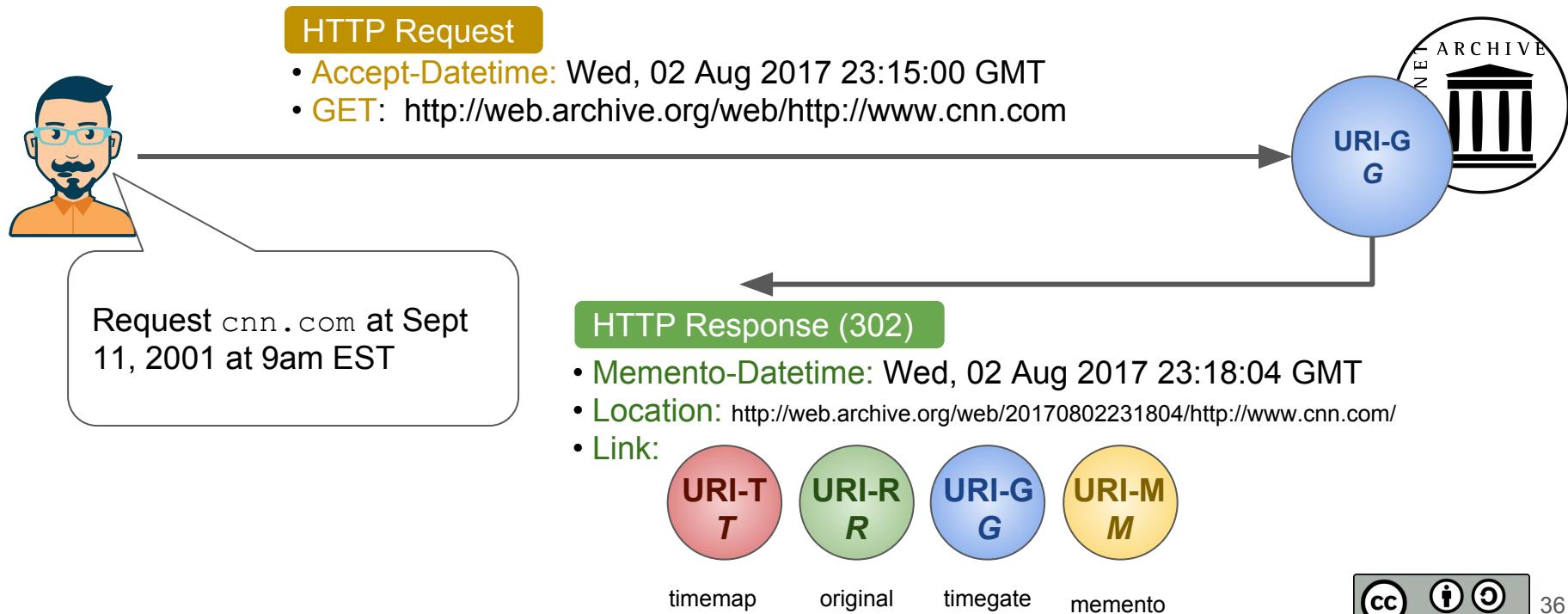


# Background: Memento Request Example

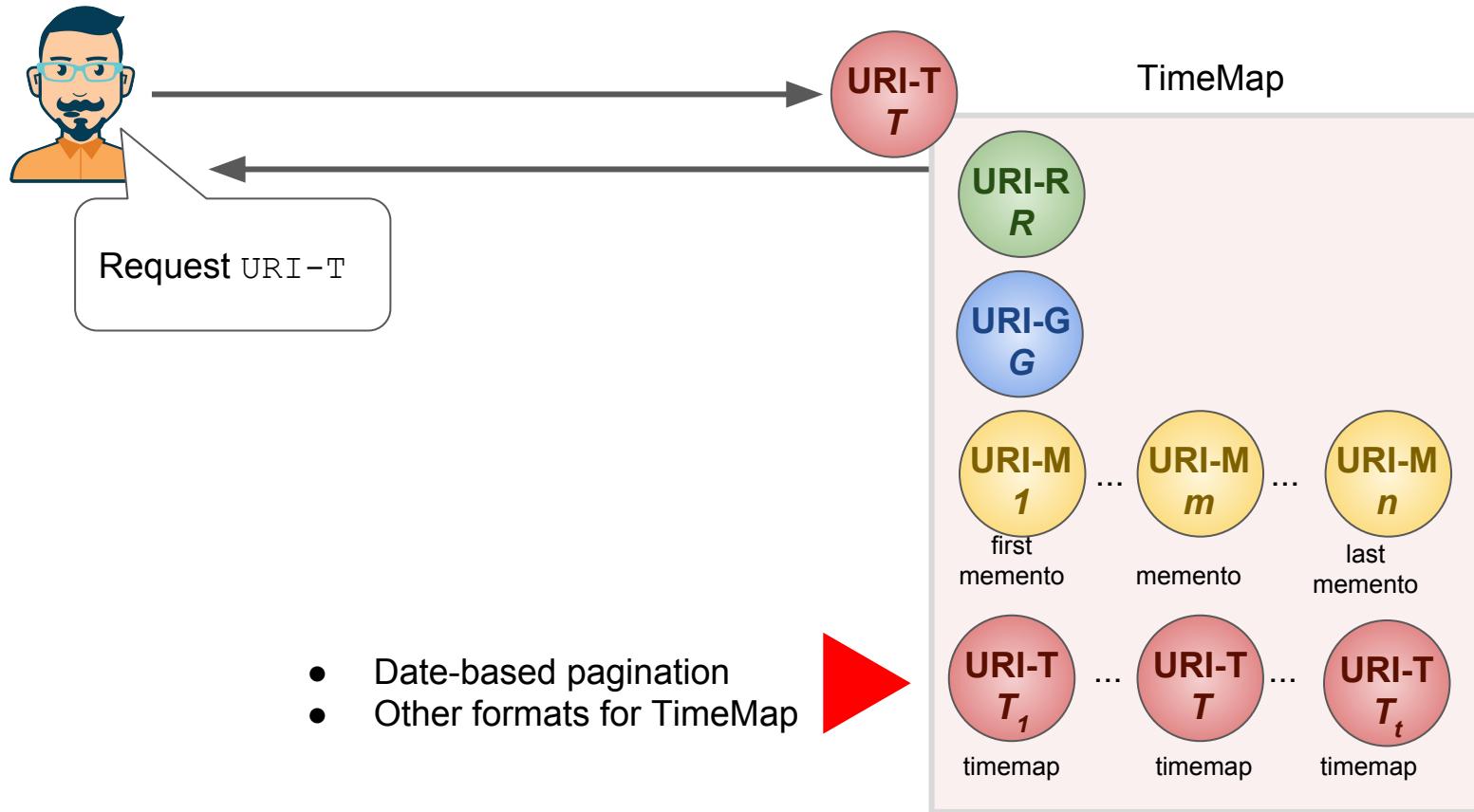




# Background: Memento Request Example

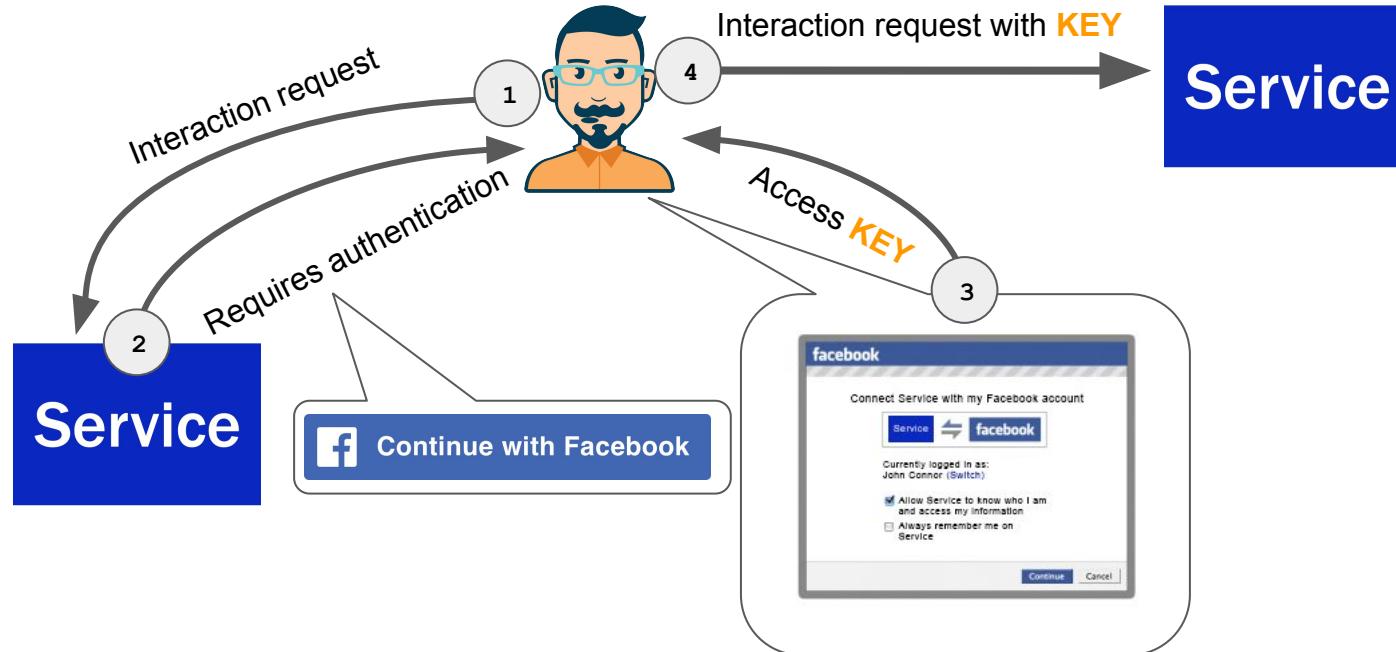


# Background: Dereferencing a TimeMap at URI-T



# Role-based delegation and authentication

A familiar paradigm used for authentication on the live Web



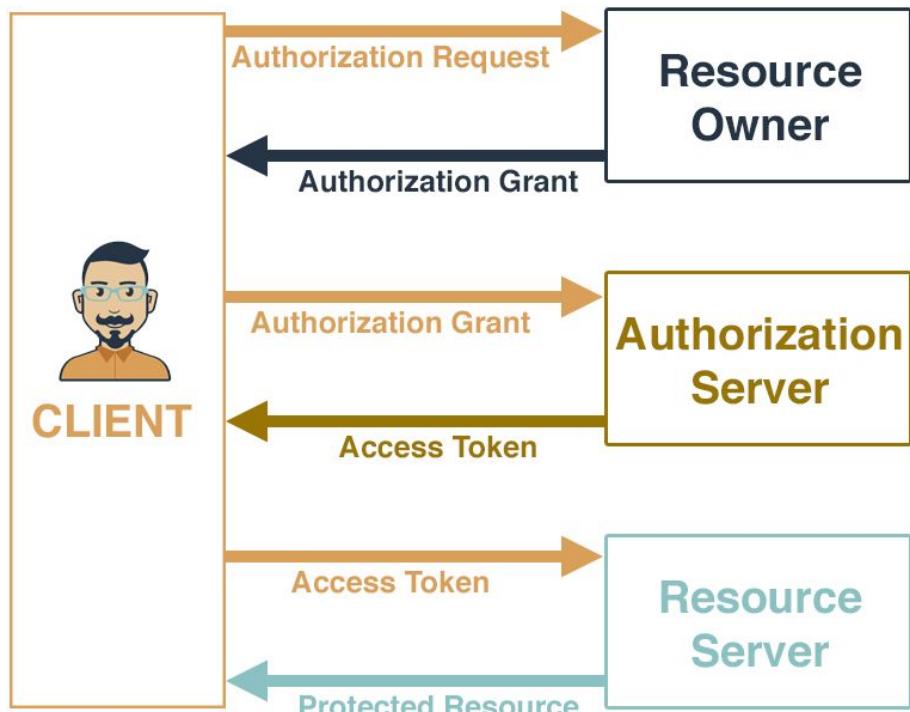


# Background - Privacy and Security

- Web users question trusting institutions to preserve private Web contents<sup>1</sup>
- OAuth 2.0<sup>2</sup> facilitates authentication cohesion of entities

RQ3: What issues exist for capturing and replaying content behind authentication?

RQ6: What kinds of access control do users who create private Web archives need to regulate access to their archives?

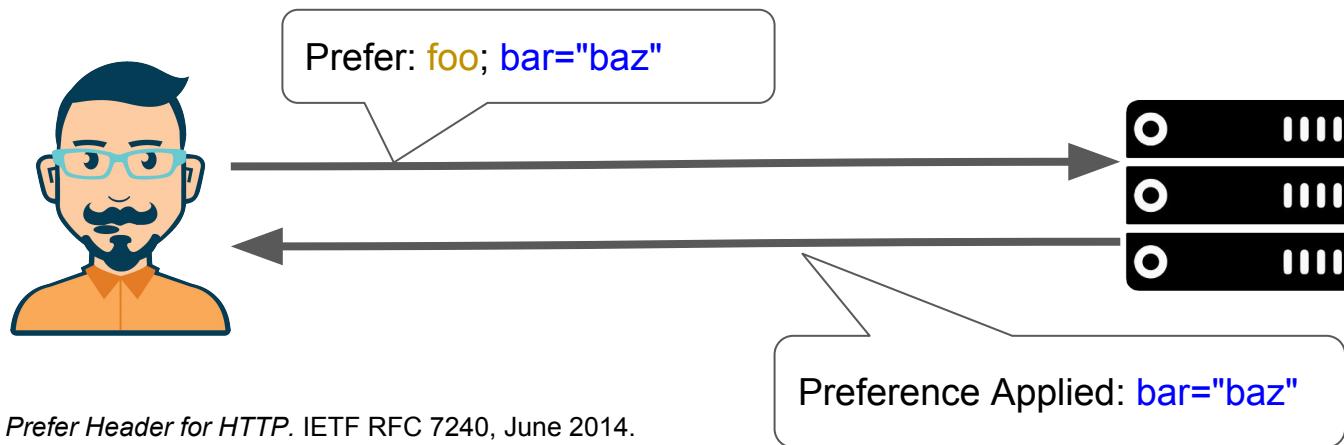


<sup>1</sup> Marshall and Shipman., “On the Institutional Archiving of Social Media”, JCDL 2012

<sup>2</sup> D. Hardt. *The OAuth 2.0 Authorization Framework*. IETF RFC 6749, October 2012.

# HTTP Prefer

- HTTP negotiation already available via Accept-\* headers
- *Prefer* syntax provide mechanism for client to specify preferences
  - ...with which servers may not comply



\* J. Snell. *Prefer Header for HTTP*. IETF RFC 7240, June 2014.



# Memento Aggregation State of the Art





# Memento Aggregation - MementoWeb

time travel

Find Mementos in Internet Archive, Archive-It, British Library, archive.today, GitHub and [many more!](#)

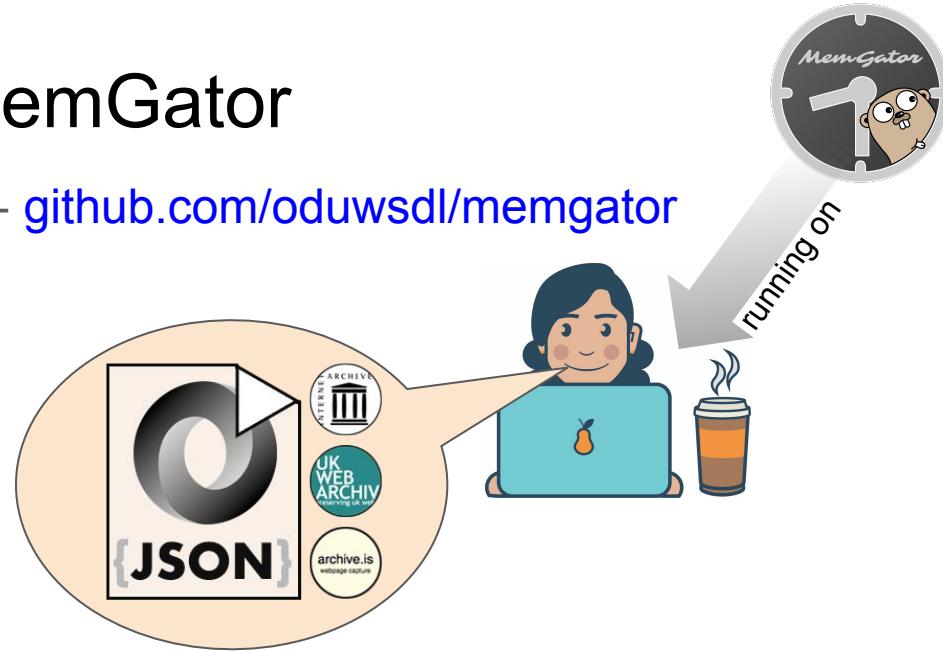
experience web time travel  
install Memento for Chrome  
enable web time travel  
install Memento for MediaWiki  
say no to \*404 Not Found\*

*Also available via CLI:*

```
$ curl http://timetravel.mementoweb.org/timemap/link/http://odu.edu
```

# Memento Aggregation - MemGator

- Open Source Memento Aggregator - [github.com/oduwsdl/memgator](https://github.com/oduwsdl/memgator)
- Easy personal/local deployment
- Specify archive list on launch
  - Easily configurable **JSON** →
  - Use default collection if not specified
- TimeMap Formats:
  - Link
  - **JSON**
  - **CDXJ**



\* Alam and Nelson, “MemGator - A Portable Concurrent Memento Aggregator: Cross-Platform CLI and Server Binaries in Go”, JCDL 2016

# CDXJ: An Alternative TimeMap Format

```
<http://matkelly.com>; rel="original",
<http://localhost:1208/timemap/link/http://matkelly.com>;
rel="self"; type="application/link-format",
<http://web.archive.org/web/20060514123511/http://www.mat
kelly.com:80/>; rel="first memento"; datetime="Sun, 14
May 2006 12:35:11 GMT",
<http://web.archive.org/web/20060516213852/http://www.mat
kelly.com/>; rel="memento"; datetime="Tue, 16 May 2006
21:38:52 GMT",
...
<http://web.archive.org/web/20180128152125/http://matkell
y.com>; rel="memento"; datetime="Sun, 28 Jan 2018
15:21:25 GMT",
<http://web.archive.org/web/20180319141920/http://matkell
y.com/>; rel="last memento"; datetime="Mon, 19 Mar 2018
14:19:20 GMT",
<http://localhost:1208/timemap/link/http://matkelly.com>;
rel="timemap"; type="application/link-format",
<http://localhost:1208/timemap/json/http://matkelly.com>;
rel="timemap"; type="application/json",
<http://localhost:1208/timemap/cdxj/http://matkelly.com>;
rel="timemap"; type="application/cdxj+ors",
<http://localhost:1208/timegate/http://matkelly.com>;
rel="timegate"
```

```
!context ["http://tools.ietf.org/html/rfc7089"]
!id {"uri": "http://localhost:1208/timemap/cdxj/http://matkelly.com"}
!keys ["memento_datetime_YYYYMMDDhhmmss"]
!meta {"original_uri": "http://matkelly.com"}
!meta {"timegate_uri":
"http://localhost:1208/timegate/http://matkelly.com"}
!meta {"timemap_uri": {"link_format":
"http://localhost:1208/timemap/link/http://matkelly.com", "json_format":
"http://localhost:1208/timemap/json/http://matkelly.com", "cdxj_format":
"http://localhost:1208/timemap/cdxj/http://matkelly.com"}}
20060514123511 {"uri":
"http://web.archive.org/web/20060514123511/http://www.matkelly.com:80/",
"rel": "first memento", "datetime": "Sun, 14 May 2006 12:35:11 GMT"}
20060516213852 {"uri":
"http://web.archive.org/web/20060516213852/http://www.matkelly.com/",
"rel": "memento", "datetime": "Tue, 16 May 2006 21:38:52 GMT"}
...
20180128152125 {"uri":
"http://web.archive.org/web/20180128152125/http://matkelly.com", "rel":
"memento", "datetime": "Sun, 28 Jan 2018 15:21:25 GMT"}
20180319141920 {"uri":
"http://web.archive.org/web/20180319141920/http://matkelly.com", "rel":
"last memento", "datetime": "Mon, 19 Mar 2018 14:19:20 GMT"}
```

## Link (RFC 7089) TimeMap

Original URI (URI-R)

Other TimeMaps (URI-Ts)

## CDXJ TimeMap

Relative Relations

# CDXJ: An Alternative TimeMap Format

```
<http://matkelly.com>; rel="original",
<http://localhost:1208/timemap/link/http://matkelly.com>;
rel="self"; type="application/link-format",
<http://web.archive.org/web/20060514123511/http://www.mat-
kelly.com:80/>; rel="first memento"; datetime="Sun, 14
May 2006 12:35:11 GMT",
<http://web.archive.org/web/20060516213852/http://www.mat-
kelly.com/>; rel="memento"; datetime="Tue, 16 May 2006
21:38:52 GMT",
...
<http://web.archive.o.../sh/20180319141920/http://matkell
y.com>; rel="memento"; datetime="Mon, 19 Mar 2018
14:19:20 GMT",
<http://web.archive.org/web/20180319141920/http://matkell
y.com/>; rel="last memento"; datetime="Mon, 19 Mar 2018
14:19:20 GMT",
<http://localhost:1208/timemap/link/http://matkelly.com>;
rel="timemap"; type="application/link-format",
<http://localhost:1208/timemap/json/http://matkelly.com>;
rel="timemap"; type="application/cdxj+json",
<http://localhost:1208/timegate/http://matkelly.com>;
rel="timegate"
```

## MAIN POINTS

### *Link, CDXJ, and JSON TimeMaps:*

Multiple formats to express same information

*Link syntax is not expandable:*

They were meant to be displayed in a constrained environment  
(in HTTP Link headers)

```
!context ["http://tools.ietf.org/html/rfc7089"]
!id {"uri": "http://localhost:1208/timemap/cdxj/http://matkelly.com"}
!meta {"uri": "http://matkelly.com"}
!meta {"timegate_uri": "http://localhost:1208/timegate/http://matkelly.com"}
!meta {"timemap_uri": {"link_format": "http://localhost:1208/timemap/link/http://matkelly.com", "json_format": "http://localhost:1208/timemap/cdxj+json", "cdxj_format": "http://localhost:1208/timemap/cdxj/http://matkelly.com"}}
20060514123511 {"uri": "http://www.matkelly.com:80/", "rel": "first memento", "datetime": "Sun, 14 May 2006 12:35:11 GMT"} 20060516213852 {"uri": "http://web.archive.org/web/20060516213852/http://www.matkelly.com/", "rel": "memento", "datetime": "Tue, 16 May 2006 21:38:52 GMT"} 20180319141920 {"uri": "http://web.archive.org/web/20180319141920/http://matkelly.com", "rel": "last memento", "datetime": "Mon, 19 Mar 2018 14:19:20 GMT"}
```

### Link (RFC 7089) TimeMap

Original URI (URI-R)

Other TimeMaps (URI-Ts)

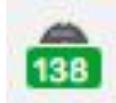
### CDXJ TimeMap

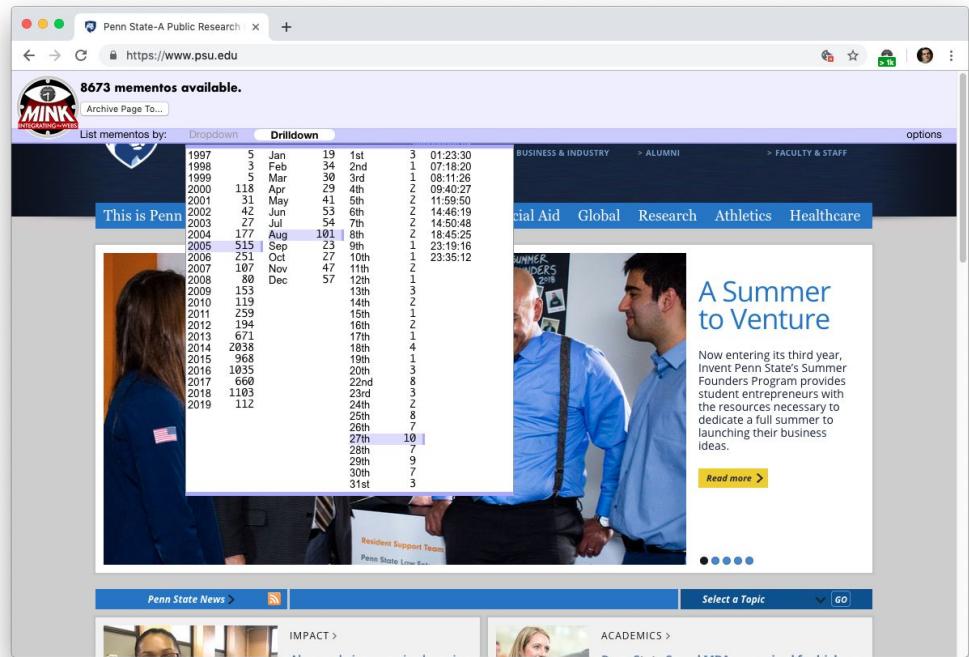
TimeGate (URI-G)

Relative Relations



# : visual user interaction with aggregators

- Bridges gap between live and archived Webs
- Leverages Memento aggregator's capability, returns TimeMaps
- Indicates # of captures for a URI while you browse
- Provides navigation of mementos while browsing live Web
- Single-click submission of URI-R to multiple Web archives



The screenshot shows a web browser displaying the Penn State-A Public Research website at <https://www.psu.edu>. The page features a prominent MINK logo in the top left corner. A modal window titled "8673 mementos available." is open, listing mementos by date from 1997 to 2019. The modal includes a "Dropdown" and "Drilldown" button. The main content area shows a photograph of two men in blue shirts and suspenders, with a banner above them reading "A Summer to Venture". Below the banner, text mentions the Invent Penn State's Summer Founders Program. At the bottom of the page, there are links for "Penn State News", "IMPACT", and "ACADEMICS".

Year	Month	Date	Count
1997	5	Jan	19
1998	3	Feb	2nd
1999	5	Mar	30
2000	118	Apr	29
2001	31	May	41
2002	42	Jun	53
2003	27	Jul	54
2004	177	Aug	78
2005	515	Sep	23
2006	251	Oct	27
2007	107	Nov	47
2008	86	Dec	57
2009	152		12th
2010	119		13th
2011	259		14th
2012	194		15th
2013	671		16th
2014	2838		17th
2015	568		18th
2016	1835		19th
2017	660		20th
2018	1103		22nd
2019	112		23rd
			24th
			25th
			26th
			27th
			28th
			29th
			30th
			31st

\* Kelly et al., "Mink: Integrating the Live and Archived Web Viewing Experience Using Web Browsers and Memento", JCDL 2014

# Outline

- Introduction/Motivation
- Background
- **Preliminary Research**
- Proposed Framework
- Evaluation Plan

# Preliminary Research

1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. JCDL 2014 - Mink
5. JCDL 2014 - Archival Acid Test
6. JCDL 2014 - Not All Mementos
7. IJDL 2015 - Impact of JavaScript
8. IJDL 2015 - Not All Mementos
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary Wayback (ipwb)
11. TPDL 2016 - ipwb extended
12. JCDL 2017 - WAIL Electron
13. JCDL 2017 - ServiceWorker Replay
14. JCDL 2017 - Impact of Canonicalization
15. JCDL 2018 - ArchiveNow
16. JCDL 2018 - Replay Banners
17. JCDL 2018 - A Framework...

A Framework for Aggregating Public and Private Web Archives  
February 14, 2019  
*Mat Kelly*

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 way to "long term storage, maintenance, and access of personal digital assets that have emotional, intellectual, and historical value to individuals" [3].

**Preserve everything you see!**

## Categories and Subject Descriptors

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

## General Terms

**RQ1:** What sort of content is difficult to capture and replay for preservation from the perspective of a Web browser?

Personal Web Archiving, WARC, Browser, Wayback Machine, Internet Archive

**RQ2:** How do Web browser APIs compare in potential functionality to the capabilities of archival crawlers?

The Internet Archive, along with web archives at other libraries and institutions, is alone a remarkable collection of archiving data 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

Copyright is held by the author/owner(s).  
JCDL '12, June 10–14, 2012, Washington, DC, USA.  
ACM 978-1-4503-154-0/12/06.

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<sup>1</sup> 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 located on the login screen, 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<sup>2</sup> of <http://www.craigslist.org> redirects to <http://sfbay.craigslist.org>.

# Preliminary Research

1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. JCDL 2014 - Mink
5. JCDL 2014 - Archival Acid Test
6. JCDL 2014 - Not All Mementos
7. IJDL 2015 - Impact of JavaScript
8. IJDL 2015 - Not All Mementos
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary Wayback (ipwb)
11. TPDL 2016 - ipwb extended
12. JCDL 2017 - WAIL Electron
13. JCDL 2017 - ServiceWorker Replay
14. JCDL 2017 - Impact of Canonicalization
15. JCDL 2018 - ArchiveNow
16. JCDL 2018 - Replay Banners
17. JCDL 2018 - A Framework...

## 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](mailto:{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 web pages that have been archived over time, including 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.

### Which things are hard to preserve?

**Keywords:** Web Archiving, Digital Preservation

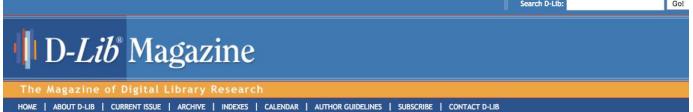
### RQ1: What sort of content is difficult to capture and replay for preservation from the perspective of a Web browser?

Adoption of JavaScript allowed the components on a web page to respond to users' actions or be manipulated in ways that made the page more usable. Ajax [9] combines multiple web technologies to give web pages the ability to perform operations asynchronously. The adoption of Ajax by web developers facilitated the fluidity of user interaction on the web. Through each phase in the progression of the web, the ability to preserve the content displayed to the user has also progressed but in a less linear trend.

A large amount of the difficulty in web archiving stems from the crawler's insufficient ability to capture content related to JavaScript. Because JavaScript is executed on the client side (i.e., within the browser after 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

# Preliminary Research

1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. JCDL 2014 - Mink
5. JCDL 2014 - Archival Acid Test
6. JCDL 2014 - Not All Mementos
7. IJDL 2015 - Impact of JavaScript
8. IJDL 2015 - Not All Mementos
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary Wayback (ipwb)
11. TPDL 2016 - ipwb extended
12. JCDL 2017 - WAIL Electron
13. JCDL 2017 - ServiceWorker Replay
14. JCDL 2017 - Impact of Canonicalization
15. JCDL 2018 - ArchiveNow
16. JCDL 2018 - Replay Banners
17. JCDL 2018 - A Framework...



## D-Lib Magazine

The Magazine of Digital Library Research

[HOME](#) | [ABOUT D-LIB](#) | [CURRENT ISSUE](#) | [ARCHIVE](#) | [INDEXES](#) | [CALENDAR](#) | [AUTHOR GUIDELINES](#) | [SUBSCRIBE](#) | [CONTACT D-LIB](#)

### November/December 2013

Volume 19, Number 11/12

[Table of Contents](#)

#### A Method for Identifying Personalized Representations in Web Archives

Mat Kelly, Justin F. Brunelle, Michele C. Weigle, and Michael L. Nelson  
Old Dominion University  
[mkelly,jbrunelle,mweigle,mln]@cs.odu.edu

doi:10.1045/november2013-kelly

#### [Printer-friendly Version](#)

#### Abstract

Web resources are becoming increasingly personalized — two different users clicking on the same link at the same time can see content customized for each individual user. These changes result in multiple representations of a resource that cannot be canonicalized in Web archives. We identify characteristics of this problem by presenting a potential solution to generalize personalized representations in archives. We also present our proof-of-concept prototype that analyzes WARC (Web ARCHive) format files, inserts metadata establishing relationships, and provides archive users the ability to navigate on the additional dimension of environment variables in a modified Wayback Machine.

#### Introduction

Personalized web resources offer different representations [8] to different users based on the user-agent string and other values in the HTTP request headers, Geoloc, and other environmental factors. This means Web crawlers capturing content for archives may receive representations based on the crawl environment which will differ from the representations returned to the interactive users. In summary, what we archive is increasingly different from what we as interactive users experience.

Some preserved things are personalized

With the increasing prevalence of mobile browsers on the Web [50], it is becoming important to capture these mobile representations of resources.

Mobile pages often contain links to additional resources instead of embedded text and often reduce the number of images embedded in the page [19]. For example,

the mobile representation of [m.cnn.com](http://m.cnn.com) has only one image, while the desktop representation has 10 images. When a user clicks on the image in the mobile representation, identified by URI-R [1], it is accessed. It redirects to <http://m.cnn.go.com>, effectively giving two separate but related URI-R values that are into the archive.

Subsequent requests for the mobile representation of [m.cnn.com](http://m.cnn.com) are identical to the first request, except they are now for the mobile representation presented to the user. To quantify the differences, the desktop representation contains 201 links, while the mobile representation contains only 58 links. These link sets are mutually exclusive, with each link pointing to specific resources (such as box-scores and gamecasts) while the desktop representation links to higher-level resources such as narratives that include box-scores and may have links to gamecasts). A user may review news articles or other content on a mobile device and be unable to recall the article in an archive. To capture and record the complete set of content at <http://cnn.go.com>, each of these different representations, both mobile and desktop, need to be stored in Web archives.

## RQ2: How do Web browser APIs compare in potential functionality to the capabilities of archival crawlers?

In this work, we explore the issue of personalized representations in Web archives, propose a framework to solve this problem, and present a proof-of-concept prototype

called mementos (identified by URI-M) to a canonical representation. This prototype extends the description of mementos from only 'when' they were archived (temporal dimension) to 'where' and 'how' (Geoloc and browser environments). Users can then browse between mementos based on temporal or environmental dimensions.

#### Personalized, Anonymous Representations

Dynamic and personalized representations of Web 2.0 resources that are generated by technologies such as JavaScript can differ greatly depending on several factors. For example, some sites attempt to provide alternate representations by interpreting the user-agent portion of the HTTP GET headers and use content negotiation to determine which representation to return.

We ran a pair of limited crawls of the [cnn.com](http://cnn.com) front page with Heritrix 2.1 and then accessed the mementos captured by Heritrix with a desktop Mac and an Android phone. The first crawl captured the [cnn.com](http://cnn.com) front page and specified a desktop version of the Mozilla browser as the user-agent in the header string, as seen in Figure 1. The resulting Web Archive (WARC) file [24] is viewed in a local installation of the Wayback Machine [20] and is shown in Figures 3(a) and 3(c).

The second crawl captured the [cnn.com](http://cnn.com) front page and specified an iPhone version of the Mozilla browser as the user-agent string in the header, as seen in Figure 2. The resulting WARC file [24] is shown in Figures 3(b) and 3(d). The mobile and desktop representations differ in archives, but their relationship as permutations of each other is never recorded nor seen by us; a user of the Wayback Machine may not understand how these representations are generated since they are identified by the same URI-R. We refer to these offering representations of the same URI-R built with differing environments as personalized representations of the resource R.

The headers in Figures 1 and 2 reference the user-agent string with <http://yourdomain.com>, which is a place holder for the URI for whom the crawl is being executed. For example, a crawl originating from Old Dominion University's Computer Science department would read <http://www.cs.odu.edu/>.

HTTP/1.0

HTTP-Type: request

HTTP-User-Agent-URI: <http://www.cnn.com>

HTTP-Date: 2013-03-05T16:57:00Z

A Framework for Aggregating Public and Private Web Archives

February 14, 2019

Mat Kelly

# Preliminary Research

1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. **JCDL 2014 - Mink**
5. JCDL 2014 - Archival Acid Test
6. JCDL 2014 - Not All Mementos
7. IJDL 2015 - Impact of JavaScript
8. IJDL 2015 - Not All Mementos
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary Wayback (ipwb)
11. TPDL 2016 - ipwb extended
12. JCDL 2017 - WAIL Electron
13. JCDL 2017 - ServiceWorker Replay
14. JCDL 2017 - Impact of Canonicalization
15. JCDL 2018 - ArchiveNow
16. JCDL 2018 - Replay Banners
17. JCDL 2018 - A Framework...



## Mink: Integrating the Live and Archived Web Viewing Experience Using Web Browsers and Memento

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

We describe Mink, a new web browser extension that provides a different model for integration of the live and archived web. While a user browses the live web, Mink actively queries the archives and reports other instances of the page in the archives without requiring active querying by the user. Further, by querying the archives dynamically and asynchronously, a user can view the extent to which the currently viewed page on the live web has been archived and proactively submit a request to various archives using an overlay

on the live web page and a simple interface.

### Categories and Subject Descriptors

H.3.7 [Online Information Systems]: Web Content Management—Archives

## Provides a seamless viewing experience

### 1. INTRODUCTION

To better integrate the past and live web, implementations of the Memento framework [1] provide the facilities to query the archives (using URI and HTTP Accept-Datetime headers as parameters) to provide resources on the past web

### RQ2: How do Web browser APIs compare in potential functionality to the capabilities of archival crawlers?

While the Memento framework is designed for use in mobile contexts, the native browsers give an *ad hoc* feel of requiring a separate client (e.g., an app). Retaining use to the client normally used to view the live web (i.e., a web browser) is more fluid to the user. The memento browser

### RQ5: How can Memento aggregators indicate that private Web archive content requires special handling to be replayed, despite being aggregated with publicly available Web archive content?

web. We have developed a new browser extension, Mink<sup>2</sup>, that instead uses an unobtrusive alert model to remind the user about the past. This model allows the user to quickly poll through the mementos available while maintaining the paradigm of relying on what is returned by the server to determine whether the user stays in the past or returns to the present. The additional feature of allowing the user to seamlessly jump from the past to the present while maintaining a quick return to the past makes Mink's approach unique.

### 2. ANOTHER APPROACH

The browser-based model of accessing archives is preferable to that of mobile apps. Bombarding the Memento proxy with requests for content negotiation is computationally expensive. We have implemented a TimeGate-based model using the TimeGate and TimeMapper services to reduce the number of requests to a URL, which reduces the negotiation complexity and still provides a more integrative model between the live web and the archived web using the user's web browser.

We chose the Google Chrome browser extension environment due to the browser's popularity, but the logic is simple enough to be ported to other browsers when a user loads

a Memento TimeMap in return. While processing the results at the bottom right of the browser viewport and provides a "spinning" animation until the TimeMap is received (Figure 1). If the TimeMap is paginated with a reference to a subsequent TimeMap, a button is provided to the user to invoke the iterative fetching of the TimeMap. The user can stop the iteration (either explicitly or to stop iterating at a number of times set in the configuration) or to stop the animation by the user. Regardless of how many TimeMaps are requested, the user can observe how well pages are archived without needing to commit to browsing the archived web nor to proactively submit a request to the archives to receive this archival metadata about the live web.

Once a user has accessed an archived page using Mink, the interface provides an additional button that allows the user to return to the live web with a single click for easy comparison.

<sup>2</sup>Named for Minkowski Space

<sup>3</sup>Available at <https://github.com/machawk1/mink>

# Preliminary Research

1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. JCDL 2014 - Mink
5. **JCDL 2014 - Archival Acid Test**
6. JCDL 2014 - Not All Mementos
7. IJDL 2015 - Impact of JavaScript
8. IJDL 2015 - Not All Mementos
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary Wayback (ipwb)
11. TPDL 2016 - ipwb extended
12. JCDL 2017 - WAIL Electron
13. JCDL 2017 - ServiceWorker Replay
14. JCDL 2017 - Impact of Canonicalization
15. JCDL 2018 - ArchiveNow
16. JCDL 2018 - Replay Banners
17. JCDL 2018 - A Framework...

## 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.

Categories and Subject Headings: H.3.3 [Information Storage and Retrieval]: Digital Libraries and Archives

### General Terms

Experimentation, Standardization, Verification

### Keywords

Web Crawler, Web Archiving, Digital Preservation

## RQ1: What sort of content is difficult to capture and replay for preservation from the perspective of a Web browser?

Archival crawlers and web archiving tools have a primary goal of capturing web pages so they can be “replayed” at a later date. Web archiving tools access these pages on the live web in a manner similar to tools used by search engines (crawlers) and preserve the pages in a format that allows the data and contextual information about the crawl to be re-experienced. These “archival crawlers” take different approaches in digital preservation and thus their capability and scope vary.

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”<sup>1</sup> helped to highlight the differences in how each browser handled various features of the standard 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 render the live page, this is not always the case when the archived version of the page is rendered. This difference can be used to highlight the features that archival crawlers are lacking compared to web browsers and thus emphasize the deviations that will occur in web archives compared to what a user would expect from a digitally preserved web page.

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<sup>1</sup>) is the de facto standard.

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

<sup>1</sup><https://github.com/iipc/openwayback>

# Preliminary Research

1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. JCDL 2014 - Mink
5. JCDL 2014 - Archival Acid Test
6. **JCDL 2014 - Not All Mementos**
7. IJDL 2015 - Impact of JavaScript
8. IJDL 2015 - Not All Mementos
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary Wayback (ipwb)
11. TPDL 2016 - ipwb extended
12. JCDL 2017 - WAIL Electron
13. JCDL 2017 - ServiceWorker Replay
14. JCDL 2017 - Impact of Canonicalization
15. JCDL 2018 - ArchiveNow
16. JCDL 2018 - Replay Banners
17. JCDL 2018 - A Framework...

A Framework for Aggregating Public and Private Web Archives  
February 14, 2019  
*Mat Kelly*

## 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, mckelly, hany, mweigle, mln}@cs.odu.edu

### ABSTRACT

Web archives do not capture every resource on every page that they attempt to archive. This results in archived pages missing a portion of their embedded resources. These embedded resources have varying historic, utility, and importance values. The proportion of missing embedded resources does not provide an accurate measure of their impact on the Web page; some embedded resources are more important to the utility of a page than others. We propose a method to

measure the relative value of embedded resources and assign a damage rating to archived pages as a way to evaluate archival success. In this paper, we show that Web users' perceptions of damage are not accurately estimated by the proportion of missing embedded resources. The proportion of missing embedded resources is not a good indicator of resource damage than a random selection. We propose a damage rating algorithm that provides closer alignment to Web user perception, providing an overall improved agreement with users on memento damage by 17% and an improvement by 31% if the mementos are now similarly damaged. We use our algorithm to measure damage in the Internet Archive, showing that it is getting better at mitigating damage over time (going from 0.16 in 1998 to 0.13 in 2013). However, we show that a greater number of important embedded resources (2.05 per memento on average) are missing.

### Not all missing resources are created equal

Throughout this paper we use Memento Framework terminology. Memento [26] is a framework that allows web users to browse in the temporal dimension by aggregating the versions of a single resource. Original (or live web) resources are identified by URI-R, and 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.

The research is motivated by three factors. First, we want to measure the impact of missing resources on the user's satisfaction (i.e., the utility of mementos). Using our proposed algorithm, we show that the utility of the memento is missing (e.g., a main image or video essential to the user's understanding of the page), or the missing embedded resource is a spacer image or a small button logo that contributes little to the memento's utility for the user. We propose a method of weighting embedded resources in a memento according to importance. We show that this is an

improved damage rating over an unweighted count of missing resources. Second, we want to measure the impact of missing resources on the Internet Archive. The Internet Archive is a large collection of digital objects. We propose a method of calculating the damage and to show an improvement over the unweighted count of missing resources.

Second, we use our algorithm to assess the damage of mementos in the Internet Archive. We use the unweighted measure of damage as the proportion of missing embedded resources to all requested resources ( $M_m$ ) and compare it to our algorithm's calculation of damage ( $D_m$ ).

Third and finally, we measure damage in the Internet

**RQ1:** What sort of content is difficult to capture and replay for preservation from the perspective of a Web browser?

#### General Terms

Design, Experimentation, Measurement

#### Keywords

Web Architecture, HTTP, Web Archiving, Digital Preservation, Memento, TimeMaps



Web Archiving, Digital Preservation

**BEST STUDENT PAPER AWARD**

at JCDL 2014

# Preliminary Research

1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. JCDL 2014 - Mink
5. JCDL 2014 - Archival Acid Test
6. JCDL 2014 - Not All Mementos
- 7. IJDL 2015 - Impact of JavaScript**
8. IJDL 2015 - Not All Mementos
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary Wayback (ipwb)
11. TPDL 2016 - ipwb extended
12. JCDL 2017 - WAIL Electron
13. JCDL 2017 - ServiceWorker Replay
14. JCDL 2017 - Impact of Canonicalization
15. JCDL 2018 - ArchiveNow
16. JCDL 2018 - Replay Banners
17. JCDL 2018 - A Framework...

## 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  
 © Springer-Verlag Berlin Heidelberg 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 user scrolls down the page). In an effort to understand why mementos (archived versions of live resources) in today's archives vary in completeness and sometimes pull content from the live web, we present a study of web resources and archival tools. We used a collection of URIs shared over Twitter and a collection of URIs curated by Archive-It in our investigation. We created local archived versions of the URIs from the Twitter and Archive-It sets using WebCite, wget, and the Heritrix crawler. We found that only 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.

## Missing JavaScript has big ramifications

**Keywords** Web archiving · Web archiving · Digital preservation

### 1 Introduction

How well can we archive the web? This is a question that is becoming increasingly important and more difficult to answer. Additionally, this question has significant impact on web users [40, 41] and commercial and government organizations [42].

The web has gone through a gradient of changes fueled by the introduction of client-side technologies. Early websites were primarily static. With the introduction of web technologies, however, do the gradients become blurred and more intertwined. JavaScript, which executes on the client, provides additional features for the web user, enabling or increasing interactivity, client-side state changes, and personalized representations. These additional features offer an enhanced browsing experience for the user.

JavaScript has enabled a wide-scale migration from web pages to web applications. This migration continued with the introduction of Ajax (first introduced in 2005 [28]), which combined multiple technologies to give web pages the ability to perform asynchronous client-server interactions after the HTML is loaded. The first wide-scale implementation of Ajax was in Google Maps in 2005, but Ajax was officially added as a standard in 2006 [70]. While archival tools per-

**RQ1:** What sort of content is difficult to capture and replay for preservation from the perspective of a Web browser?

The web has gone through a gradient of changes fueled by the introduction of client-side technologies. Early websites were primarily static. With the introduction of web technologies, however, do the gradients become blurred and more intertwined. JavaScript, which executes on the client, provides additional features for the web user, enabling or increasing interactivity, client-side state changes, and personalized representations. These additional features offer an enhanced browsing experience for the user.

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

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

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

# Preliminary Research

1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. JCDL 2014 - Mink
5. JCDL 2014 - Archival Acid Test
6. JCDL 2014 - Not All Mementos
7. IJDL 2015 - Impact of JavaScript
- 8. IJDL 2015 - Not All Mementos**
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary Wayback (ipwb)
11. TPDL 2016 - ipwb extended
12. JCDL 2017 - WAIL Electron
13. JCDL 2017 - ServiceWorker Replay
14. JCDL 2017 - Impact of Canonicalization
15. JCDL 2018 - ArchiveNow
16. JCDL 2018 - Replay Banners
17. JCDL 2018 - A Framework...

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

Justin F. Brunelle<sup>1</sup> · Mat Kelly<sup>1</sup> · Hany SalahEldeen<sup>1</sup> · Michele C. Weigle<sup>1</sup> · Michael L. Nelson<sup>1</sup>

Received: 3 December 2014 / Revised: 22 April 2015 / Accepted: 22 April 2015 / Published online: 6 May 2015  
 © Springer-Verlag Berlin Heidelberg 2015

**Abstract** Web archives do not always capture every resource on every page that they attempt to archive. This results in archived pages missing a portion of their embedded resources. These embedded resources have varying historic, utility, and importance values. The proportion of missing embedded resources can vary greatly in terms of their impact on the Web page; some embedded resources are more important than others. In this paper, we propose a method to estimate the relative value of embedded resources and assign a damage rating to archived pages as a way to evaluate archival success. In this paper, we show that web users' perceptions of damage are not accurately estimated by the proportion of missing embedded resources. In fact, the proportion of missing embedded resources is a less accurate estimate of resource damage than a random selection. We propose a damage rating algorithm that provides closer alignment to web user perception, providing over 17 % and an improvement by 51 % if the mementos have a damage rating. The impact of missing embedded resources on getting better at mitigating damage over time, going from a damage

## Is the metric for missing resources applicable across Web?

rating of 0.16 in 1998 to 0.13 in 2013). However, we show that a greater number of important embedded resources (2.05 per memento on average) are missing over time. Alternatively, the damage in Webcite is increasing over time (going from 0.375 in 2007 to 0.475 in 2014), while the missing embedded resources are decreasing over time (the resources are missing on average). Finally, we investigate the impact of JavaScript on the damage of Webarchives, showing that a crawler that can archive JavaScript-dependent representations will reduce memento damage by 13.5 %.

**Keywords** Web architecture · Web archiving · Digital preservation · Memento damage

### 1 Introduction

Archives are collections of historical documents and static Web content. People (and robots) use archives like the Library of Congress [1] and the Internet Archive [2] to access and store Web material [20, 21]. Archives are used for a variety of purposes and in a variety of ways [2]. However, the resources being stored by Webarchives are not complete; embedded resources are sometimes missing from an archived Web page [6]. Missing embedded resources return a non-200 HTTP status (e.g., 404, 503) when their URI is dereferenced.

Archivists work to ensure archives are as complete—and as high quality—as possible. Through identifying sources of missing content or archival difficulties, archivists can address archival challenges by taking steps to adjust processes or to fill in gaps in archive collections.

Reyes et al. identified current efforts within several archives to assess their archival collections [4]. Of the archivists sampled, 61 % confirmed that their goal is to assess the quality of every Web page captured, 43 % assess quality

✉ Justin F. Brunelle  
 jbrunelle@cs.odu.edu  
 Mat Kelly  
 mckelly@cs.odu.edu  
 Hany SalahEldeen  
 hany@cs.odu.edu  
 Michele C. Weigle  
 mweigle@cs.odu.edu  
 Michael L. Nelson  
 mln@cs.odu.edu

<sup>1</sup> Department of Computer Science, Old Dominion University, Norfolk, VA 23529, USA

# Preliminary Research

1. JCDL 2012 - WARCreate
  2. TPDL 2013 - Change in Archivability
  3. DLib 2013 - Method for Identifying
  4. JCDL 2014 - Mink
  5. JCDL 2014 - Archival Acid Test
  6. JCDL 2014 - Not All Mementos
  7. IJDL 2015 - Impact of JavaScript
  8. IJDL 2015 - Not All Mementos
  9. **JCDL 2015 - Mobile Mink**
  10. JCDL 2016 - InterPlanetary Wayback (ipwb)
  11. TPDL 2016 - ipwb extended
  12. JCDL 2017 - WAIL Electron
  13. JCDL 2017 - ServiceWorker Replay
  14. JCDL 2017 - Impact of Canonicalization
  15. JCDL 2018 - ArchiveNow
  16. JCDL 2018 - Replay Banners
  17. JCDL 2018 - A Framework...

A Framework for Aggregating Public and Private Web Archives  
February 14, 2019  
*Mat Kelly*

## Mobile Mink: Merging Mobile and Desktop Archived Webs

Vesley Jordan<sup>1</sup>, Mat Kelly<sup>2</sup>, Justin F. Brunelle<sup>2,3</sup>, Laura Vobrak<sup>1</sup>, Michele C. Weigle<sup>2</sup>, and Michael L. Nelson<sup>2</sup>

## New Horizons Regional Education Center Governor's School for Science and Technology

<sup>2</sup> Old Dominion University, Department of Computer Science

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 URLs and provides the user an aggregated TimeMap of both mobile and desktop mementos. Mobile Mink also allows users to submit mobile and desktop URLs 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 |

# Recoupled mobile and design experience measurement

### Keywords

## 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 the page.

**RQ2:** How do Web browser APIs compare in potential functionality to the capabilities of archival crawlers?

by URI-R. Archived versions of URI-IDs are called *mementos* and are identified by URI-M. Memento timelines 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 popularity, and ubiquity and the mobile web continue to expand and become more prevalent [9].



Permission to \_\_\_\_\_ 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). Copyright is held by the author(s)/owner(s).

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

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 view of the mobile and desktop versions of the same resource, and provides the user with 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].

# Webs

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 allows the user to navigate between past and present web pages. Mobile Mink also allows the user to submit mobile and desktop URL-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 bar. One of the options is called “view as” which is a submenu of options, one of which is the option to “Share” the page (Figure 1(a)). This option will add 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 URLs 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 URL or a mobile URL. Second, if the URL is a desktop URL, Mobile Mink translates the URI to a mobile URL. Finally, Mobile Mink translates the URI to a mobile URL and performs some URI modifications and then sends the modified URL to JCDL's work [7] and test for the most recent version of the live web (i.e., returns an HTTP 200 response) and in the archives (returns a TimeMap of commands > 0 from the archive's 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

# Preliminary Research

1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. JCDL 2014 - Mink
5. JCDL 2014 - Archival Acid Test
6. JCDL 2014 - Not All Mementos
7. IJDL 2015 - Impact of JavaScript
8. IJDL 2015 - Not All Mementos
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary Wayback (ipwb)
11. TPDL 2016 - ipwb extended
12. JCDL 2017 - WAIL Electron
13. JCDL 2017 - ServiceWorker Replay
14. JCDL 2017 - Impact of Canonicalization
15. JCDL 2018 - ArchiveNow
16. JCDL 2018 - Replay Banners
17. JCDL 2018 - A Framework...

A Framework for Aggregating Public and Private Web Archives  
 February 14, 2019  
*Mat Kelly*

Sawood Alam, Mat Kelly, and Michael L. Nelson  
 Old Dominion University, Department of Computer Science, Norfolk VA, 23529, USA  
 {salam,mkelly,mln}@cs.odu.edu

## ABSTRACT

To facilitate permanence and collaboration in web archives, we built InterPlanetary Wayback to disseminate the contents of WARC files into the IPFS network. IPFS is a peer-to-peer content-addressable file system that inherently allows deduplication and facilitates opt-in replication. We split the header and payload of WARC response records before disseminating into IPFS to leverage the deduplication, build a CDXJ index, and combine them at the time of replay. From a 1.0 GB sample Archive-It collection of WARCs containing 21,994 mementos, we found that on an average, 570 files can be indexed and disseminated into IPFS per minute. We also found that in our naive prototype implementation, replay took on an average 370 milliseconds per request.

## 1. INTRODUCTION

The recently created InterPlanetary File System (IPFS) [2] is showing the potential to facilitate data persistence through a peer-to-peer distributed file system. In this paper, we introduce the InterPlanetary Wayback (ipwb), that partitions, indexes, and deploys the payloads of archival WARC files into IPFS peer-to-peer “permanent web” [1] to enable redundant preservation and replay.

The Web ARCHive (WARC) format is an ISO standard<sup>2</sup> to store web archive content in a compressed record-based file. LA’s web crawler, Heritrix [3], generates WARC files to be read and the content re-experienced in an archival replay system. OpenWayback<sup>3</sup> (written in Java) and pwby<sup>4</sup> (written in Python) are two such replay systems. We leverage

and extend the CDXJ file format<sup>5</sup> to store the metadata and payloads of WARC files in a single JSON object. The CDXJ file holds one index record. The line begins with a SURT<sup>6</sup> URI and datetime followed by a single-line JSON block that stores reference to the content and other arbitrary metadata (Figure 1). We utilize the last field in a CDXJ record (a JSON object) to store the original URL, the file size, and the content digest of the file. This is done because neither the payload nor the content reference in the CDXJ file is encoded into a single field called “locator” using a CTRV scheme<sup>6</sup>.

<sup>2</sup><https://github.com/oduwSDL/ipwb>

<sup>3</sup><https://github.com/lpbc/openwayback>

<sup>4</sup><https://github.com/oduwSDL/pwby>

<sup>5</sup><https://www.w3.org/TR/uri-clarification/>

<sup>6</sup><https://www.w3.org/TR/uri-clarification/#locators>

**RQ4: How can content that was captured behind authentication signal to Web archive replay systems that requires special handling?**

For instance, if a digital or hard copies of part or all of this work for personal or

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/authors.

JCDL '16 June 19-23, 2016, Newark, NJ, USA

© 2016 Copyright held by the owner/authors.

ACM ISBN 978-1-4503-4229-2/16/06...\$15.00

DOI: <http://dx.doi.org/10.1145/2910896.2925467>

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

Figure 1: A single-line CDXJ record template, shown on multiple lines for readability

about WARC records within IPFS (i.e., the content digest needed for lookup in IPFS).

IPFS is a content addressable peer-to-peer distributed file system [2]. By extracting the HTTP response body (henceforth “payload”) from the records within a WARC file, IPFS allows us to generate a signature uniquely representative of this content. The payload can then be pushed into the IPFS system and retrieved at a later date when the URI-M is queried. Content addressability allows us to store the content of the WARC file in IPFS without propagating the content in the peer-to-peer network.

## 2. IMPLEMENTATION

CDXJ is a text-based file format that we utilize to store index records of web archive content and payloads. Index records are stored in a CDXJ file, which is a CTRV file. A CTRV file is composed of three fields: the URI-R, the DATETIME, and the locator. The locator field contains the URL of the content, the content digest, and the file size. The locator field is encoded into a single field called “locator” using a CTRV scheme<sup>6</sup>.

In designing ipwb, it was critical to consider the HTTP header returned at crawl time separately from the HTTP response body. The HTTP response header’s content will change with every capture, as the datetime returned from a server is temporally dependent. Compare this to the response body, which very often contains the same content on each capture. The HTTP response body is mapped to IPFS, where every IPFS hash would be unique, nullifying the potential for replay. The ipwb design decision is that the state of the art of web archive replay systems do not consider the WARC request record upon replay. While including request records may be useful in the future, for instance, to take into account the user-agent originally used to view the live website, WARC content is currently fully replayable without preserving the request records.

<sup>5</sup>[http://crawler.archive.org/articles/user\\_manual/glossary.html#surt](http://crawler.archive.org/articles/user_manual/glossary.html#surt)

<sup>6</sup><https://www.w3.org/TR/uri-clarification/>

# Preliminary Research

1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. JCDL 2014 - Mink
5. JCDL 2014 - Archival Acid Test
6. JCDL 2014 - Not All Mementos
7. IJDL 2015 - Impact of JavaScript
8. IJDL 2015 - Not All Mementos
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary Wayback (ipwb)
11. TPDL 2016 - ipwb extended
12. JCDL 2017 - WAIL Electron
13. JCDL 2017 - ServiceWorker Replay
14. JCDL 2017 - Impact of Canonicalization
15. JCDL 2018 - ArchiveNow
16. JCDL 2018 - Replay Banners
17. JCDL 2018 - A Framework...

Mat Kelly, Sawood Alam, Michael L. Nelson, and Michele C. Weigle

Old Dominion University, Department of Computer Science  
Norfolk VA, 23529, USA  
[{mkelly,salam,mln,mweigle}@cs.odu.edu](mailto:{mkelly,salam,mln,mweigle}@cs.odu.edu)

**Abstract.** We have integrated Web ARChive (WARC) files with the peer-to-peer content addressable InterPlanetary File System (IPFS) to allow the payload content of web archives to be easily propagated. We also provide an archival replay system extended from pywb to fetch the WARC content from IPFS and re-assemble the originally archived HTTP responses for replay. From a 1.0 GB sample Archive-It collection of WARCs containing 21,994 mementos, we show that extracting and indexing the HTTP response content of WARCs containing IPFS lookup hashes takes

How much does it cost to have  
resilient personal archives?  
IPFS

## 1 Motivation

The recently created InterPlanetary File System (IPFS) [9] facilitates data persistence and access. While web archives like Internet Archive (IA) provide a system and means of preservation, IPFS provides a way to signal to the resilience of the organization and the availability of the data [5]. In this paper, we present a scheme and software prototype<sup>1</sup>, InterPlanetary Wayback (ipwb), that partitions, indexes, and deploys the payloads of archival data records into the IPFS peer-to-peer “permanent web” for sharing and offsite massive

**RQ4: How can content that was captured behind authentication signal to Web archive replay systems that it requires special handling?**

**RQ6: What kinds of access control do users who create private Web archives need to regulate access to their archives?**

<sup>2</sup> Background and Related Work

The Web ARChive (WARC) format is an ISO standard [4] to store live web archive content in a concatenated record-based file. IA’s web crawler, Heritrix [7], generates WARC files to be read and the content re-experienced in an archival

<sup>1</sup> <https://github.com/oduwsdl/ipwb>

# Preliminary Research

1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. JCDL 2014 - Mink
5. JCDL 2014 - Archival Acid Test
6. JCDL 2014 - Not All Mementos
7. IJDL 2015 - Impact of JavaScript
8. IJDL 2015 - Not All Mementos
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary Wayback (ipwb)
11. TPDL 2016 - ipwb extended
- 12. JCDL 2017 - WAIL Electron**
13. JCDL 2017 - ServiceWorker Replay
14. JCDL 2017 - Impact of Canonicalization
15. JCDL 2018 - ArchiveNow
16. JCDL 2018 - Replay Banners
17. JCDL 2018 - A Framework...

A Framework for Aggregating Public and Private Web Archives  
 February 14, 2019  
*Mat Kelly*

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.

## KEY WORDS

Personal Web Archiving

### ACM Reference format:

John A. Berlin, Mat Kelly, Michael L. Nelson, Michele C. Weigle. 2017. WAIL-Electron: Collection-Based Personal Web Archiving. In *Proceedings of Joint Conference on Digital Libraries, Toronto, Ontario, Canada, June 2017 (JCDL 17)*, 2 pages.  
 DOI: <https://doi.org/10.1145/3058896.3058900>

## 1 INTRODUCTION

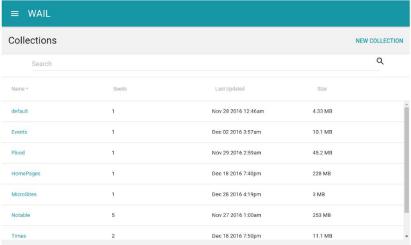
Subscription-based Web archiving services like Archive-it allow users with limited technical knowledge to create and replay personalized collections of Web content. Archive-it provides users with a simple interface to manage their collections and to archive them with WIAL. Similar to Archive-it, WebRecorder<sup>1</sup> which allows any user to create and manage personalized collections of Web archives. But unlike Archive-it, WebRecorder requires its user to manually drive the preservation process or upload content for replay while 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

<sup>1</sup><https://webrecorder.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.XXXX/XXXXX



Name	Seeds	Last updated	Size
default	1	Nov 29 2016 12:56am	4.23 MB
Events	1	Dec 02 2016 9:27pm	101 MB
Fluid	1	Nov 29 2016 2:35pm	45.2 MB
HomePage	1	Dec 18 2016 7:40pm	228 MB
Memento	1	Dec 26 2016 4:19pm	3 MB
Mobile	5	Nov 27 2016 1:10pm	250 MB
Tweets	2	Dec 16 2016 7:40pm	111 MB

Figure 1: Collections screen

## Archive from the desktop With higher fidelity than institutions

skill level. Even if a user is able to successfully set up these tools, they will need to learn how to use Heritrix and come up with their own means of associating the Web archives to each other for reuse. We believe that WAIL-Electron provides a better solution to both

Heritrix and Wayback while providing an interoperable mechanism for personal collection-based Web archiving from their personal computers. Users can create and add to these collections through WAIL-Electron with the software taking care of the details in managing the collections, crawls, and replay. We have integrated a native Chromium<sup>2</sup> 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.

## RQ2: How do Web browser APIs compare in potential functionality to the capabilities of archival crawlers?

2 WAIL

WAIL is the desktop application component of the original WAIL application [5]. 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 of seeds contained in the collection along with the collection's size and the last time it was 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

<sup>2</sup><https://www.chromium.org/>

<sup>3</sup><http://electron.atom.io/>

# Preliminary Research

1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. JCDL 2014 - Mink
5. JCDL 2014 - Archival Acid Test
6. JCDL 2014 - Not All Mementos
7. IJDL 2015 - Impact of JavaScript
8. IJDL 2015 - Not All Mementos
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary Wayback (ipwb)
11. TPDL 2016 - ipwb extended
12. JCDL 2017 - WAIL Electron
13. **JCDL 2017 - ServiceWorker Replay**
14. JCDL 2017 - Impact of Canonicalization
15. JCDL 2018 - ArchiveNow
16. JCDL 2018 - Replay Banners
17. JCDL 2018 - A Framework...

A Framework for Aggregating Public and Private Web Archives  
February 14, 2019  
*Mat Kelly*

## Client-side Reconstruction of Composite Mementos Using ServiceWorker

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

### ABSTRACT

We use the ServiceWorker (SW) web API to intercept HTTP requests for embedded resources and reconstruct Composite Mementos without the need for conventional URL rewriting typically performed by web archives. URL rewriting is a problem for archival replay systems, especially for URLs constructed by JavaScript, frequently resulting in incorrect URI references. By intercepting requests on the client using SW, we are able to strategically reroute instead of rewrite. Our implementation moves rewriting to clients, saving servers' computing resources and allowing servers to return responses more quickly. Our experiments show that retrieving the original instead of rewritten pages from the live archive takes time overhead by 35.66% and data overhead by 19.68%. Our system prevents Composite Mementos from leaking the live web while being easy to distribute and maintain.

### CCS CONCEPTS

•Information systems — World Wide Web;

### KEYWORDS

ServiceWorker, Memento, Composite Memento, Web Archive, Archival Replay

### ACM Reference format:

Sawood Alam, Mat Kelly, Michele C. Weigle, and Michael L. Nelson. 2017. Client-side Reconstruction of Composite Mementos. In *Proceedings of the 2017 Conference on Digital Libraries (Toronto, Ontario, Canada), June 2017*, 1–10. DOI: <https://doi.org/10.1145/3058295.3058305>

## RQ2: How do Web browser APIs compare in potential functionality to the capabilities of archival crawlers?

### 1 INTRODUCTION

ServiceWorker (SW) is a new client-side web API [11] that can be used to intercept all the network requests for embedded resources originating from web pages in its scope. A Composite Memento [2] is an archived HTML page along

Permission to make digital or hard copies of all or part 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 components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

*JCDL'17, Toronto, Ontario, Canada*

© 2017 Copyright held by the owner/author(s). Publication rights licensed to ACM. 000-0000-00-00/00/00...\$15.00  
DOI: [00.000/000.00](https://doi.org/10.1145/3058295.3058305)



Figure 1: Live Ad Zombie Leaks into Archived Page  
with all the embedded resources that are necessary to render the page correctly. Web archival replay systems rewrite embedded resources to point to their archival versions e.g., a reference to `external.example.net/logo.png` is changed to `internal.example.net/logo.png`.

We use SW API to reconstruct Composite Mementos from the originally captured data without any such URL rewriting. By intercepting requests on the client-side we are essentially rerouting instead of rewriting. Rerouting is an effective mechanism to block live web leakage, or "zombies" that might happen after executing potential JavaScript (JS), otherwise not discoverable by static analysis. For example, in Figure 1 the page was archived on September 10th, 2008, but when observed on September 28, 2017 it contained an ad from the 2012 president candidates [5]. Client-side rerouting also saves the cost of reconstructing the content on the client side, such as to include archival banners, hence,

there is no need to send extra data with each HTTP response. Client-side solutions such as Memento for Chrome<sup>1</sup> involve installing a browser add-on, which limits the adoption by users and adds the burden of maintaining the add-on while only available for Google Chrome users. Our exploratory technique works well when SW is supported. However, a server-side fallback is necessary for production usage to avoid the risk of zombies and broken references when SW is not supported.

Our experiments show that retrieving the original instead of rewritten pages from the Internet Archive (IA) reduces time overhead by 35.66% and data overhead by 19.68%. Our system prevents Composite Mementos from zombies while being easy to distribute and maintain. It is a lightweight and portable system that can be used with any Memento server such as a web archive or a Memento aggregator.

<sup>1</sup><http://bit.ly/memento-for-chrome>

# Preliminary Research

1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. JCDL 2014 - Mink
5. JCDL 2014 - Archival Acid Test
6. JCDL 2014 - Not All Mementos
7. IJDL 2015 - Impact of JavaScript
8. IJDL 2015 - Not All Mementos
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary Wayback (ipwb)
11. TPDL 2016 - ipwb extended
12. JCDL 2017 - WAIL Electron
13. JCDL 2017 - ServiceWorker Replay
14. JCDL 2017 - Impact of Canonicalization
15. JCDL 2018 - ArchiveNow
16. JCDL 2018 - Replay Banners
17. JCDL 2018 - A Framework...

## 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](mailto:{mkelly,lalkwai,salam,mln,mweigle}@cs.odu.edu)

Herbert Van de Sompel  
Los Alamos National Laboratory  
Los Alamos, New Mexico, USA  
[herberty@lanl.gov](mailto:herberty@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 random sample of 100 TimeMaps from each of the 20 sites, the 13 academic institutions, and the 13 large web sites' and two of the thirteen academic institutions' TimeMaps had a ratio of less than one, indicating that more than half of the URI-Ms in these TimeMaps result in redirects when queried.

### 1 INTRODUCTION

Web archives return TimeMaps with a list of URI-Ms for the HTTP transactions observed at archival time. TimeMaps have generally been used as a count of the number of representations available for a resource [6].

RQ3: What issues exist for capturing and replaying content behind authentication?

URI-Ms in the TimeMap that returns a HTTP Status OK. TimeMaps do not explicitly return a "count" value to indicate the number of mementos listed in the TimeMap that produce a non-redirecting HTTP status code when dereferenced. The heuristic of determining how many captures are represented by URI-Ms in a TimeMap is to count them without



Redirection in web archive can be attributed to the 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

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	5,605	0.942

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 inclusion of redirects when dereferencing, and  $DI$  is the ratio of  $M_{RC}$  to  $M_{TM}$ .

(URI-Ms) in a TimeMap are identifiers for archived HTTP transactions (e.g., transmission of HTTP 2XX, 3XX, 4XX, etc.) rather than identifiers for representations.

Based on the number of URI-Ms in a TimeMap not necessarily resolving to unique mementos when archival redirects are followed, we examined the mementos from contemporaneous TimeMaps from the same and different web sites to quantify the difference between the number of mementos available as reported by the TimeMap through naive "rel" counting heuristics to the temporally unique mementos identified once these mementos are dereferenced.

### 2 BACKGROUND AND RELATED WORK

URI canonicalization associates differently formatted URIs [4] and allows after-the-fact clustering of URIs that likely reference the same resource. As URI schemes from a Web

BEST POSTER AWARD

HTTP [1] used memento redirection patterns relating to HTTP 3XX 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.

# Preliminary Research

1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. JCDL 2014 - Mink
5. JCDL 2014 - Archival Acid Test
6. JCDL 2014 - Not All Mementos
7. IJDL 2015 - Impact of JavaScript
8. IJDL 2015 - Not All Mementos
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary Wayback (ipwb)
11. TPDL 2016 - ipwb extended
12. JCDL 2017 - WAIL Electron
13. JCDL 2017 - ServiceWorker Replay
14. JCDL 2017 - Impact of Canonicalization
15. JCDL 2018 - ArchiveNow
16. JCDL 2018 - Replay Banners
17. JCDL 2018 - A Framework...

## ArchiveNow: Simplified, Extensible, Multi-Archive Preservation

Mohamed Aturban, Mat Kelly, Sawood Alam, John A. Berlin,  
Michael L. Nelson, and Michele C. Weigle  
Old Dominion University  
Department of Computer Science  
Norfolk, Virginia, USA  
{maturban,mkelly,salam,jberlin,mln,mweigle}@cs.odu.edu

### ABSTRACT

ArchiveNow is a Python module for preserving web pages in on-demand web archives. This module allows a user to submit a URI of a web page for archiving at several configured web archives. Once the web page is captured, ArchiveNow provides the user with links to the archived copies of the web page. ArchiveNow is initially configured to use four archives but is easily configurable to add or remove other archives. In addition to pushing web pages to public archives, ArchiveNow, through the use of *Wget* and *Squidwarc*, allows users to generate local WARC files, enabling them to create their own personal and private archives.

### CCS CONCEPTS

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

**Create & Submit archives through CLI and local WARC generation**

### KEYWORDS

Web Archiving, Memento, WARC

### ACM Reference Format:

Mohamed Aturban, Mat Kelly, Sawood Alam, John A. Berlin, Michael L. Nelson, and Michele C. Weigle. 2018. ArchiveNow: Simplified, Extensible, Multi-Archive Preservation. 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, 2 pages. <https://doi.org/10.1145/3197026.3203880>

### 1 INTRODUCTION

Preserving a page in only one site we archive is risky. Archives can fail, become unavailable, or become unreliable or permanently unreachable. Thus, preserving web pages in multiple locations is important. For example, Kelly et al. [6] built *Mink*, a Google Chrome extension

that notifies a user of any available archived copies for a viewed page and suggests to archive the page in three archives. Welsh [10] developed several tools intended to archive news-related resources. For example, Welsh's *Savemy.news* ([www.savemy.news](http://www.savemy.news)) saves web pages in two archives. Users of this service are required to create accounts. In addition to *Savemy.news*, Welsh built three

permissions-based tools to reward copies. If part or all of this work for personal use is given away without fee provided that copies are not made or distributed for profit or commercial advantage and that on the copy or copies the full URL and the components of this work must be honored. For all other uses, such as derivative works, prior permission is required from the copyright holders.

*JCDL '18, June 3–7, 2018, Fort Worth, TX, USA*  
© 2018 Copyright held by the owner/authors.  
ACM ISBN 978-1-4503-5178-2/18/06.  
<https://doi.org/10.1145/3197026.3203880>

```
% archivenow --all --cc_api_key=7e..3f http://money.cnn.com/2018/01/27/technology/future/spacex-falcon-heavy-everything-you-need-to-know/index.html
{
  "uri": "http://money.cnn.com/2018/01/27/technology/future/spacex-falcon-heavy-everything-you-need-to-know/index.html",
  "request_datetime": "20180129094723",
  "mementos": [
    "archive.org": "https://web.archive.org/web/20180129094728/http://money.cnn.com/2018/01/27/technology/future/spacex-falcon-heavy-everything-you-need-to-know/index.html",
    "archive.is": "https://archive.is/hr41S",
    "webcitation.org": "http://www.webcitation.org/978puj3TC2"
  ]
}
```

**webcitation.org**: <http://www.webcitation.org/978puj3TC2>

**archive.org**: <https://web.archive.org/web/20180129094728/http://money.cnn.com/2018/01/27/technology/future/spacex-falcon-heavy-everything-you-need-to-know/index.html>

**archive.is**: <https://archive.is/hr41S>

**webcitation.org**: <http://www.webcitation.org/978puj3TC2>

**Create & Submit archives through CLI and local WARC generation**

separate on libraries to interact with on-demand archiving services. *Webrecorder* [9] and *WARCreate* [7] can be used to generate WARC files [5], but the only way to use these tools is through a web browser. *ArchiveNow* can save pages in four web archives, generate WARC files, and allows customization of the set of archives used to preserve the web. *ArchiveNow* does not require users to have an account and can be run through the command-line (CLI), a web-based user interface (UI), a self-contained Docker container, or as a Python module. *ArchiveNow* is available for download at <https://github.com/maturban/ArchiveNow>.

**MULTI-ARCHIVE PRESERVATION**

*ArchiveNow* is designed to make it easy for anyone to use it as a user for archiving at the following four archives: the Internet Archive (IA) at [archive.org](http://archive.org), Archive.is ([archive.is](http://archive.is)), Perma ([perma.cc](http://perma.cc)), and WebCite ([webcitation.org](http://webcitation.org)). Figure 1 shows an example of running *ArchiveNow* to request capturing a web page by all configured archives. The value of `--cc_api_key` is an API key required by Perma. The user can select one or more archives by replacing `--all` with the corresponding archive identifiers, such as `--ia` for Internet Archive, `--perma` for Perma, `--wclite`, and `--cc` for WebCite.

**BEST POSTER AWARD**

**at JCDL 2018**

For more information about *ArchiveNow*, see the GitHub page [1].

The UI shown in Figure 3. A full list of options for running *ArchiveNow* is available on GitHub [1].

**BEST POSTER AWARD**

**at JCDL 2018**

For more information about *ArchiveNow*, see the GitHub page [1].

The UI shown in Figure 3. A full list of options for running *ArchiveNow* is available on GitHub [1].

# Preliminary Research

1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. JCDL 2014 - Mink
5. JCDL 2014 - Archival Acid Test
6. JCDL 2014 - Not All Mementos
7. IJDL 2015 - Impact of JavaScript
8. IJDL 2015 - Not All Mementos
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary Wayback (ipwb)
11. TPDL 2016 - ipwb extended
12. JCDL 2017 - WAIL Electron
13. JCDL 2017 - ServiceWorker Replay
14. JCDL 2017 - Impact of Canonicalization
15. JCDL 2018 - ArchiveNow
16. JCDL 2018 - Replay Banners
17. JCDL 2018 - A Framework...

## Unobtrusive and Extensible Archival Replay Banners Using Custom Elements

Sawood Alam, Mat Kelly, Michele C. Weigle, and Michael L. Nelson

Old Dominion University

Department of Computer Science

Norfolk, Virginia, USA

{salam,mkelly,mweigle,mln}@cs.odu.edu

### ABSTRACT

We compare and contrast three different ways to implement an archival replay banner. We propose an implementation that utilizes *Custom Elements* and adds some unique behaviors, not common in existing archival replay systems, to enhance the user experience. Our approach has a minimal user interface footprint and resource overhead while still providing rich interactivity and extended on-demand provenance information about the archived resources.

### CCS CONCEPTS

• Information systems → Digital libraries and archives; • Human-centered computing → User interface design;

### KEYWORDS

Memento; Archival replay; archival navigation; archival banner; Memento

### ACM Reference Format:

Sawood Alam, Mat Kelly, Michele C. Weigle, and Michael L. Nelson. 2018. Unobtrusive and Extensible Archival Replay Banners Using Custom Elements. 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, 2 pages.  
<https://doi.org/10.1145/3197026.3203881>

### 1 MOTIVATION

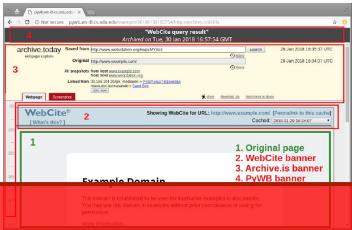
Web archival replay systems express that a user is interacting with a *memento* (an archived representation of a resource) by adding an archival banner. Archival banners provide metadata about both the memento and the original page, such as the timestamp, the URL, and the title. They also provide the user with the ability to interactively replicate the live web experience when viewing a memento, because resources and hyperlinks are referenced to the web archive and not the live web. Any component injection in the page makes it different from the original and may consume additional screen real estate. We illustrate this in Figure 1 by archiving example.com in its entirety. The original page contains a toolbar with links to other websites. When we view the memento, the toolbar is removed, and only the proposed archival banner is present. This is a potential problem for crawlers that rely on the original page for their analysis. If a classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, or that copies bear this notice and the full citation on the first page. Copyright rights for third party components of this work may be reserved. For all other uses, contact the owner/author(s).  
JCDL '18, June 3–7, 2018, Fort Worth, TX, USA  
© 2018 Copyright held by the owner/author(s).  
ACM ISBN 978-1-4503-5178-2/18/06.  
<https://doi.org/10.1145/3197026.3203881>

**RQ1: What sort of content is difficult to capture and replay for preservation from the perspective of a Web browser?**

The three primary ways to serve an archival banner with an archived web page that shares the rendering space with the memento. Browser toolbars (e.g., the now defunct *MementoFox*) and archival emulators (e.g., *NetCapsule*) are out of scope of this work. *InLine Plain HTML Banners* – This is the simplest and most common approach to add an archival banner. *OpenWayback* and *ArchiveIt* use this method. *OpenWayback* is commonly used and *ArchiveIt* is a good example of this approach. This approach, necessary markup and style for the banner is injected directly in the archived *HTML*. While simple, it poses some serious issues, such as conflicts with the style of the memento or hiding important elements of the page, such as the header of the site.

**RQ2: How do Web browser APIs compare in potential functionality to the capabilities of archival crawlers?**

*PyWB* is a popular archival replay system that uses the *Custom Elements* API to inject the banner into the parent page. The banner can be implemented by the crawler injecting the banner document into the parent page or by making the banner document the outer page and serving the memento inside an *iframe*. For example, *WebCite* uses the first approach while *PyWB* and *ArchiveNow* use the second. *PyWB*, a popular web archival replay system, uses the second approach by default, but allows using plain *inLine HTML Banners*. *Iframes* provide full document isolation, both style and origin. Therefore, *iframe* banners do not conflict with the position



## Enable archival navigation on replay to be more extensible

It makes the archival banner extensible and allows drag-and-drop repositioning, and it can hide itself when not needed. In the on-demand extended mode it provides a set of interactive visualizations and provenance information that are customizable by the archive.

# Preliminary Research

1. JCDL 2012 - WARCreate
  2. TPDL 2013 - Change in Archivability
  3. DLib 2013 - Method for Identifying
  4. JCDL 2014 - Mink
  5. JCDL 2014 - Archival Acid Test
  6. JCDL 2014 - Not All Mementos
  7. IJDL 2015 - Impact of JavaScript
  8. IJDL 2015 - Not All Mementos
  9. JCDL 2015 - Mobile Mink
  10. JCDL 2016 - InterPlanetary Wayback (ipwb)
  11. TPDL 2016 - ipwb extended
  12. JCDL 2017 - WAIL Electron
  13. JCDL 2017 - ServiceWorker Replay
  14. JCDL 2017 - Impact of Canonicalization
  15. JCDL 2018 - ArchiveNow
  16. JCDL 2018 - Replay Banners
  17. **JCDL 2018 - A Framework...**



# **BEST PAPER AWARD FINALIST**

## **at JCDL 2018**

# A Framework for Aggregating Private and Public Web Archives

Mat Kelly  
Old Dominion University  
Norfolk, Virginia, USA  
[mkelly@cs.odu.edu](mailto:mkelly@cs.odu.edu)

Michael L. Nelson  
Old Dominion University  
Norfolk, Virginia, USA  
[mln@cs.odu.edu](mailto:mln@cs.odu.edu)

Michele C. Weigle  
Old Dominion University  
Norfolk, Virginia, USA  
[mweigle@cs.odu.edu](mailto: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 potentially sensitive information contained in private captures. We amend

elements, syntax and semantics to allow Timemap enrichment to account for additional attributes to be expressed inclusive of the preliminary research agenda. The Web archivist can also provide a method to involve the user further in the negotiation of archival captures in dimensions beyond time. We introduce a mechanism for aggregating privacy precedence and short-circuiting the process 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

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 WARCCreate, 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 public. The responsibility for managing privacy in the Web archive lies with the archivist. Strategies for mitigating privacy issues, strategically regulating access to these personal and private records, and the potential for legal challenges, will be discussed. Adding users to Web archives and privacy considerations to the aggregate view of the Web 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

**Preliminary research aggregating private and public Web archives**

**RQ3:** What issues exist for capturing and replaying content behind authentication?  
CCS CONCEPTS  
TimeMap/Link Enrichment: Provide additional, more descriptive attributes to URI-Ms for more efficient querying and

**RQ4:** How can content that was captured behind interaction (Section 4).  
Multi-dimensional user-driven content negotiation of

**RQ4:** How can content that was captured behind authentication signal to Web archive replay systems that it requires special handling?

DOI: <https://doi.org/10.1145/2374071.2374072> © 2014 ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/2374071.2374072>

**RQ5:** How can Memento aggregators indicate that private Web archive content requires special handling to be replayed, despite being aggregated with publicly available Web archive content?

**RQ6:** What kinds of access control do users who create have even been mementoed at the same time? Others may be hesitant to share their mementos of facebook.com for other personal or private

**Corporate Web archives need to regulate access to their archives?**

# Outline

- Introduction/Motivation
- Background
- Preliminary Research
- **Proposed Framework**
- Evaluation Plan

# Proposed Framework

(for aggregating private and public Web archives)

# Proposed Framework

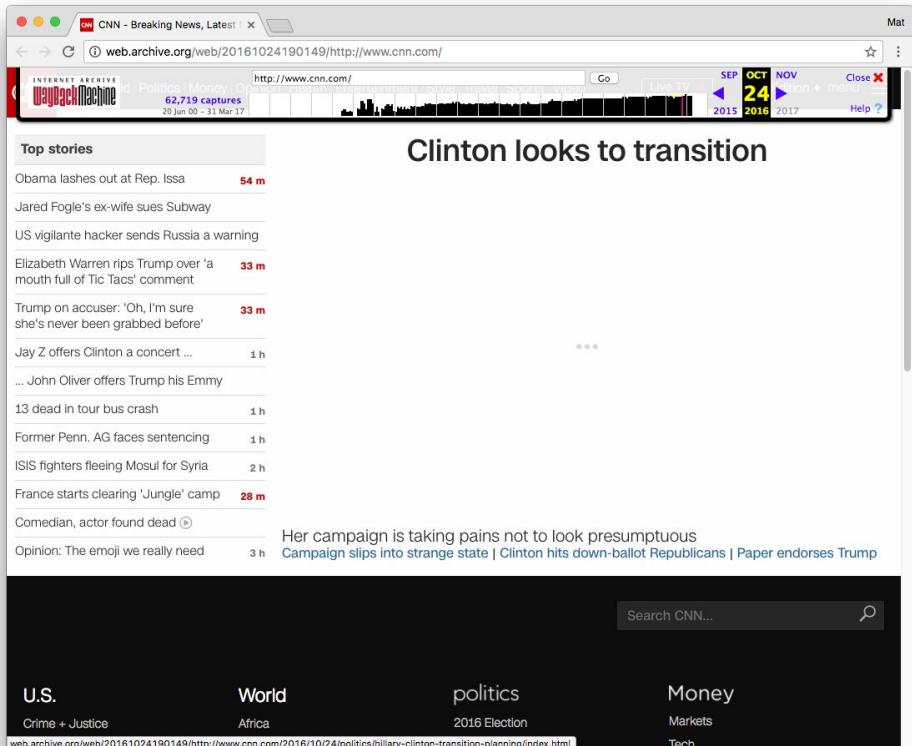
- Archival negotiation beyond time
- Query precedence & short-circuiting
- Mementies

# PROPOSED FRAMEWORK

Archival Negotiation Beyond Time

# More Expressive TimeMaps

- Memento Quality (e.g., Damage)<sup>1</sup>
  - How Many Captures?<sup>2</sup>
  - How Many Are Identical?<sup>2,3</sup>
  - Other Attributes of Mementos...



<sup>1</sup> Brunelle et al., JCDL 2014, IJDL 2015

<sup>2</sup> Kelly et al., JCDL 2017

<sup>3</sup> AlSum and Nelson, ECIR 2014

# Additional TimeMap Attributes

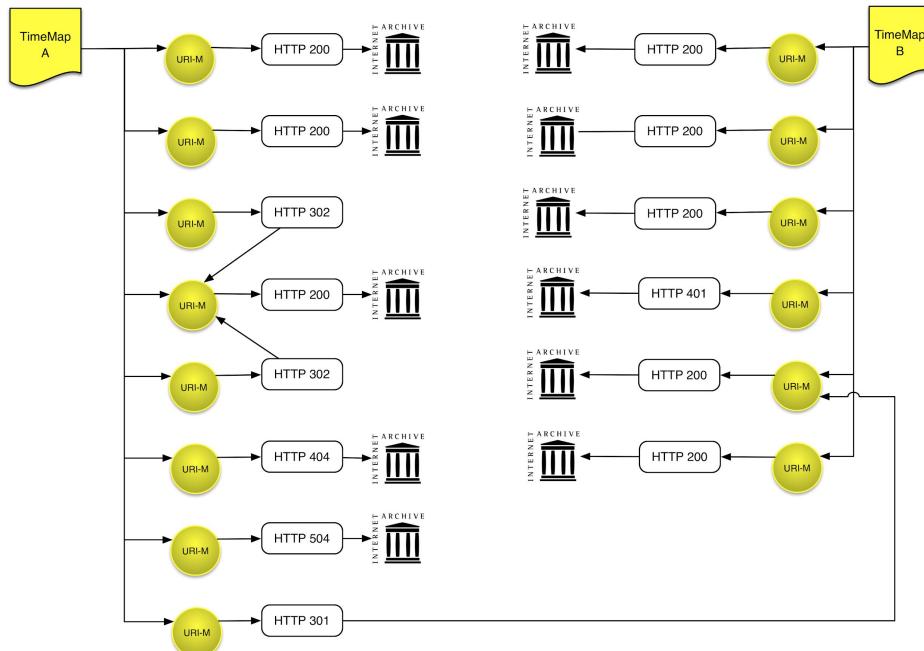
Content-based Attributes

Derived Attributes

Access Attributes

# TimeMap Enrichment: Content-Based Attributes

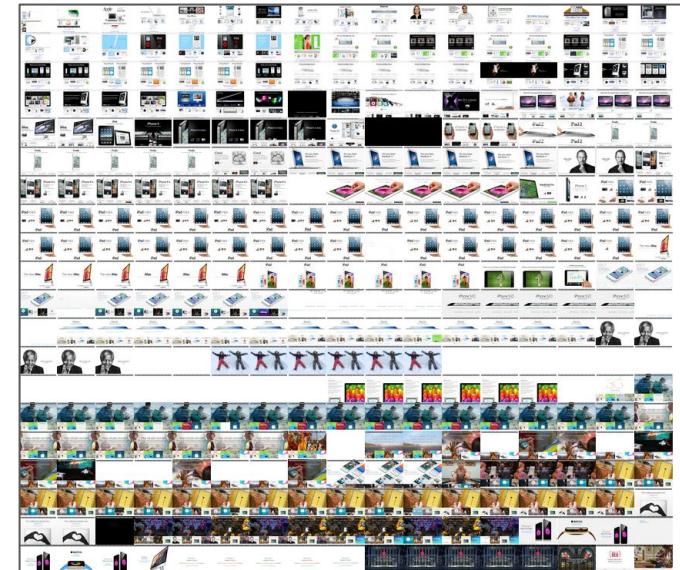
- Status Code<sup>1</sup>
- Content-Digest
  - In WARC & CDX
  - Not all archives expose CDX
- Would allow more info about mementos without requiring comprehensive dereferencing



<sup>1</sup> Kelly et al., “Impact of URI Canonicalization on Memento Count”, JCDL 2017, arXiv 1703.03302

# TimeMap Enrichment: Derived Attributes

- Thumbnails (e.g, via SimHash)<sup>1</sup>
  - Calculation based on root memento's HTML
- Memento Damage (JCDL 2014, IJDL)<sup>2</sup>
  - Requires dereferencing embedded resources



apple.com, many duplicate mementos!

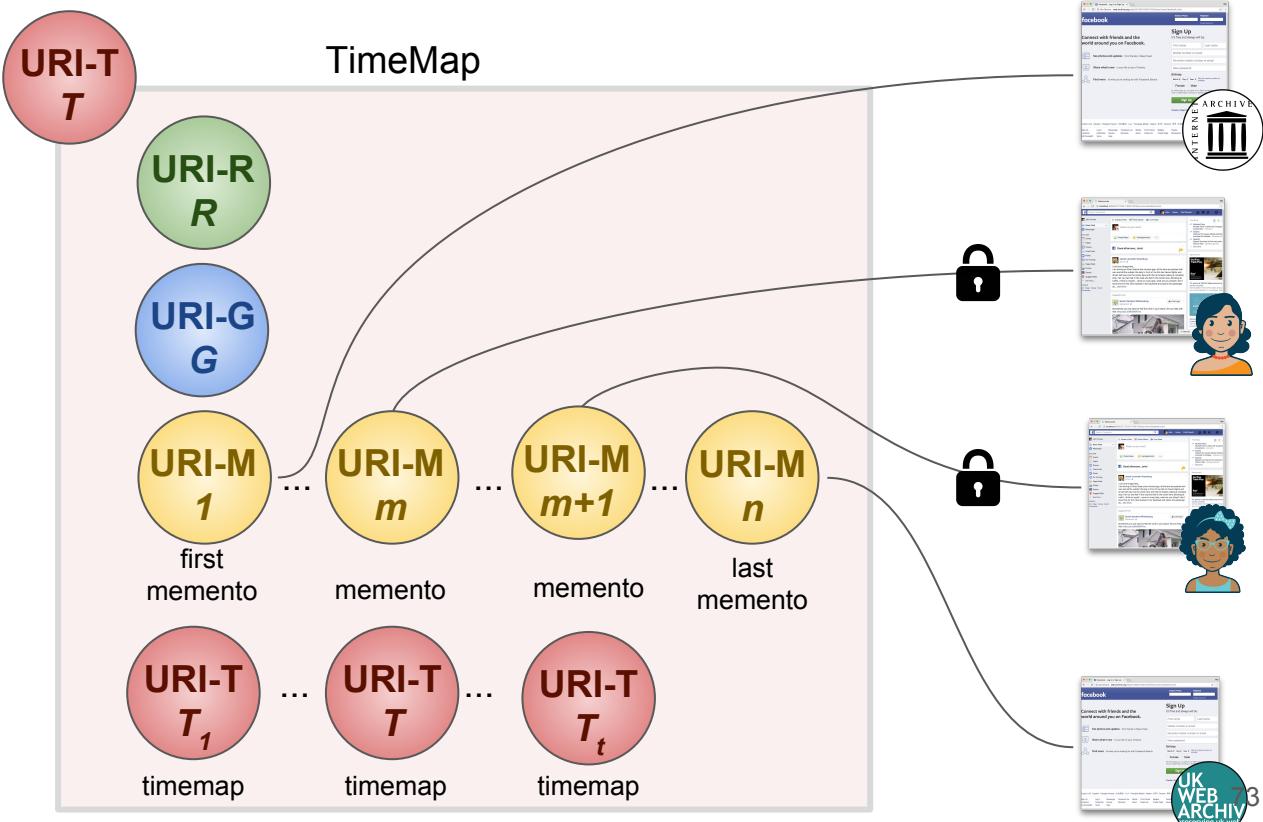
<sup>1</sup> AlSum and Nelson, Thumbnail Summarization Techniques for Web Archives, ECIR 2014, pp. 299-310.

<sup>2</sup> Brunelle *et al.*, "The Impact of JavaScript on Archivability," IJDL, 17(2), pp. 95-117. January 2016.

# TimeMap Enrichment: Access Attributes

How to distinguish  
**Private captures**  
from  
**Public captures**  
in a TimeMap?

RQ5: How can Memento aggregators indicate that private Web archive content requires special handling to be replayed, despite being aggregated with publicly available Web archive content?



# TimeMap Enrichment - in a CDXJ TimeMap

*Line breaks added for clarity, CDXJ records occupy a single line*

```
19981212013921 {  
    "uri": "http://localhost:8080/20101116060516/http://facebook.com/",  
    "rel": "memento",  
    "datetime": "Tue, 16 Nov 2010 06:05:16 GMT",  
    "status_code": 200,  
    "digest": "sha1:1K26DRRQJ4WATC6LBVF3B3Z4P2CP5ZZ7",  
    "damage": 0.24,  
    "simhash": "6551110622422153488",  
    "content-language": "en-US",  
    "access": {  
        "type": "Blake2b",  
        "token": "c6ed419e74907d220c69858614d86...ef0a3a88a41"  
    }  
}
```

**Content-based attributes**

**Derived Attributes**

**Access Attributes**

TimeMap

+

Enrichment with Additional Attributes

---

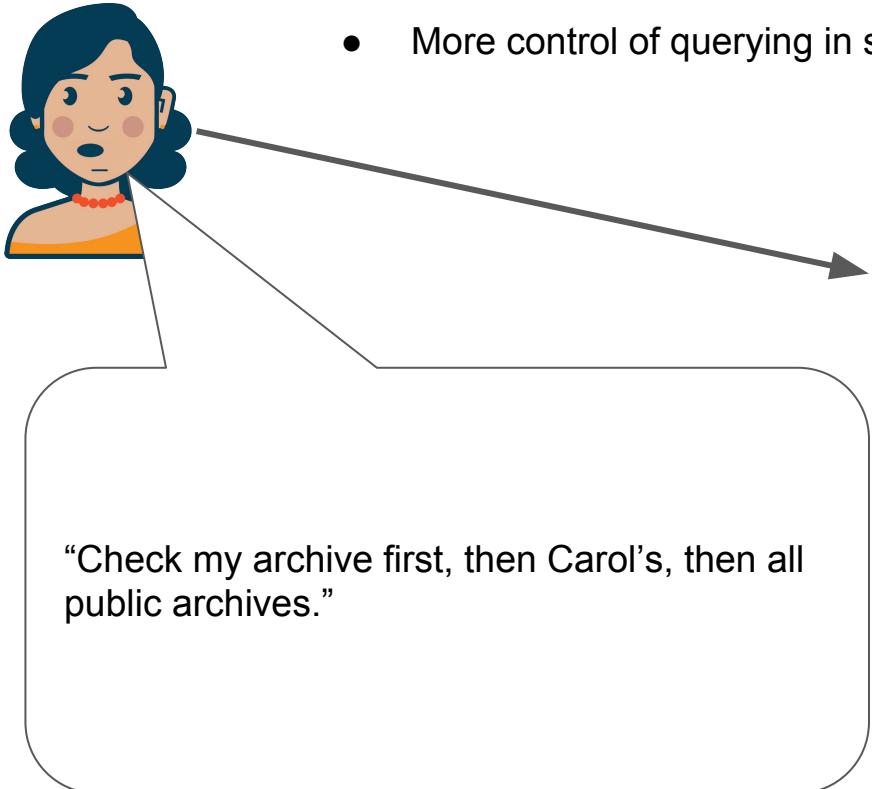
“StarMap”

# PROPOSED FRAMEWORK

Query Precedence  
- and -  
Short Circuiting

# Query Precedence

- More control of querying in series and parallel

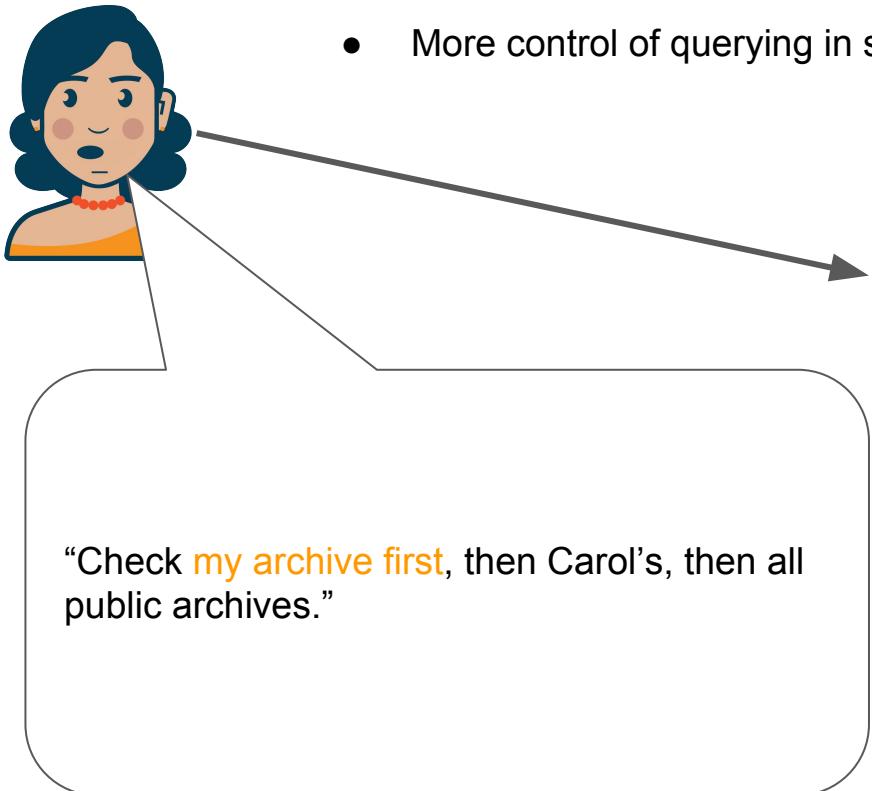


"Check my archive first, then Carol's, then all public archives."



# Query Precedence

- More control of querying in series and parallel



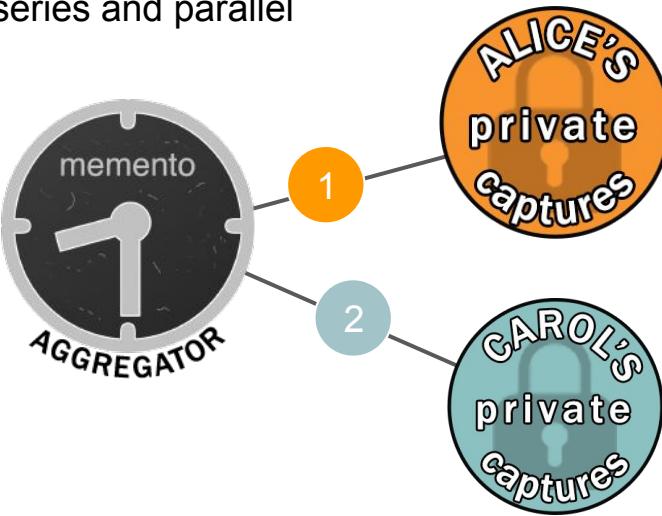
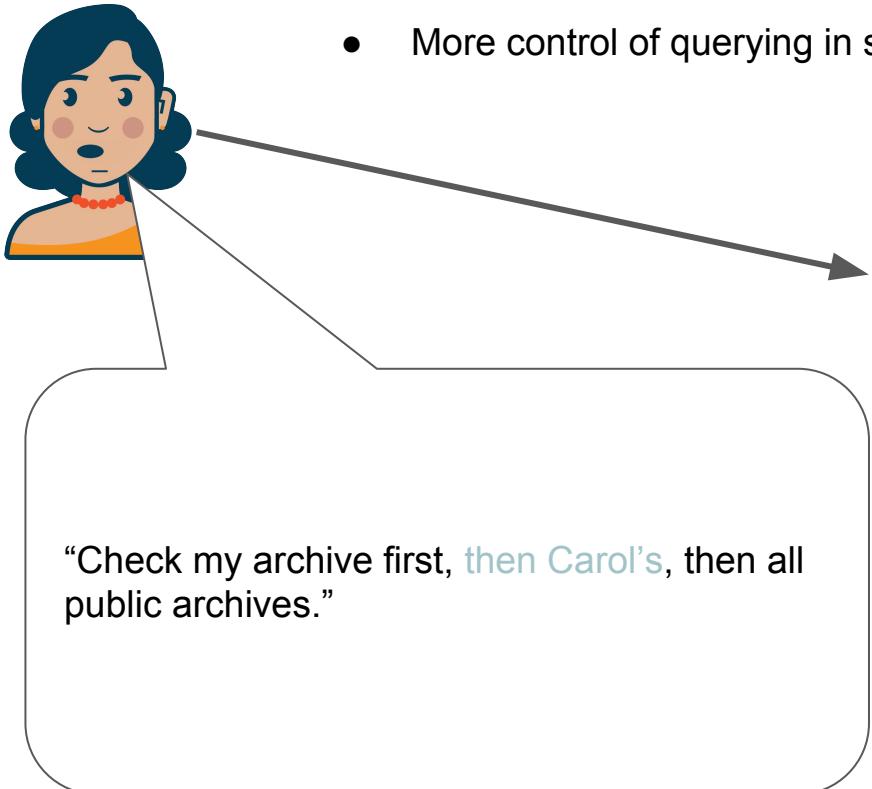
1



"Check **my archive first**, then Carol's, then all public archives."

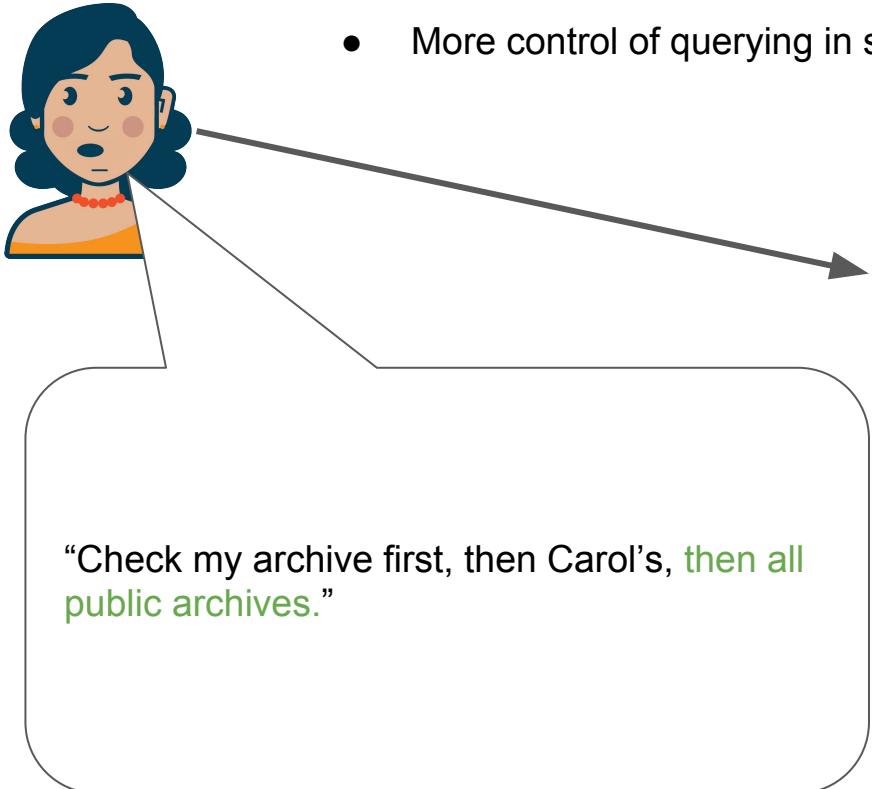
# Query Precedence

- More control of querying in series and parallel



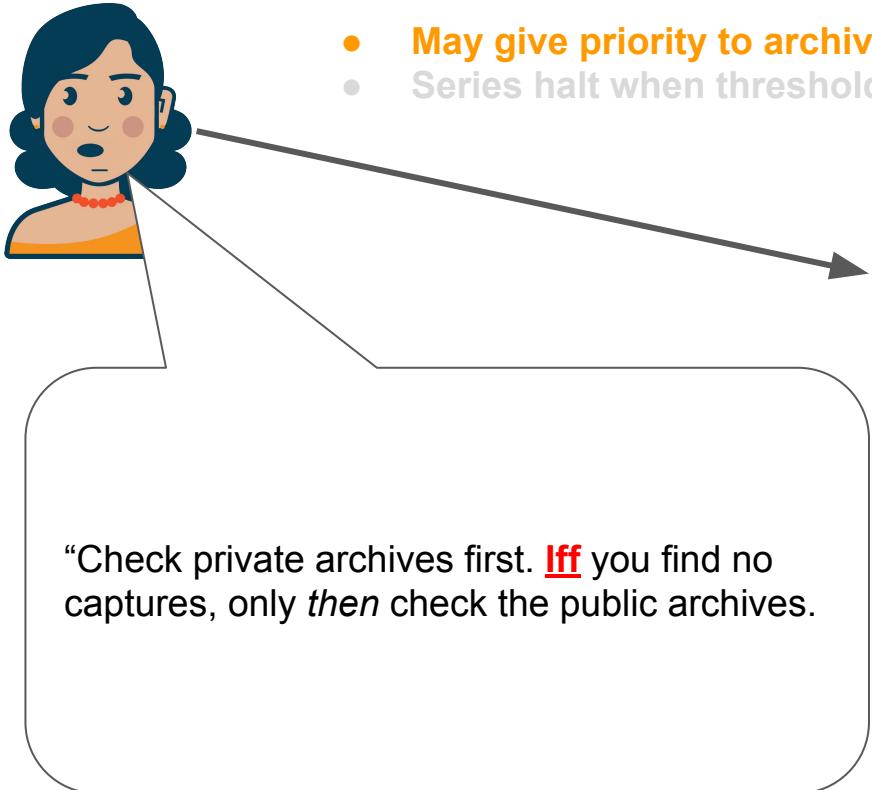
# Query Precedence

- More control of querying in series and parallel



# Query Short-Circuiting

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

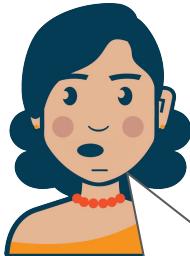


"Check private archives first. **Iff** you find no captures, only *then* check the public archives.

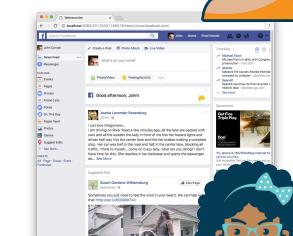
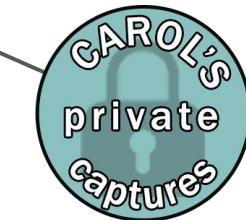


# Query Short-Circuiting

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



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

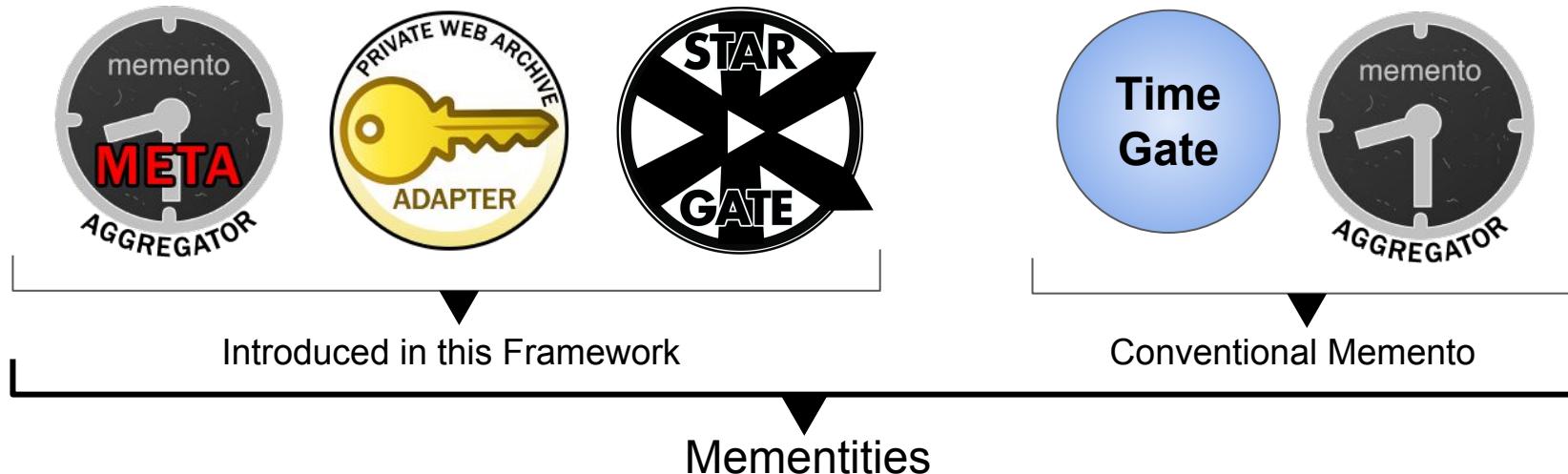


# PROPOSED FRAMEWORK

Mementies

# Mementities

- Memento + Entity (*entity* term already overused)



# PROPOSED FRAMEWORK

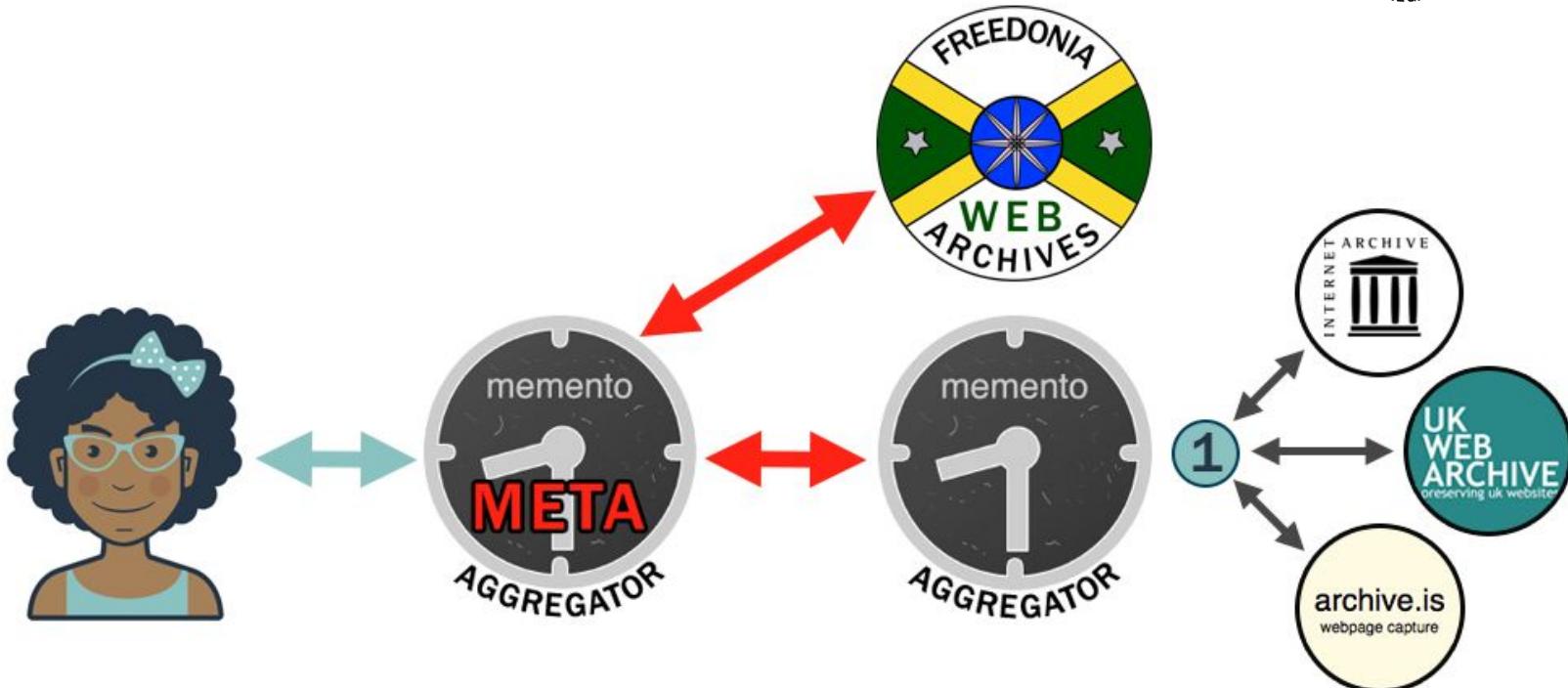
Mementities



A Framework for Aggregating Public and Private Web Archives  
February 14, 2019  
*Mat Kelly*



# Memento Meta-Aggregator (MMA)



functional  
≡



# MMA: Archive Selection



# MMA: User-Driven Archival Specification



# MMA Aggregation sources

$MMA_{\alpha}$ :

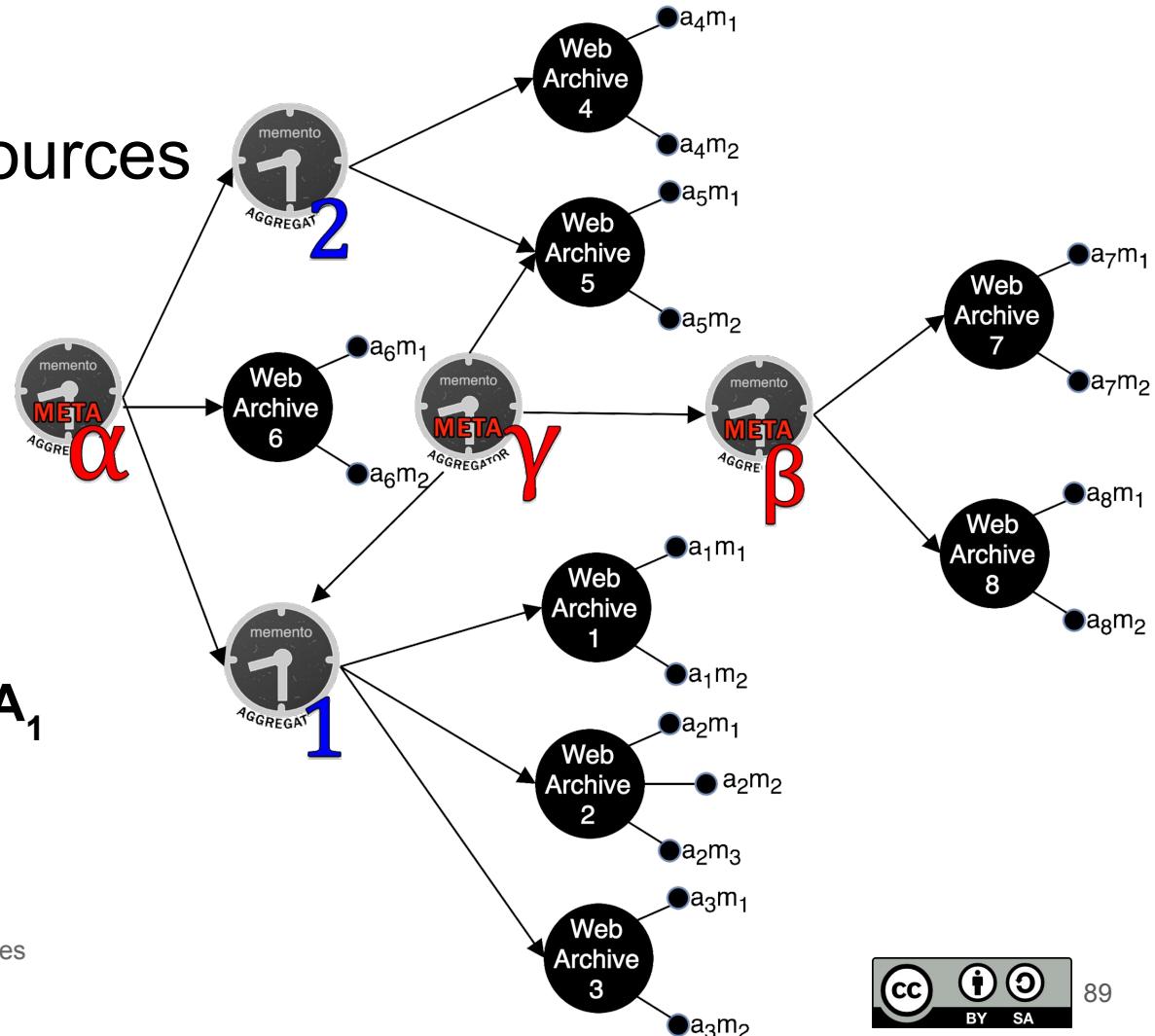
from  $MA_2$ ,  $MA_1$  and  $WA_6$

$MMA_{\beta}$ :

from  $WA_7$  and  $WA_8$

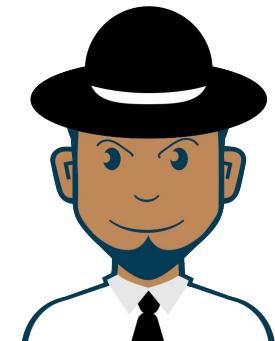
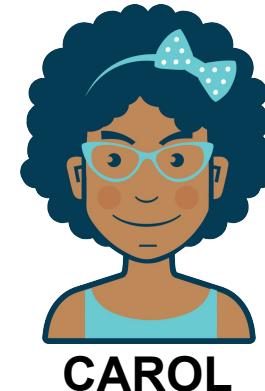
$MMA_{\gamma}$ :

from  $MMA_{\beta}$ ,  $MA_5$ , and  $WA_1$



# MMA Dynamics By-Example

- Personal Archive Aggregation
- MMA Chaining
- Client-Side Aggregation Preference



# MMA Dynamics - Personal Archive Aggregation



bbc

homepage

Public videos



FB

bank

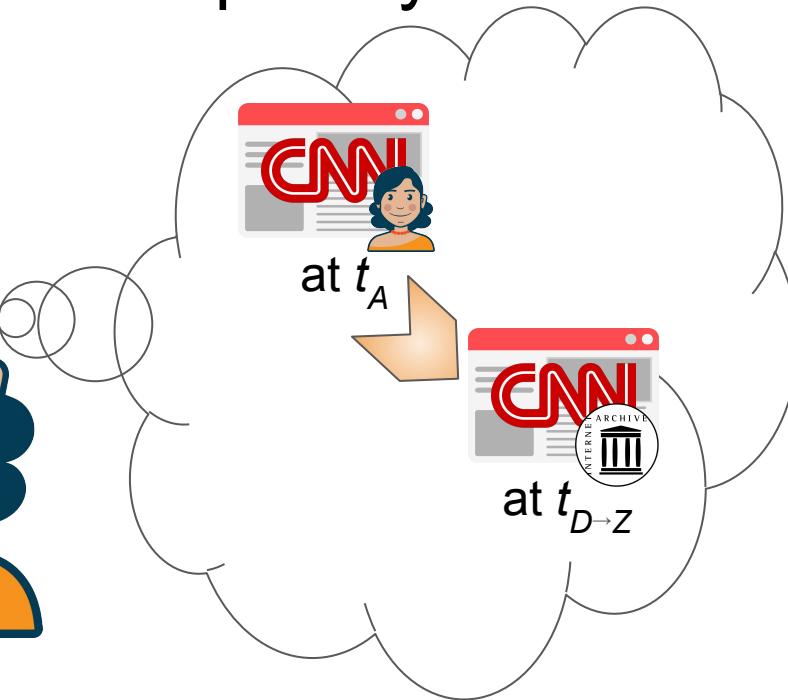
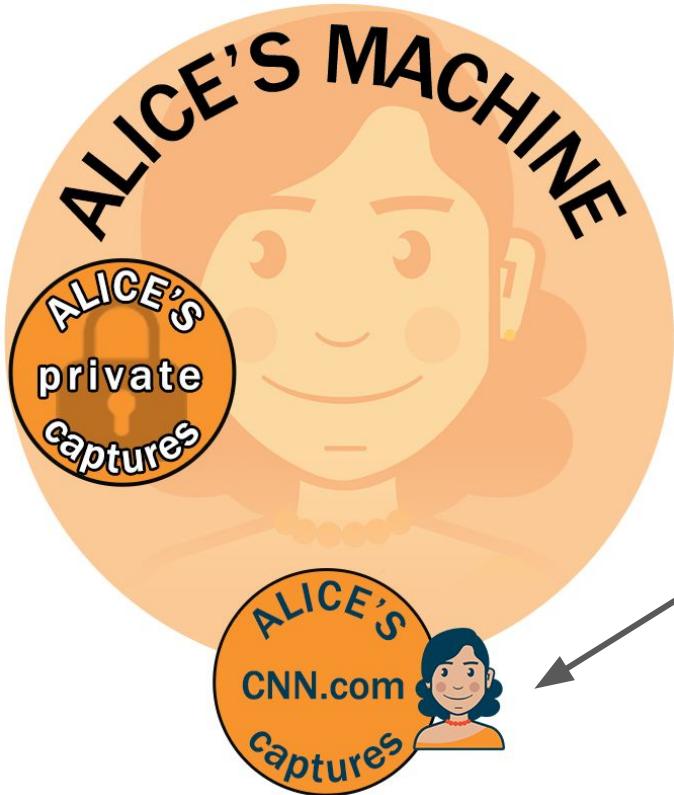
flickr

# Alice Saves the Web



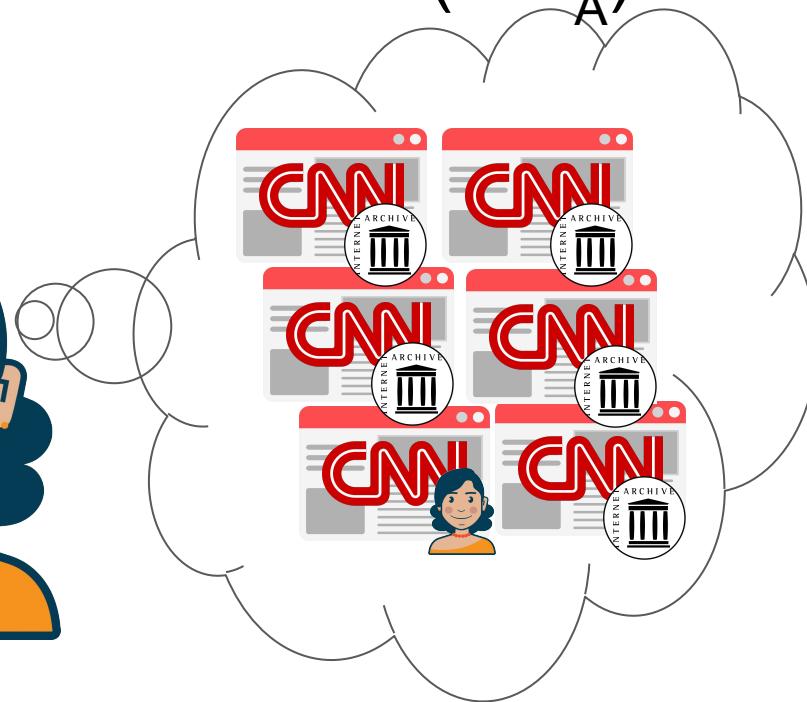
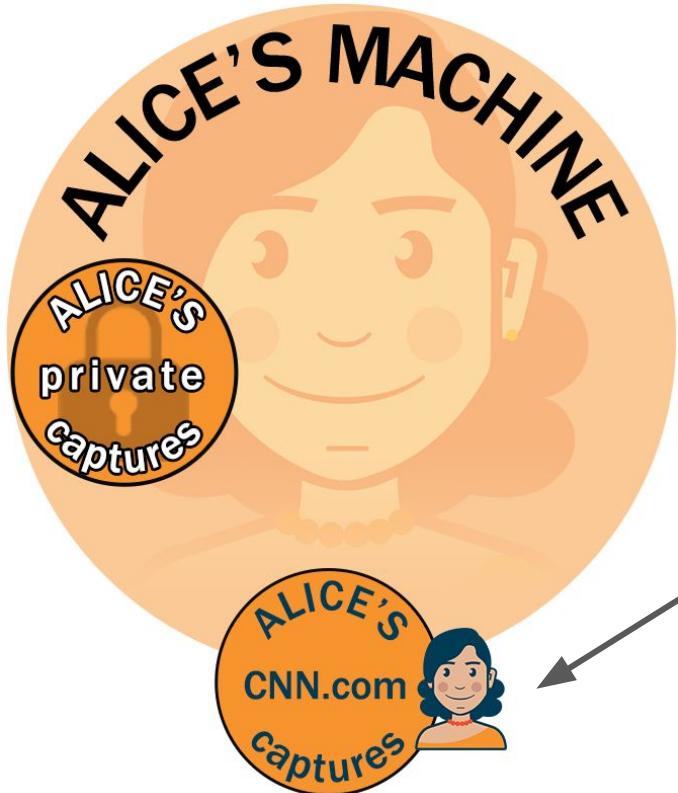
*Personal Archive Aggregation*

# Alice Wants to See Her Captures Temporally Inline



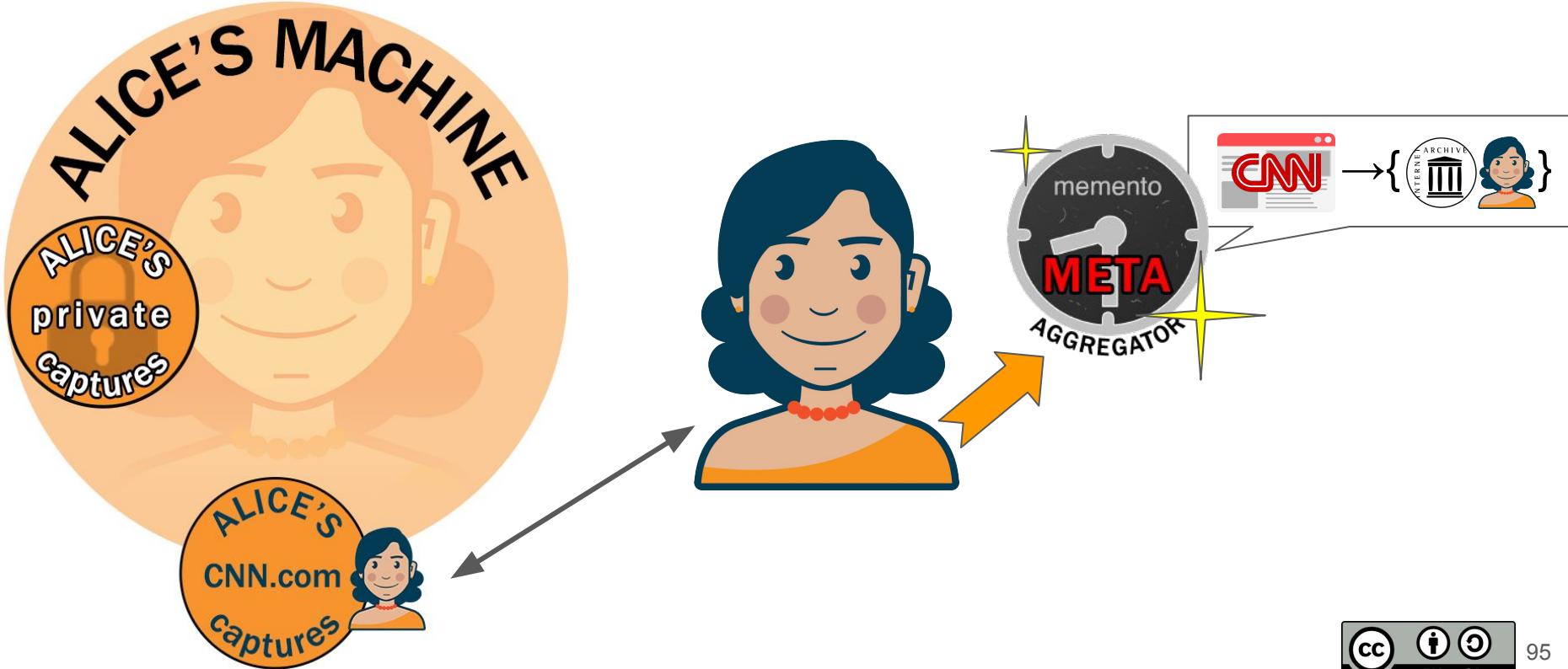
Personal Archive Aggregation

# Mementity Dynamics - Alice & Her Archives ( $WA_A$ )

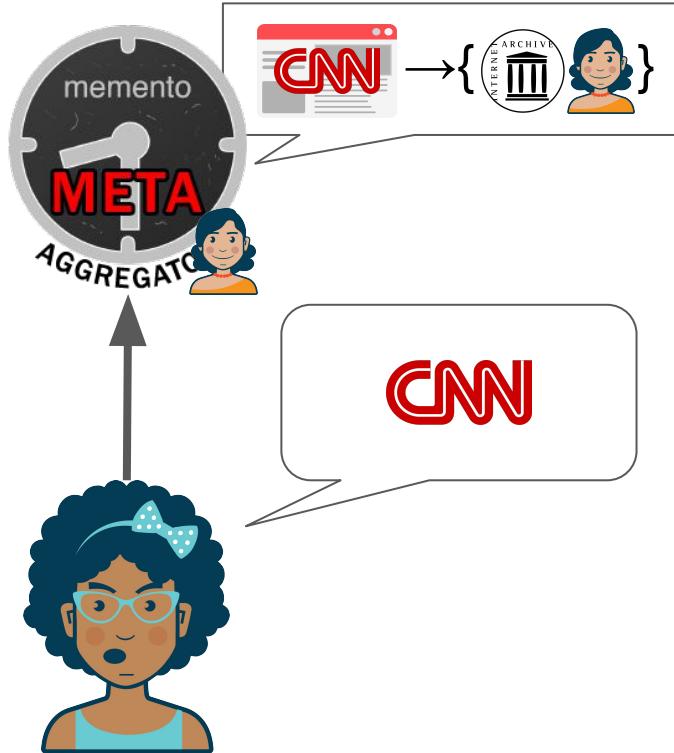


*Personal Archive Aggregation*

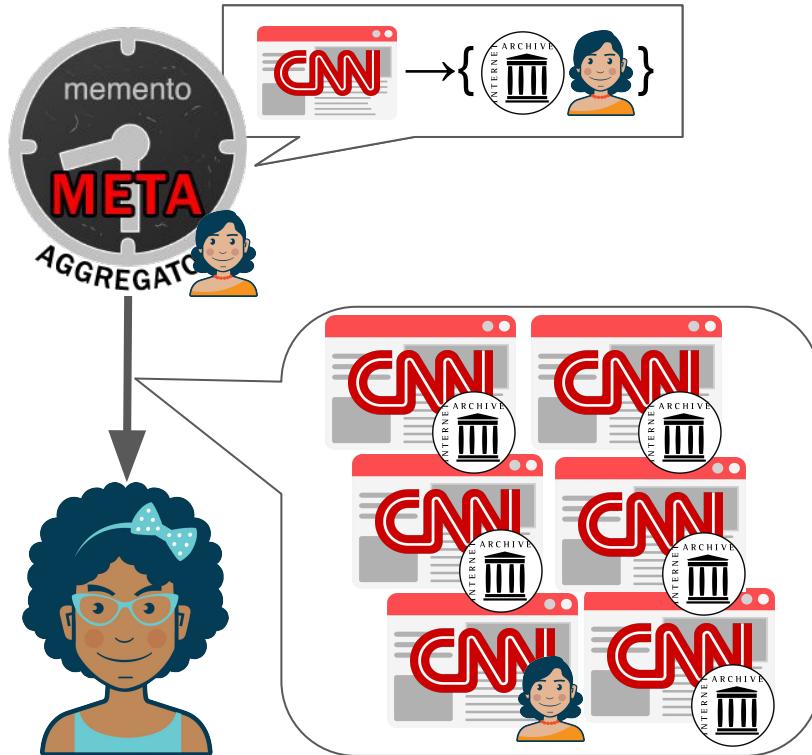
# Alice Deploys MMA<sub>A</sub>



# Carol Asks MMA<sub>A</sub> for CNN



# $MMA_A$ returns CNN Memento $\{M_A, M_{IA}\}$

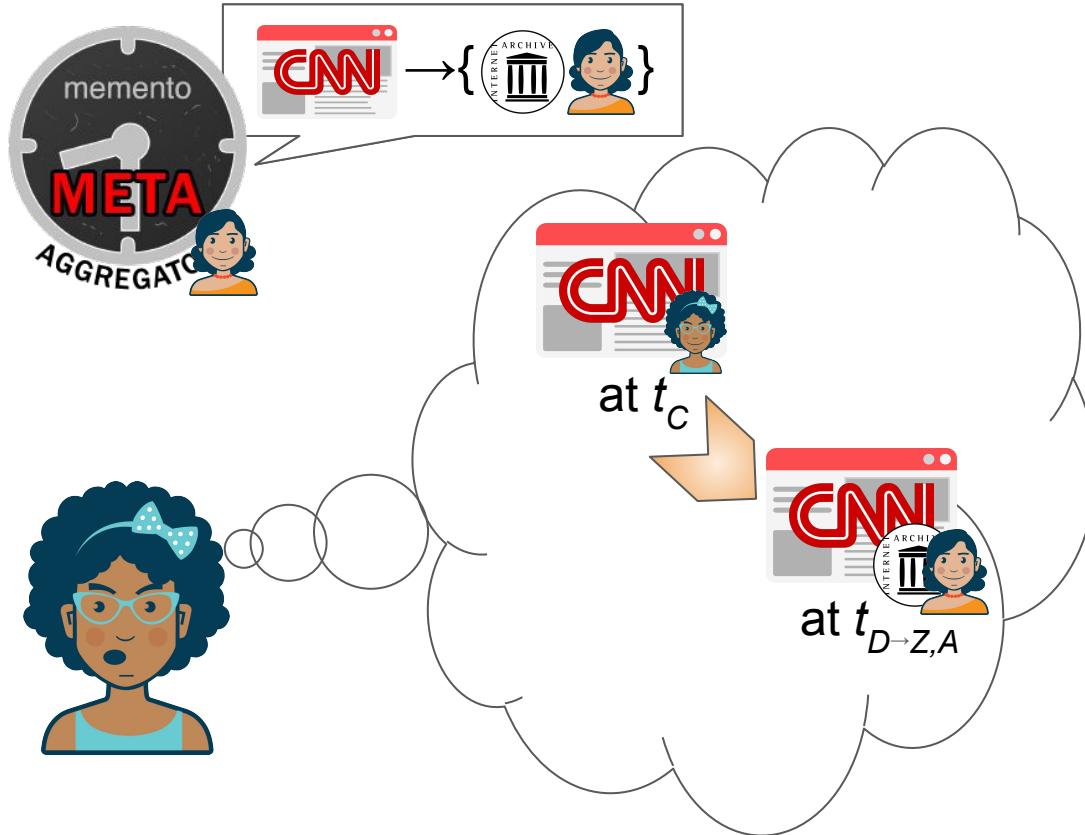


*MMA Chaining*



# Carol Wants to Aggregate Her Own Captures

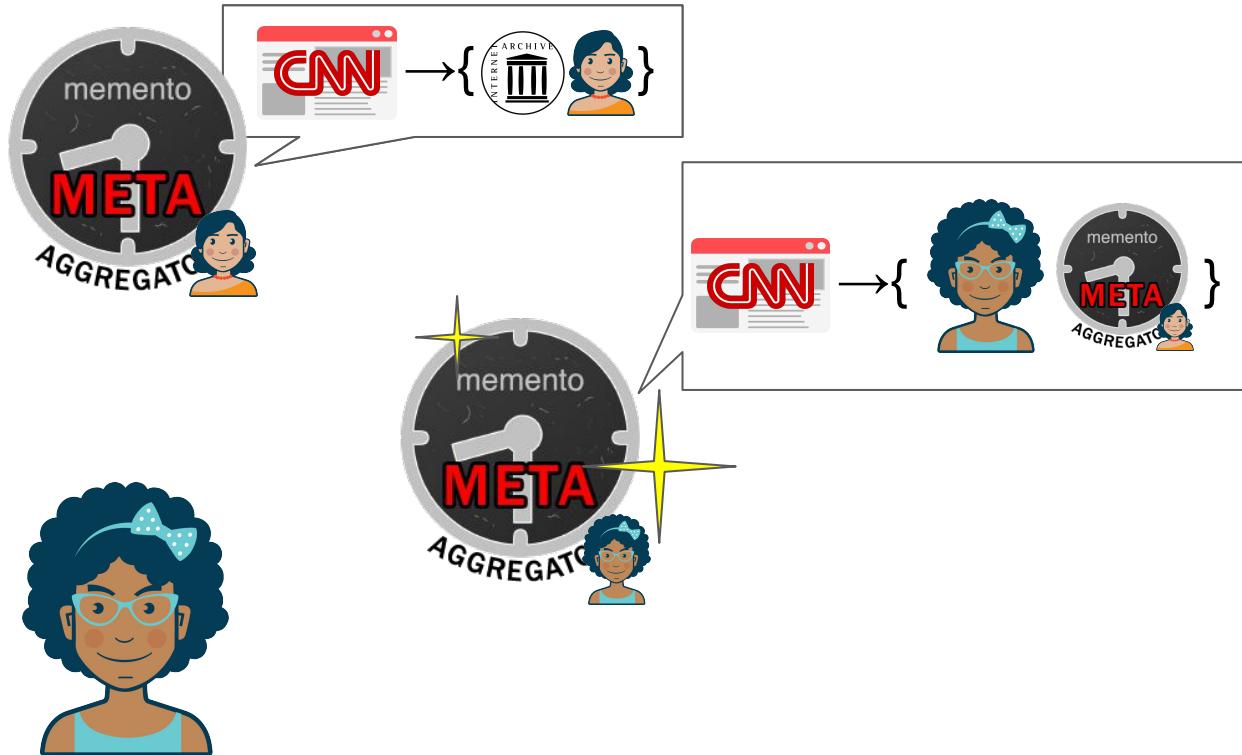
**CNN**(M(WA<sub>C</sub>))



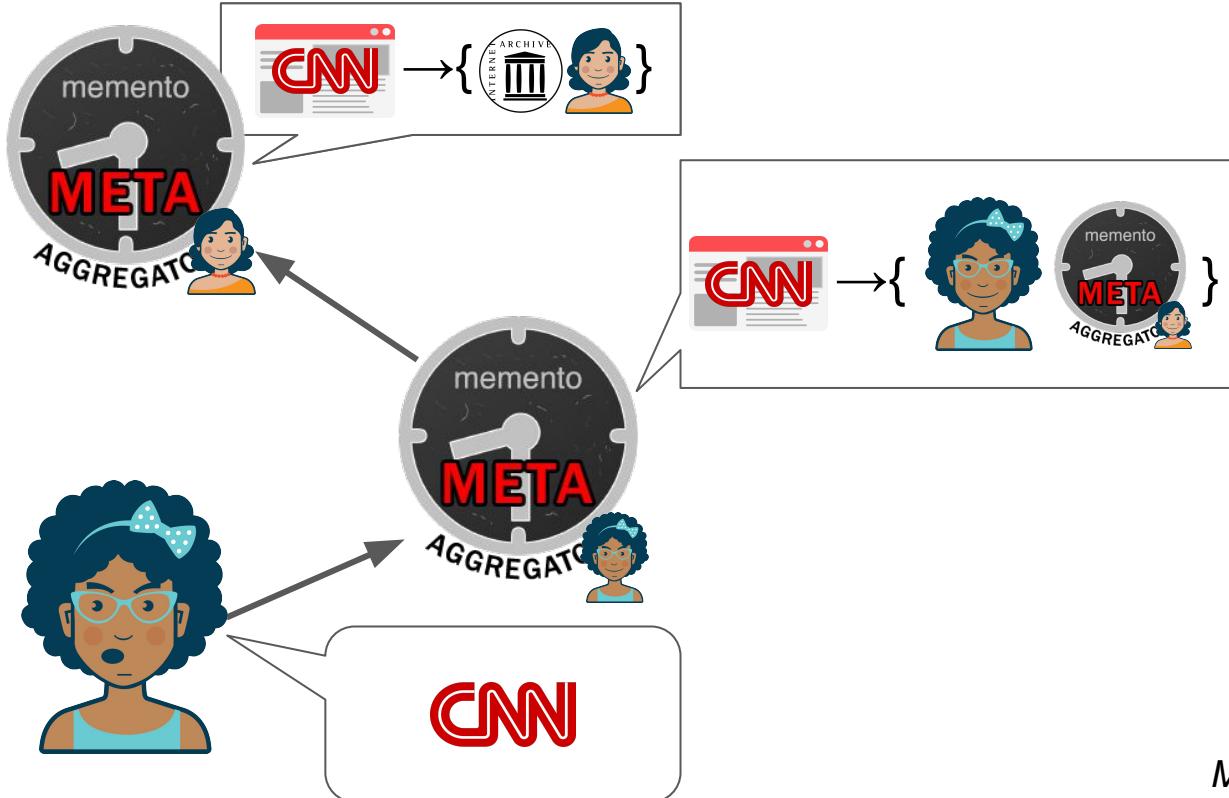
MMA Chaining



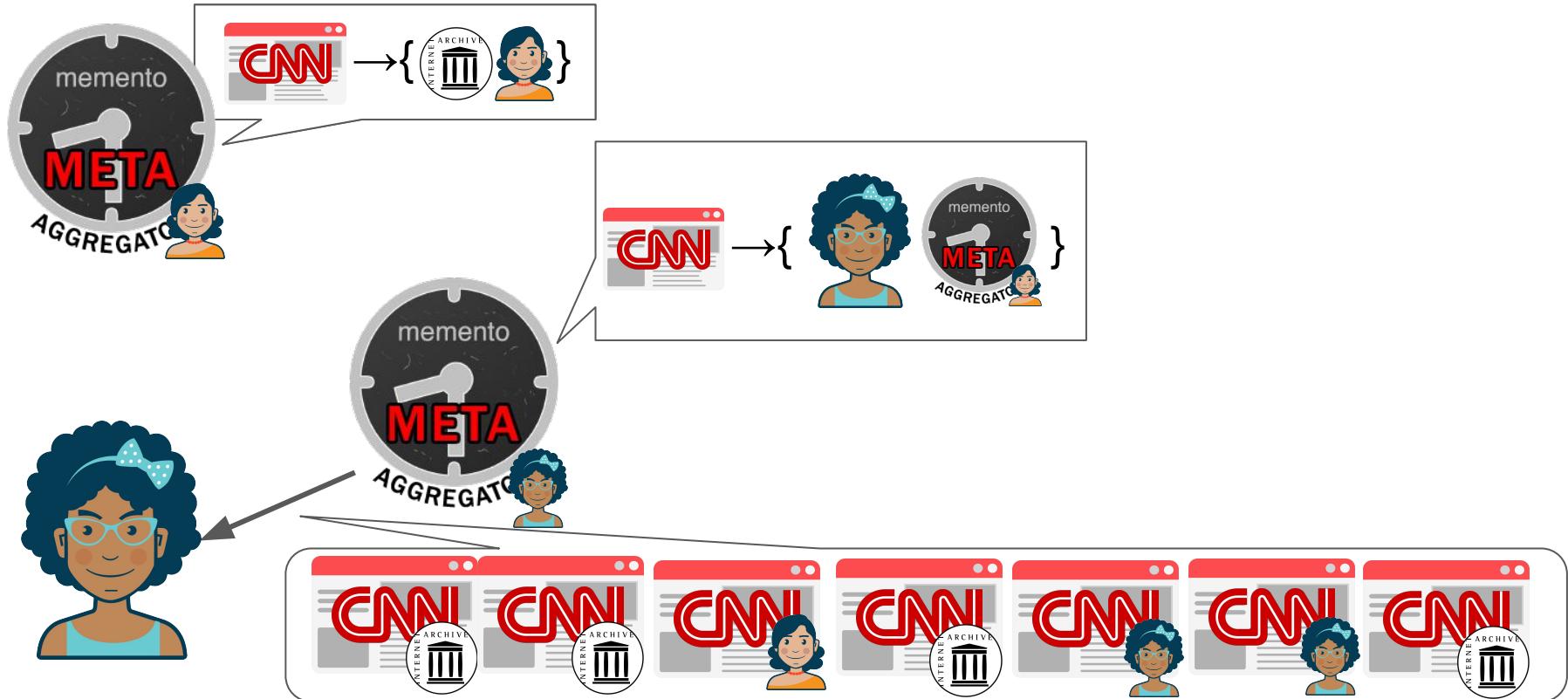
# Carol Creates MMA<sub>C</sub> to Access WA<sub>C</sub> and MMA<sub>A</sub>



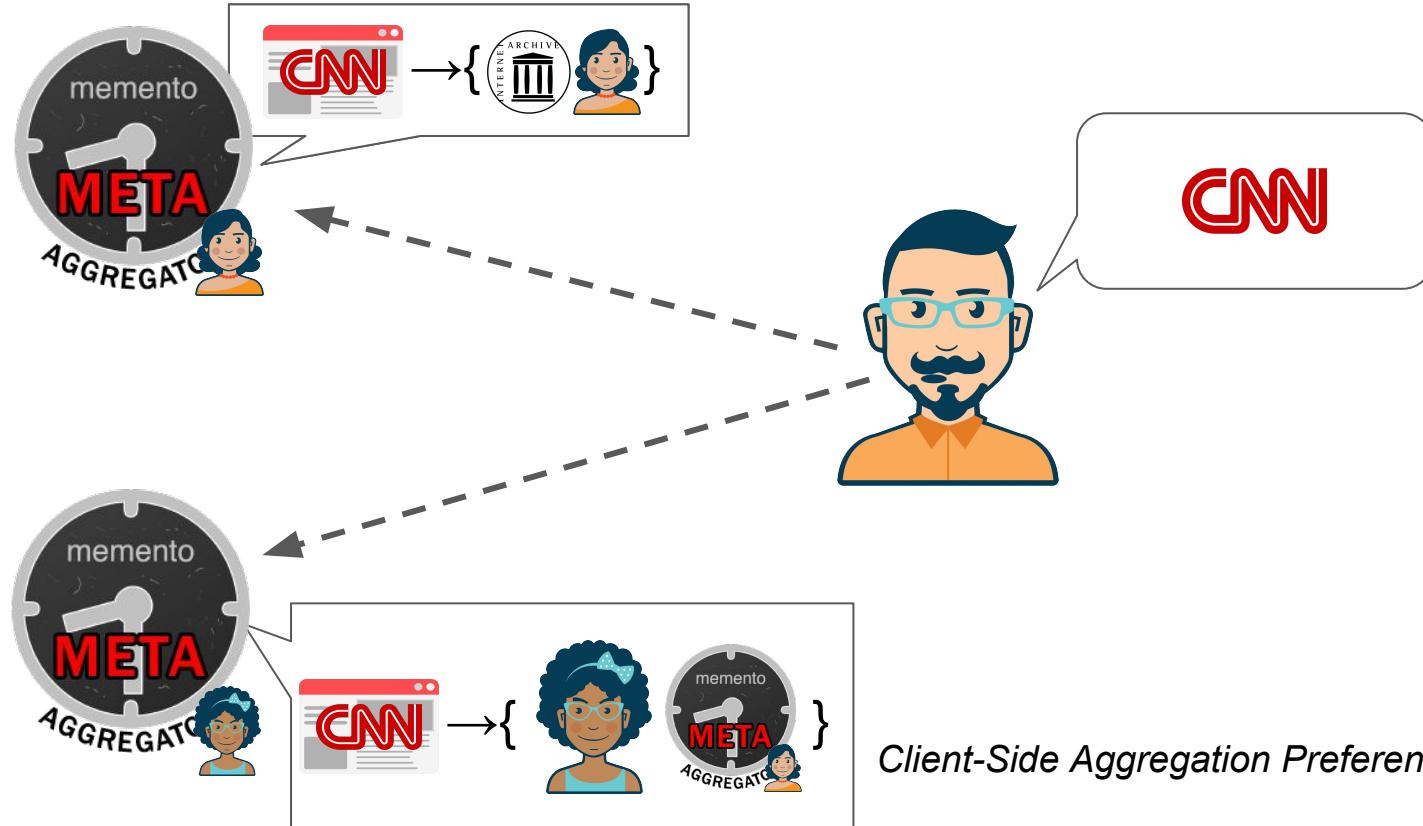
# Carol Asks MMA<sub>C</sub> For CNN



# $MMA_A$ returns CNN Memento $\{M_A, M_{IA}, M_C\}$

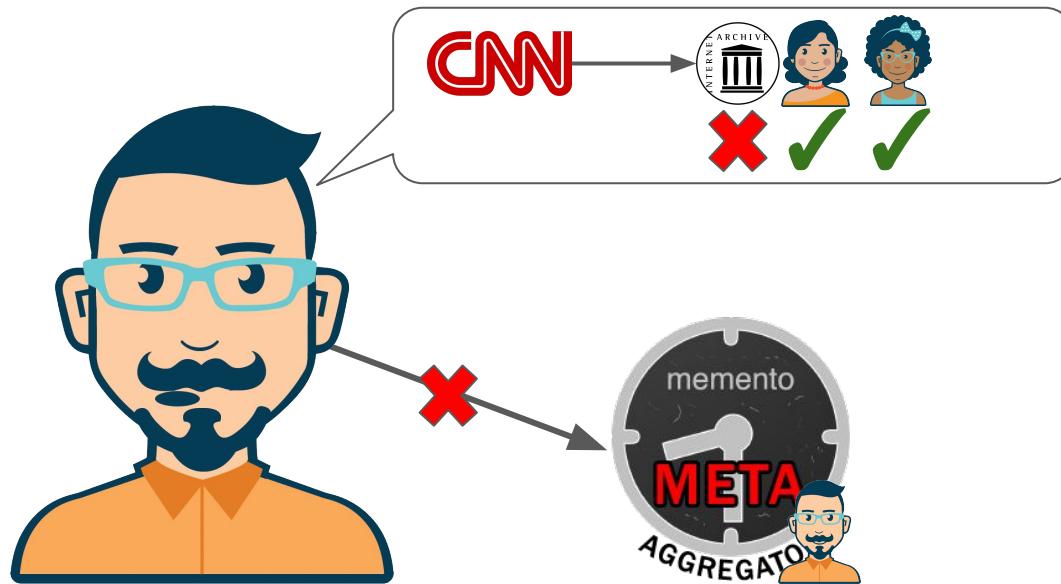


# Bob May Request M(CNN) From MMA<sub>A</sub> or MMA<sub>C</sub>



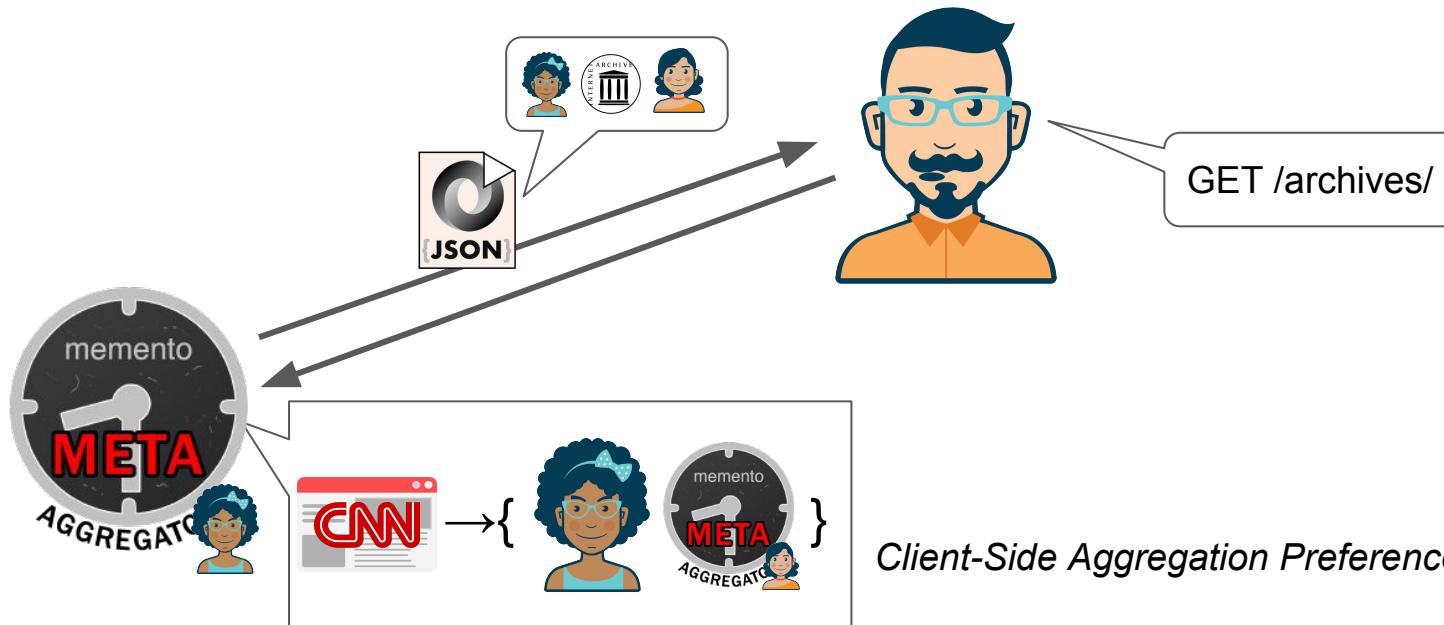
# Bob Prefers to Exclude IA Captures

...and does not want to setup his own MMA

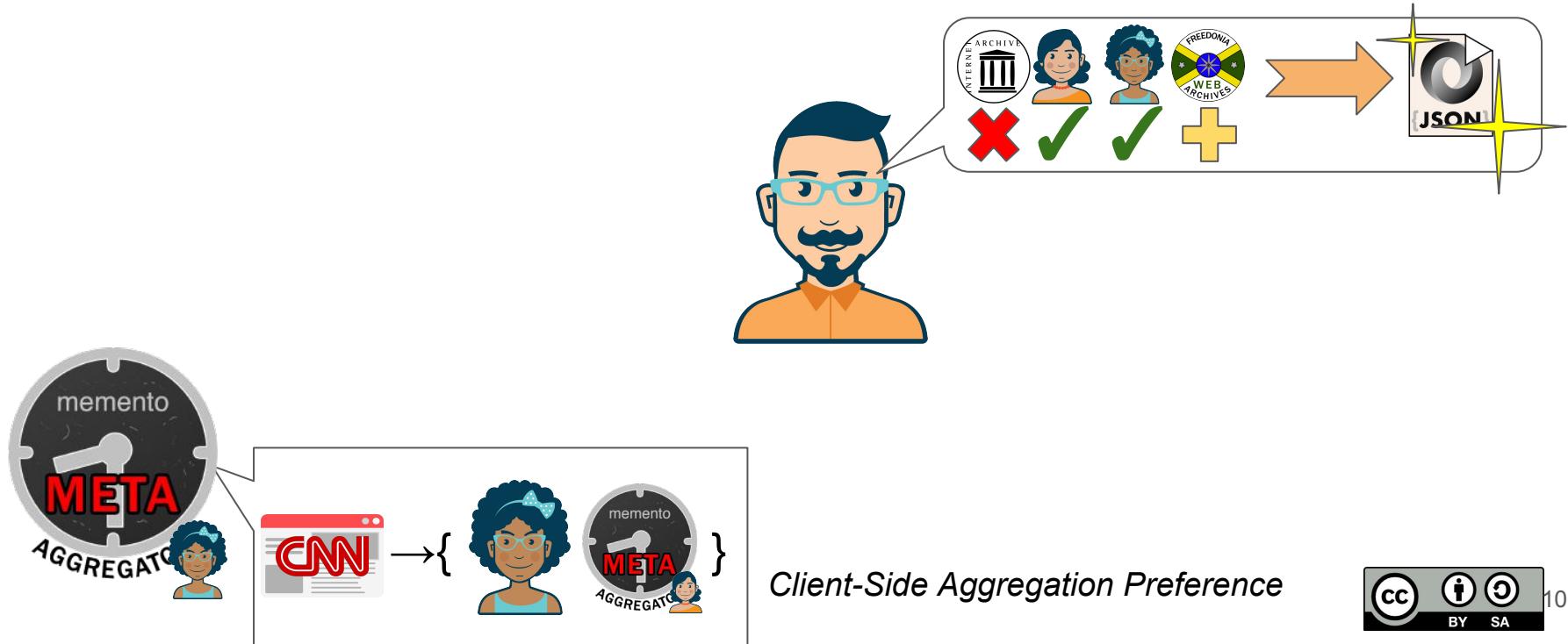


*Client-Side Aggregation Preference*

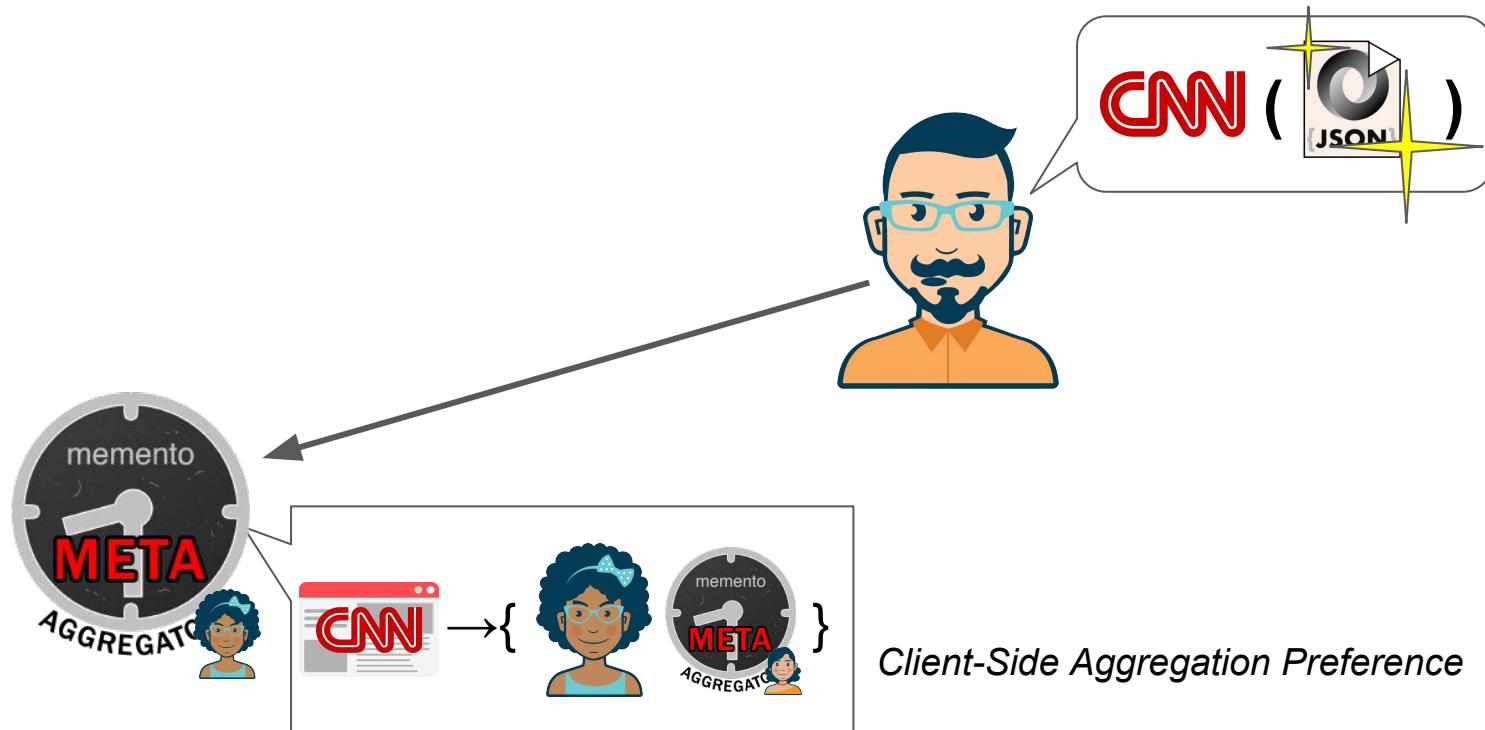
# Bob Requests Supported Archives



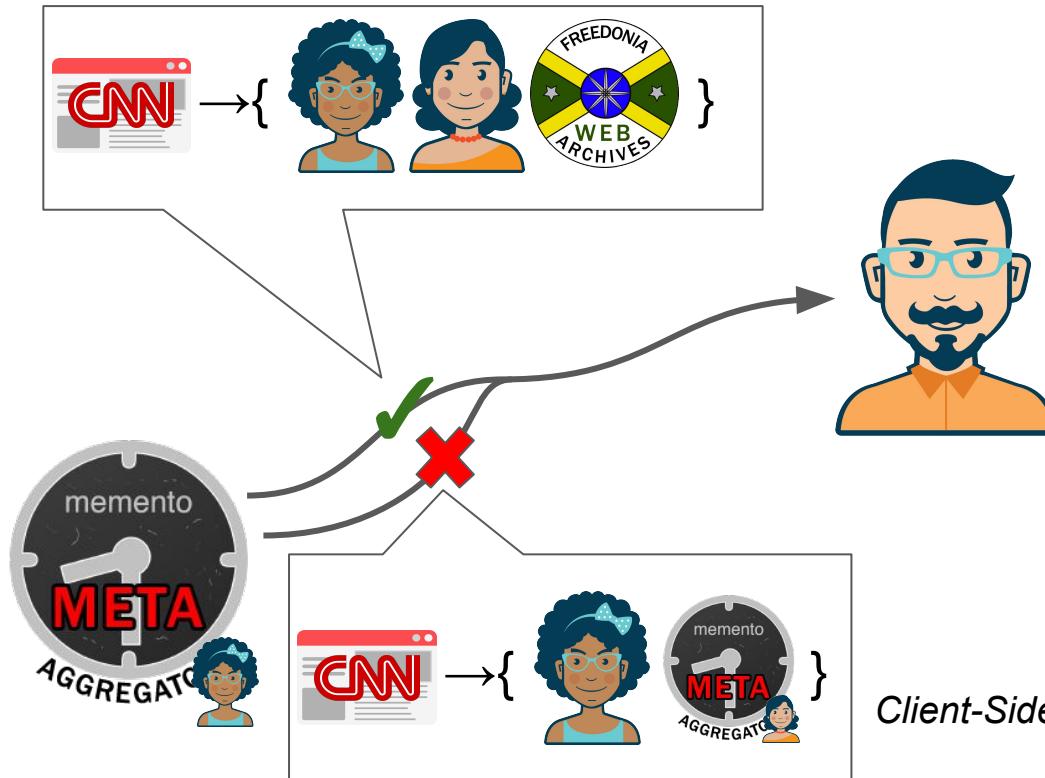
# Bob Customizes the Set in the JSON



# Bob Requests CNN for His Custom Set

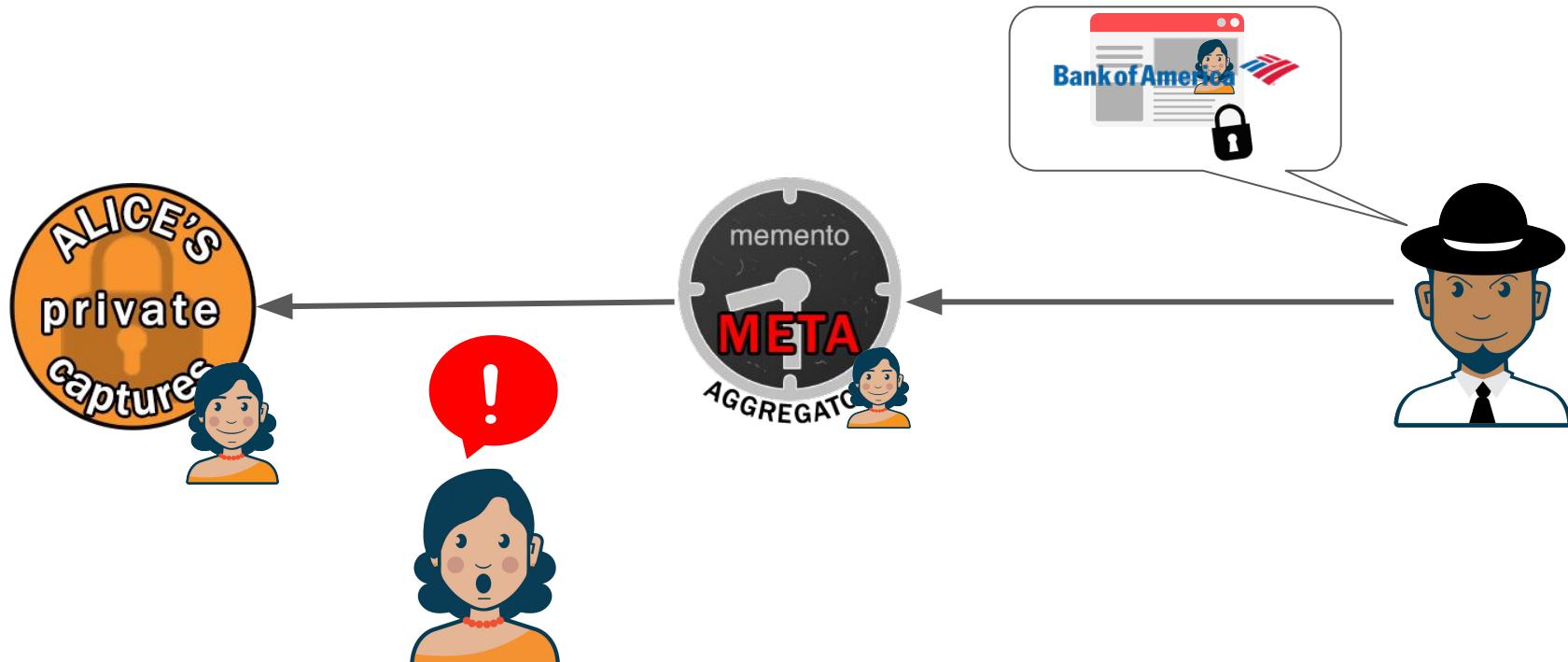


# MMA Complies or Ignores Preference



*Client-Side Aggregation Preference*

# Hooray, Aggregation!



A Framework for Aggregating Public and Private Web Archives  
February 14, 2019  
Mat Kelly



# PROPOSED FRAMEWORK

Mementities

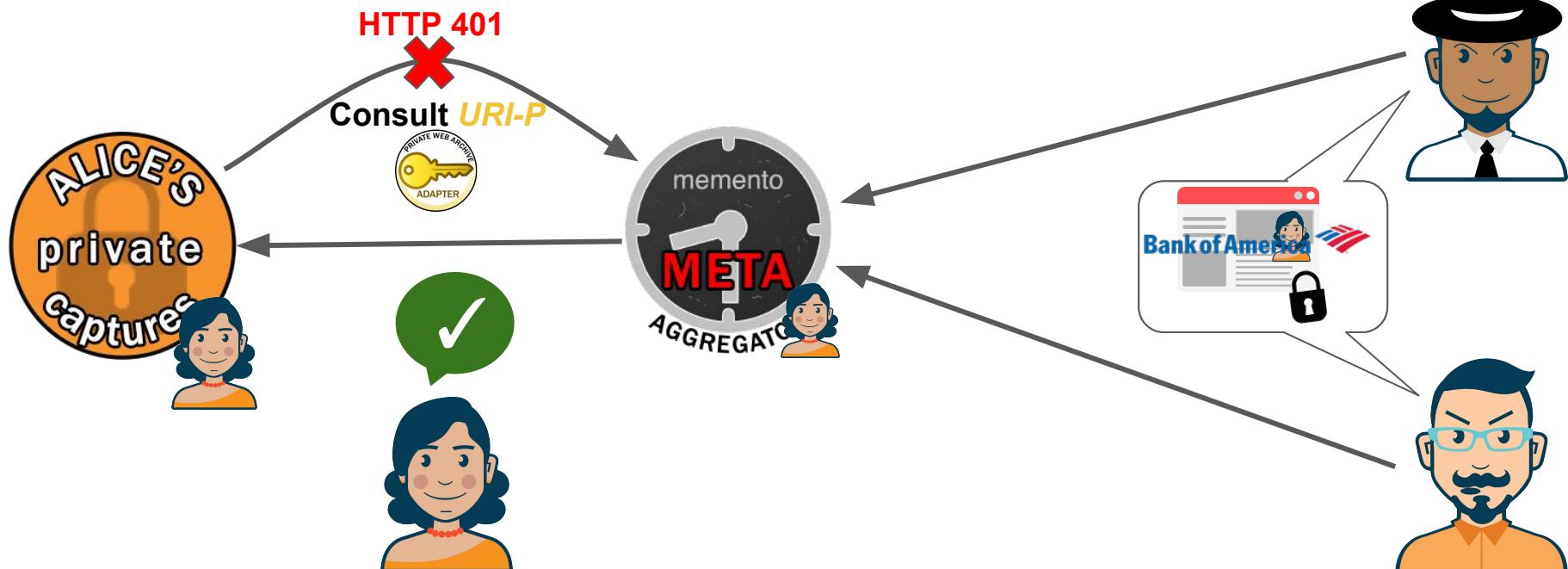


A Framework for Aggregating Public and Private Web Archives  
February 14, 2019  
*Mat Kelly*



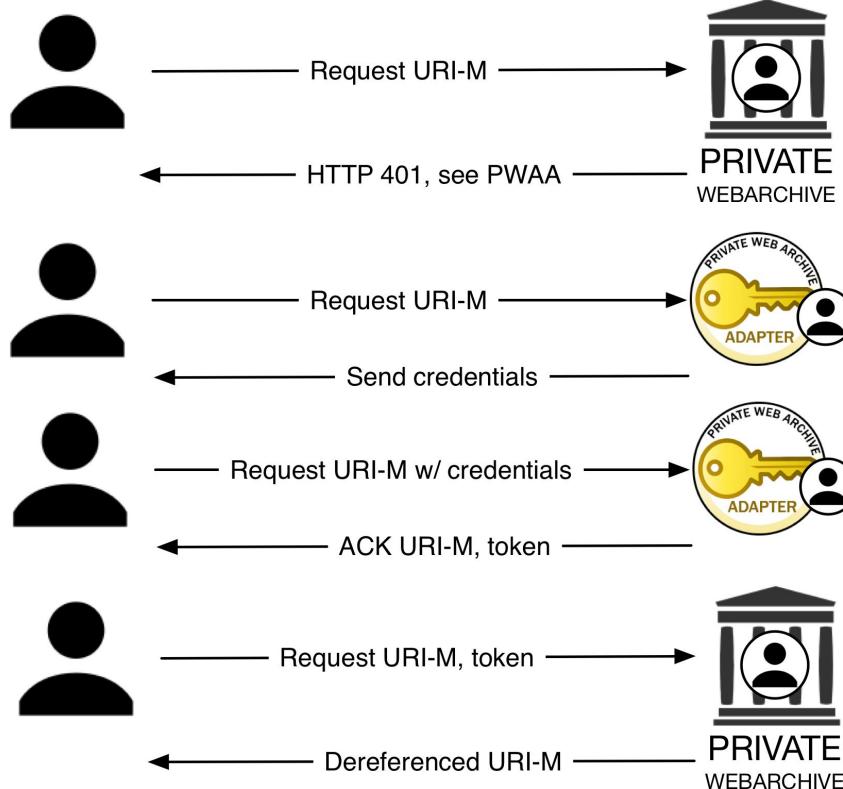
# Hooray, Aggregation!

RQ4: How can content that was captured behind authentication signal to Web archive replay systems that it requires special handling?





# Private Web Archive Adapter (PWAA)



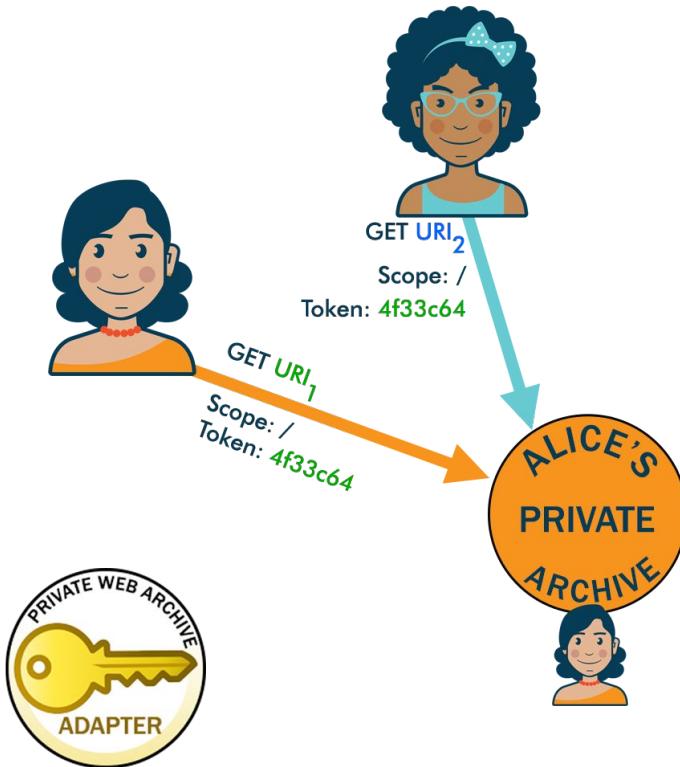
- Auth Layer for to encourage Private Web archive aggregation
- Typical OAuth 2.0 flow
- Auth role cohesive to PWAA
- Persistent access through tokenization

**RQ6:** What kinds of access control do users who create private Web archives need to **regulate access** to their archives?



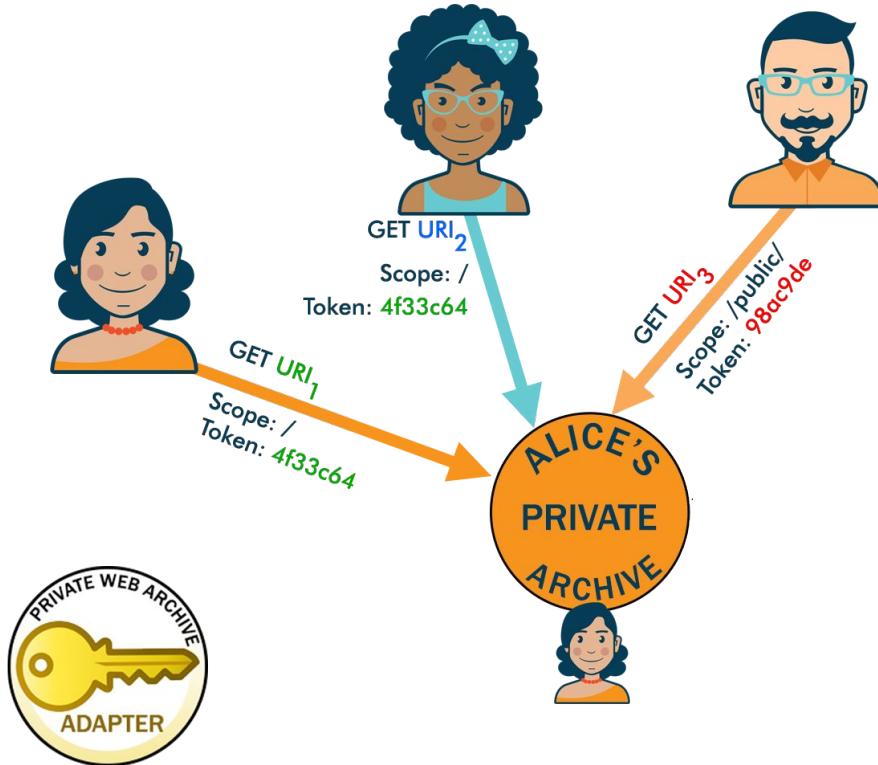
# PWAA - Sharing Tokens

RQ6: What kinds of access control do users who create private Web archives need to regulate access to their archives?

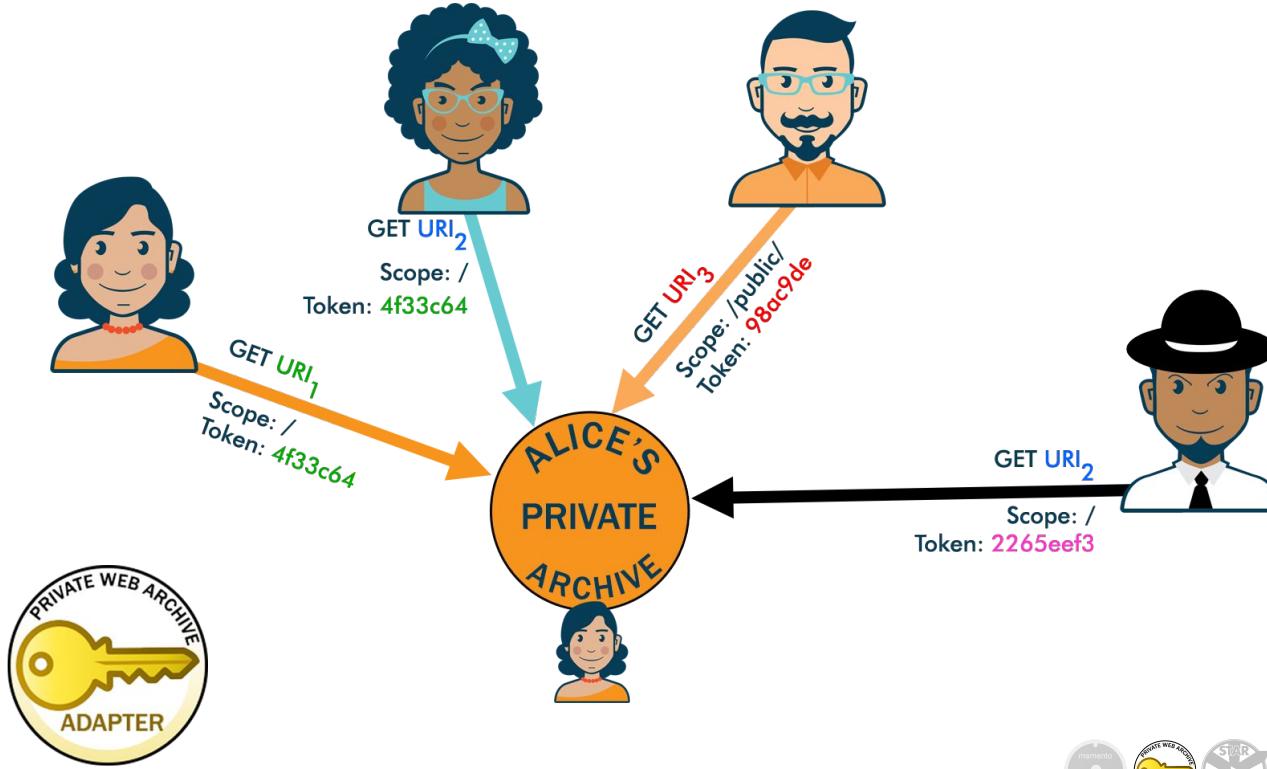




# PWAA - Previously Authorized

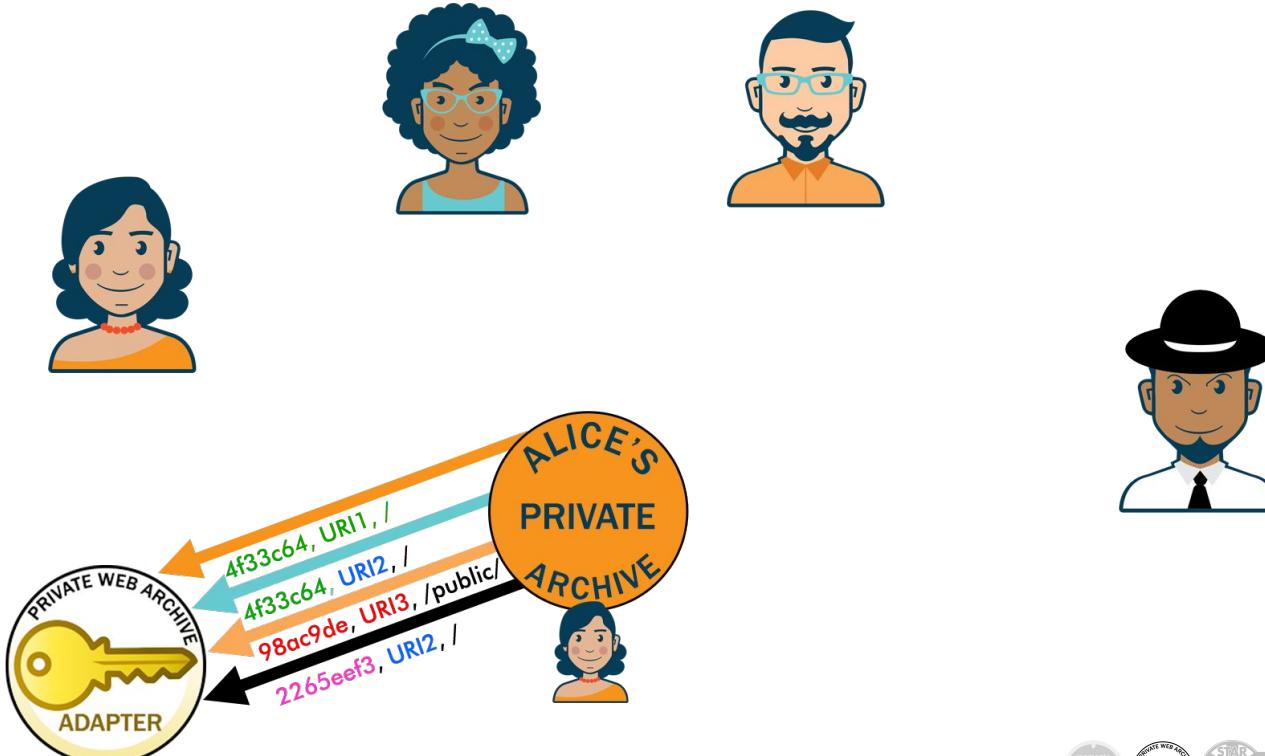


# PWAA - Unauthorized Request



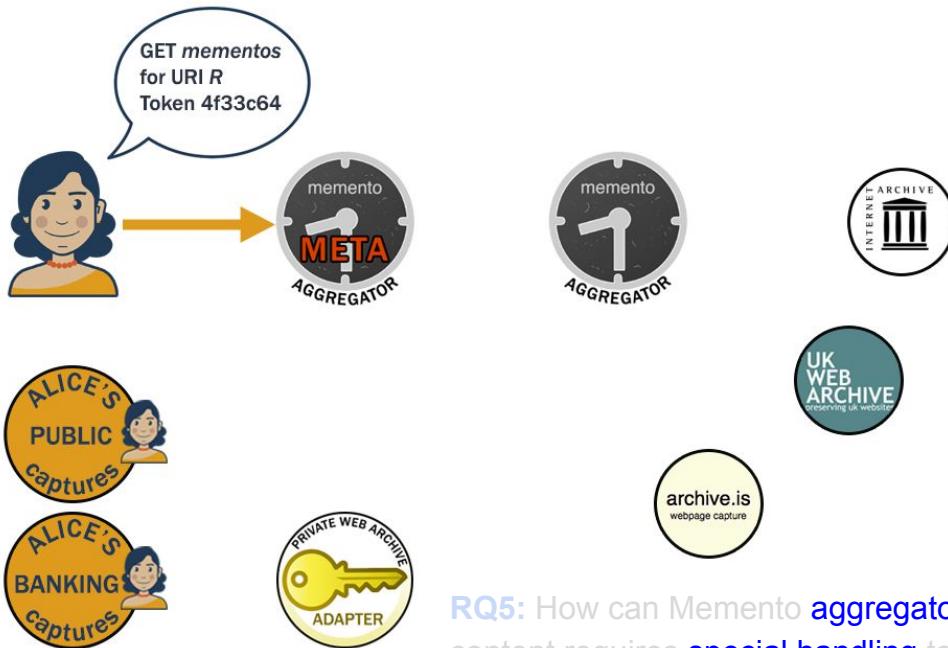
# PWAA - Sharing Tokens

RQ6: What kinds of access control do users who create private Web archives need to regulate access to their archives?





# Alice Passes Associative Token to MMA



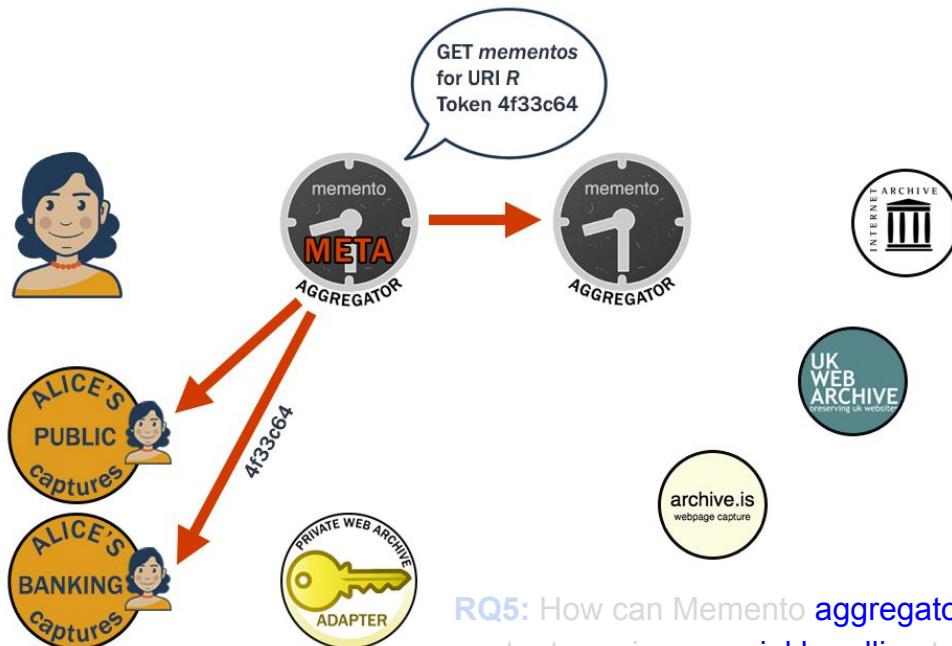
RQ5: How can Memento **aggregators indicate** that private Web archive content requires **special handling** to be replayed, despite being aggregated with publicly available Web archive content?





# MMA requests URI-R...

...relays token where applicable

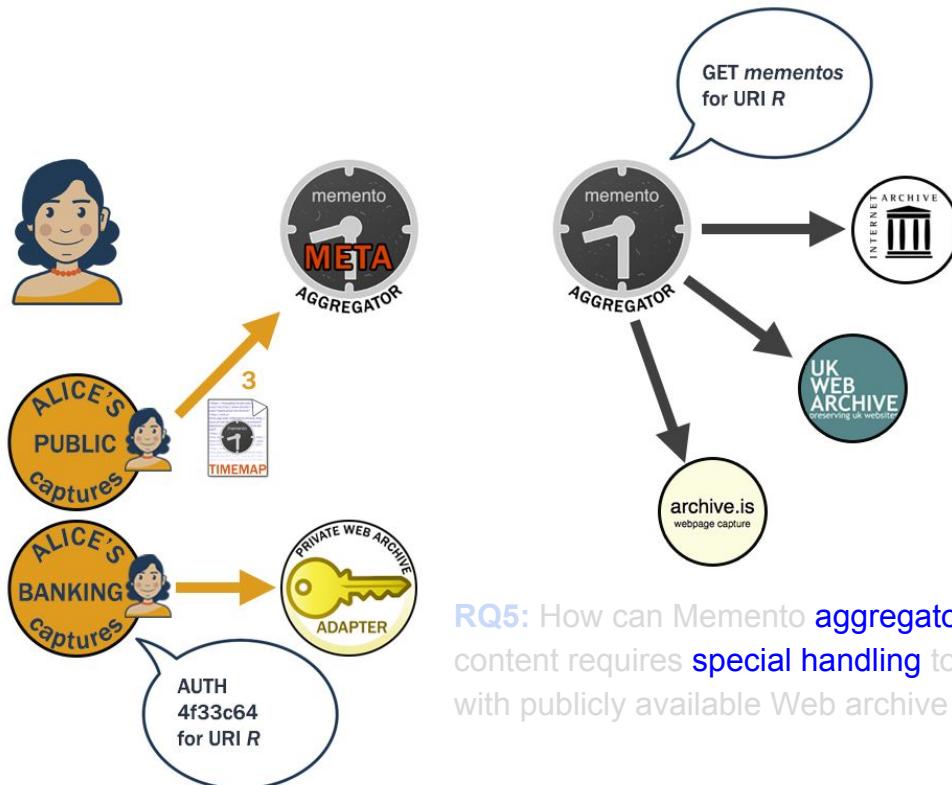


RQ5: How can Memento aggregators indicate that private Web archive content requires special handling to be replayed, despite being aggregated with publicly available Web archive content?





# Private Archive Validates with PWAA

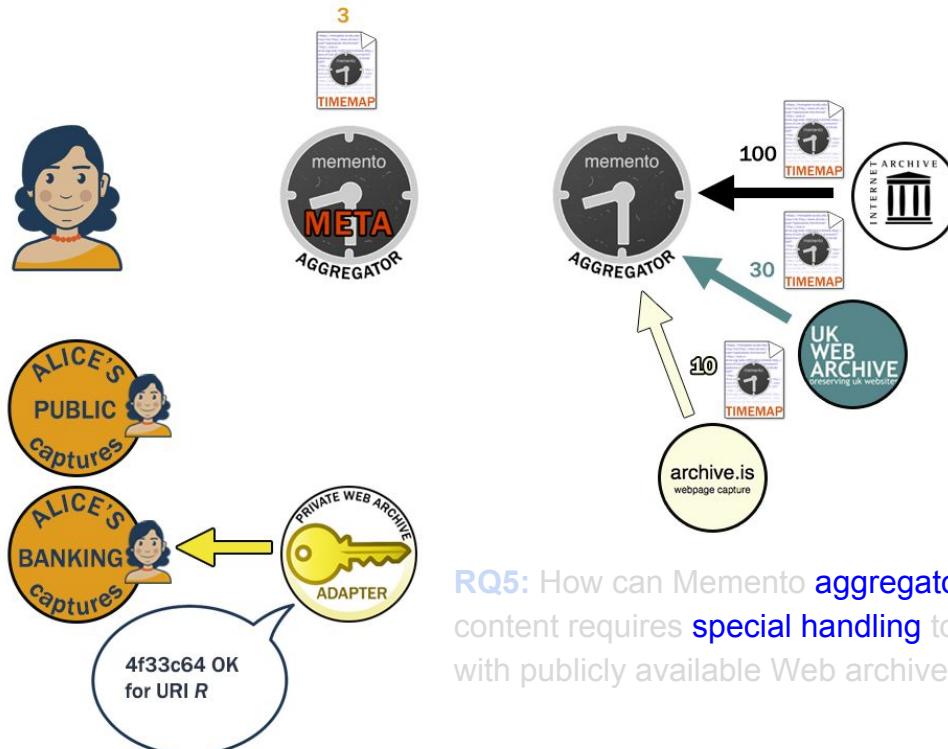


RQ5: How can Memento aggregators indicate that private Web archive content requires **special handling** to be replayed, despite being aggregated with publicly available Web archive content?





# PWAA Confirms Token

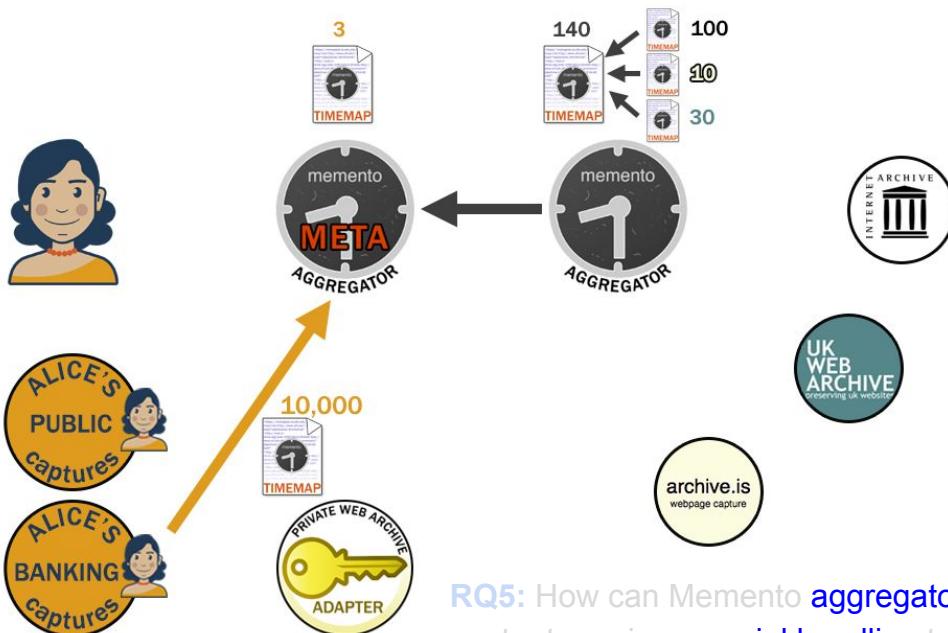


RQ5: How can Memento aggregators indicate that private Web archive content requires **special handling** to be replayed, despite being aggregated with publicly available Web archive content?





# Private Archive Returns Captures

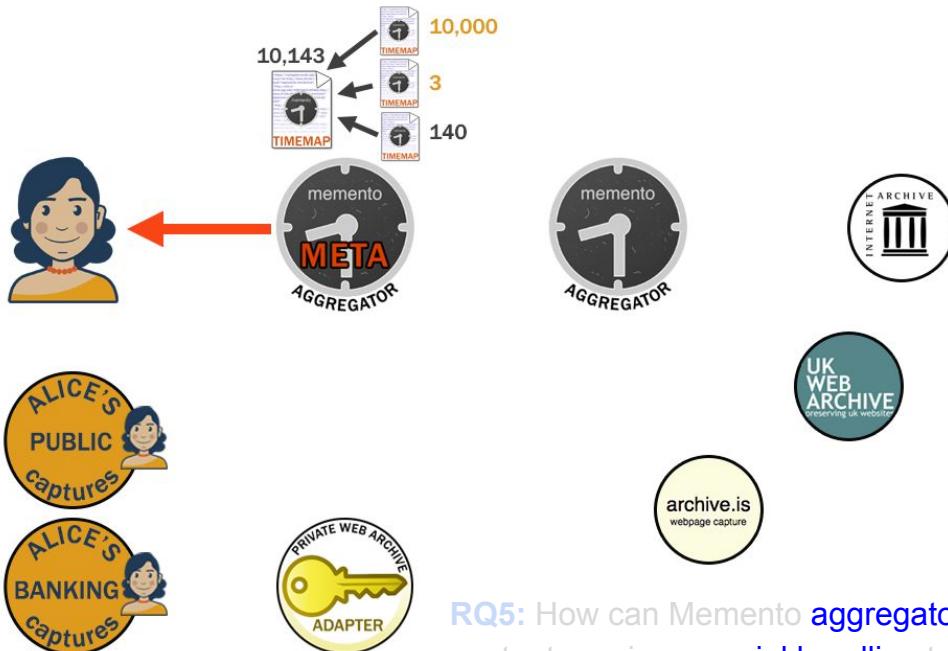


RQ5: How can Memento aggregators indicate that private Web archive content requires **special handling** to be replayed, despite being aggregated with publicly available Web archive content?





# MMA Aggregates, Associates Token



RQ5: How can Memento aggregators indicate that private Web archive content requires **special handling** to be replayed, despite being aggregated with publicly available Web archive content?

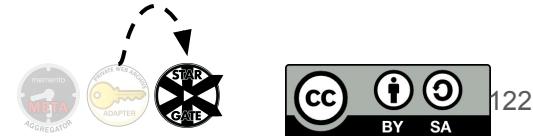


# PROPOSED FRAMEWORK

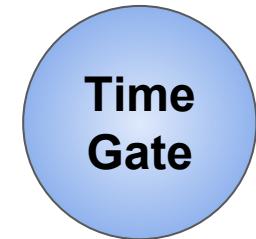
## Mementities



A Framework for Aggregating Public and Private Web Archives  
February 14, 2019  
*Mat Kelly*



# StarGate

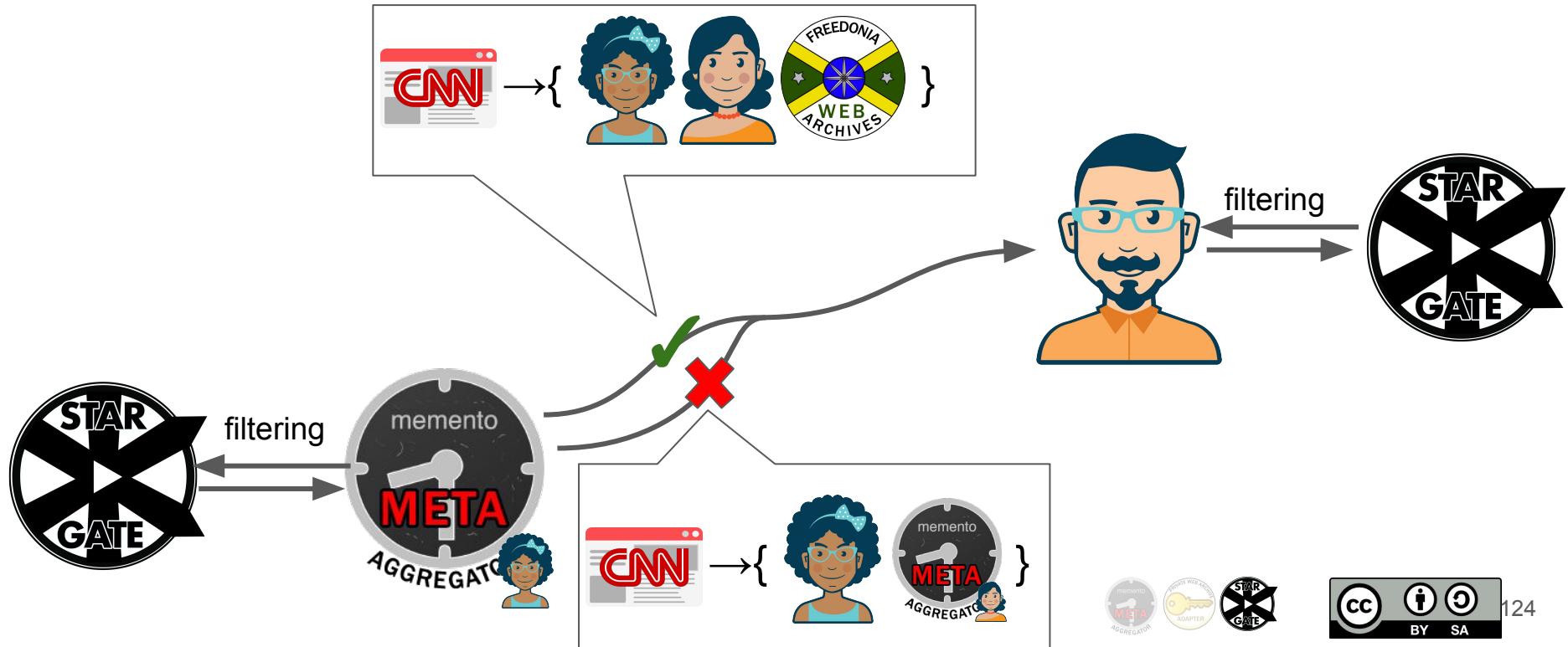


functional  
subseteq



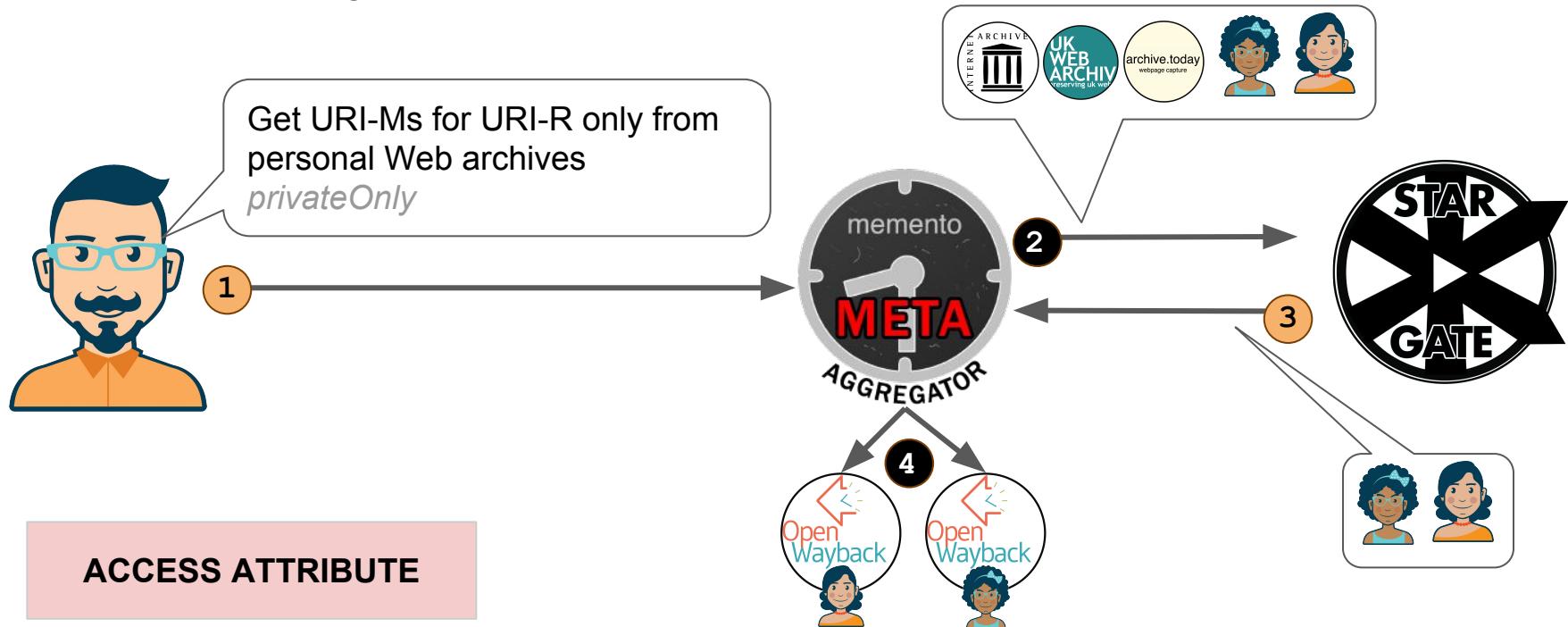
- Content negotiation in Web archives **beyond time**
- “Star” ~ wildcard (\*) → any dimension of negotiation
- Allow for queries like: *Only show me mementos...*
  - That are not redirects (*content-based attribute* HTTP Status  $\neq$  3XX)
  - Of a sufficient quality (*derived attribute* Memento Damage  $<$  0.4)
  - Are from personal Web archives (*access attribute* indicate Facebook.com memento is not a login page)

# Implicit Filtering via MMA or Directly (a la TG)



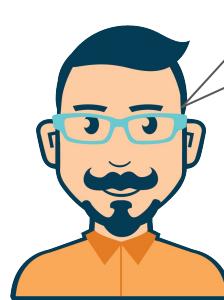
# Negotiation in the Privacy Dimension

(via short circuiting)

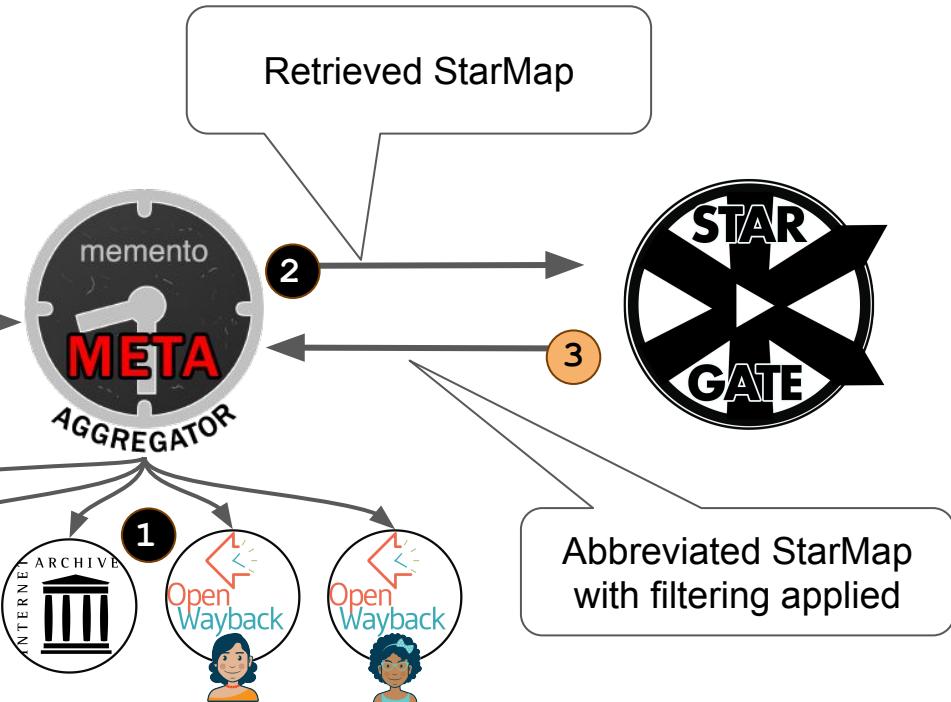


# Negotiation on Content-Based or Derived Attributes

(with response filtering)



Get URI-Ms for URI-R of good quality that are unique  
 $M_D < 0.25$ , *unique(simhash)*



CONTENT-BASED  
ATTRIBUTE

&

DERIVED ATTRIBUTE



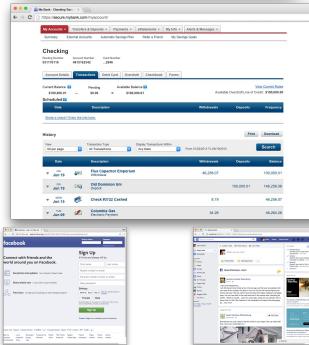
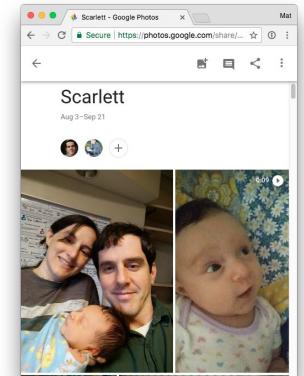
# Outline

- Introduction/Motivation
- Background
- Preliminary Research
- Proposed Framework
- **Evaluation Plan**

# Framework Evaluation

- Evaluation of mementity design decisions
- Costs of more expressive TimeMaps (StarMaps) and Link header enrichment
- Evaluation through implementation





# Evaluation of Mementity Design Decisions

- Effectiveness in resolving initial use cases and access patterns
- *"It was there yesterday, where did it go?"*
- *"Save this, but only for me."*
- *"I want to share this but control who can see it."*



A Framework for Aggregating Public and Private Web Archives  
February 14, 2019  
Mat Kelly

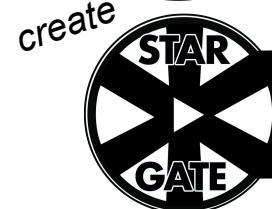
EVALUATION: [Design/Practicality](#) • Enrichment costs • Implementation



# Costs of more expressive TimeMaps (StarMaps) and Link header enrichment

- Computational:
  - Mostly server-side, potential to further leverage client
- Temporal
  - Required on variant generation
- Spatial
  - Permutation variant storage
- Access
  - Variant negotiation

# Evaluation Through Implementation

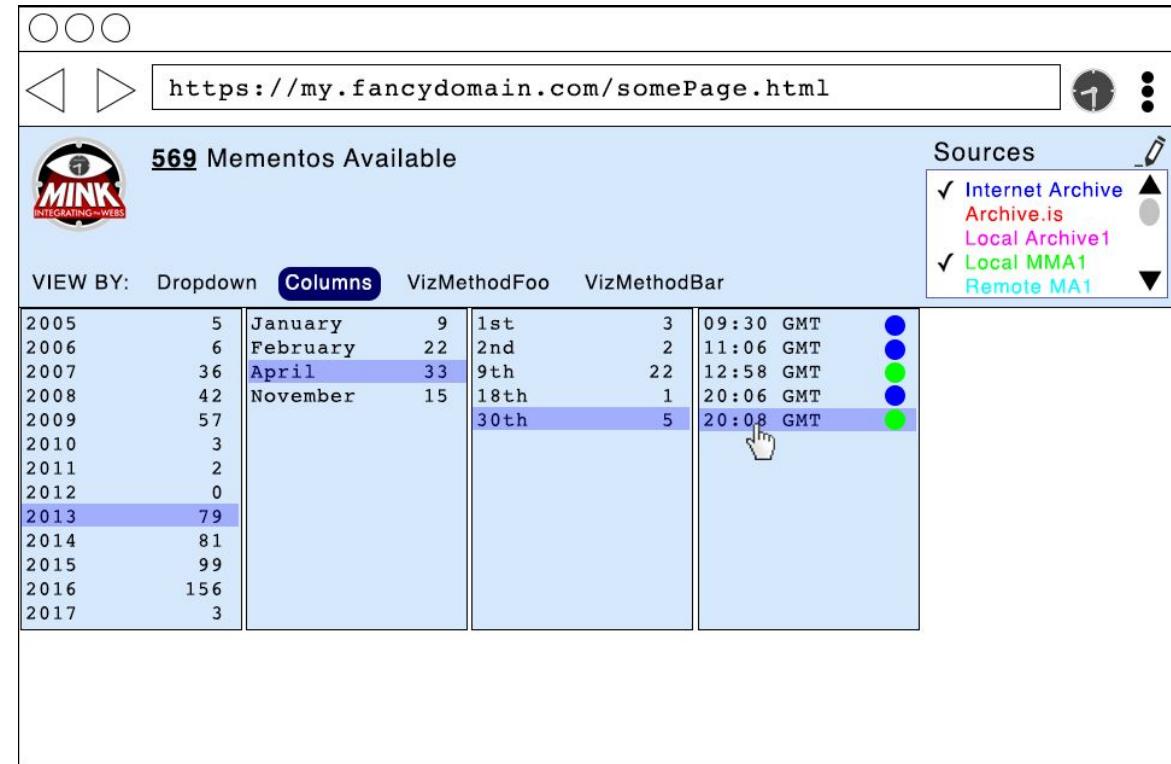


Extend for client-side archival specification

Exhibit features of an MMA

Regulate access to Private Web archives

Facilitate archival negotiation in more dimensions



# A Framework for Aggregating Private and Public Web Archives

Mat Kelly

Old Dominion University  
Web Science & Digital Libraries Research Group

Department of Computer Science  
Norfolk, Virginia USA

[mkelly@cs.odu.edu](mailto:mkelly@cs.odu.edu)



Seminar, Penn State University  
February 14, 2019



# Backup Slides

# Research Questions

**RQ1:** What sort of content is difficult to capture and replay for preservation from the perspective of a Web browser?

**RQ2:** How do Web browser APIs compare in potential functionality to the capabilities of archival crawlers?

**RQ3:** What issues exist for capturing and replaying content behind authentication?

**RQ4:** How can content that was captured behind authentication signal to Web archive replay systems that it requires special handling?

**RQ5:** How can Memento aggregators indicate that private Web archive content requires special handling to be replayed, despite being aggregated with publicly available Web archive content?

**RQ6:** What kinds of access control do users who create private Web archives need to regulate access to their archives?

# User Access Patterns

- Pattern 1: Single archive access
- Pattern 2: Aggregation of multiple Web archives



Pre-existing archival usage

- Pattern 3: Aggregator chaining
- Pattern 4: Aggregation with authentication
- Pattern 5: Aggregation including a hybrid public-private archive
- Pattern 6: Aggregation with filtering via MMA interaction
- Pattern 7: Aggregation with filtering via SG interaction



Contribution beyond proposal



# CDXJ: An Alternative TimeMap Format

```
<http://matkelly.com>; rel="original",
<http://localhost:1208/timemap/link/http://matkelly.com>;
rel="self"; type="application/link-format",
<http://web.archive.org/web/20060514123511/http://www.mat
kelly.com:80/>; rel="first memento"; datetime="Sun, 14
May 2006 12:35:11 GMT",
<http://web.archive.org/web/20060516213852/http://www.mat
kelly.com/>; rel="memento"; datetime="Tue, 16 May 2006
21:38:52 GMT",
...
<http://web.archive.org/web/20180128152125/http://matkell
y.com>; rel="memento"; datetime="Sun, 28 Jan 2018
15:21:25 GMT",
<http://web.archive.org/web/20180319141920/http://matkell
y.com/>; rel="last memento"; datetime="Mon, 19 Mar 2018
14:19:20 GMT",
<http://localhost:1208/timemap/link/http://matkelly.com>;
rel="timemap"; type="application/link-format",
<http://localhost:1208/timemap/json/http://matkelly.com>;
rel="timemap"; type="application/json",
<http://localhost:1208/timemap/cdxj/http://matkelly.com>;
rel="timemap"; type="application/cdxj+ors",
<http://localhost:1208/timegate/http://matkelly.com>;
rel="timegate"
```

```
!context ["http://tools.ietf.org/html/rfc7089"]
!id {"uri": "http://localhost:1208/timemap/cdxj/http://matkelly.com"}
!keys ["memento_datetime_YYYYMMDDhhmmss"]
!meta {"original_uri": "http://matkelly.com"}
!meta {"timegate_uri":
"http://localhost:1208/timegate/http://matkelly.com"}
!meta {"timemap_uri": {"link_format":
"http://localhost:1208/timemap/link/http://matkelly.com", "json_format":
"http://localhost:1208/timemap/json/http://matkelly.com", "cdxj_format":
"http://localhost:1208/timemap/cdxj/http://matkelly.com"}}
20060514123511 {"uri":
"http://web.archive.org/web/20060514123511/http://www.matkelly.com:80/",
"rel": "first memento", "datetime": "Sun, 14 May 2006 12:35:11 GMT"}
20060516213852 {"uri":
"http://web.archive.org/web/20060516213852/http://www.matkelly.com/",
"rel": "memento", "datetime": "Tue, 16 May 2006 21:38:52 GMT"}
...
20180128152125 {"uri":
"http://web.archive.org/web/20180128152125/http://matkelly.com", "rel":
"memento", "datetime": "Sun, 28 Jan 2018 15:21:25 GMT"}
20180319141920 {"uri":
"http://web.archive.org/web/20180319141920/http://matkelly.com", "rel":
"last memento", "datetime": "Mon, 19 Mar 2018 14:19:20 GMT"}
```

## Link (RFC 7089) TimeMap

Original URI (URI-R)

Other TimeMaps (URI-Ts)

## CDXJ TimeMap

Relative Relations

# Private & Public Archives May Differ for the Same URI



# Should Public Archives *Really* Capture the Private Web?

