A Framework for Aggregating Private and Public Web Archives

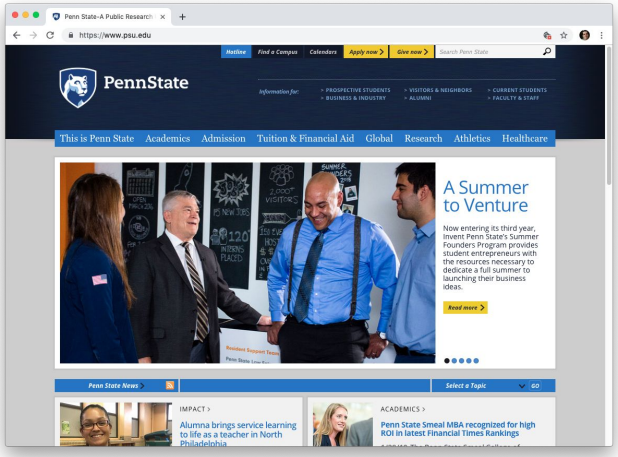
Mat Kelly
Old Dominion University
Web Science & Digital Libraries Research Group
Department of Computer Science
Norfolk, Virginia USA
mkelly@cs.odu.edu

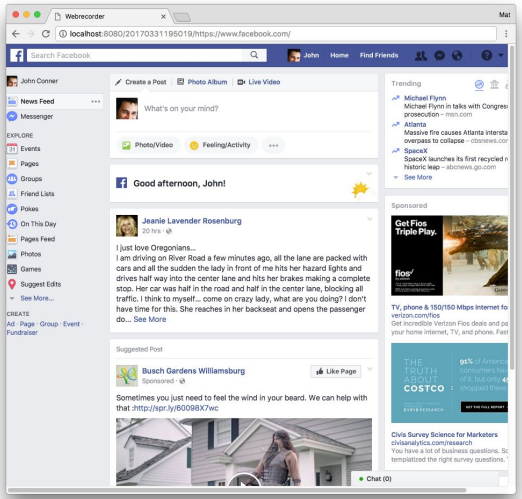
OLD DOMINION UNIVERSITY

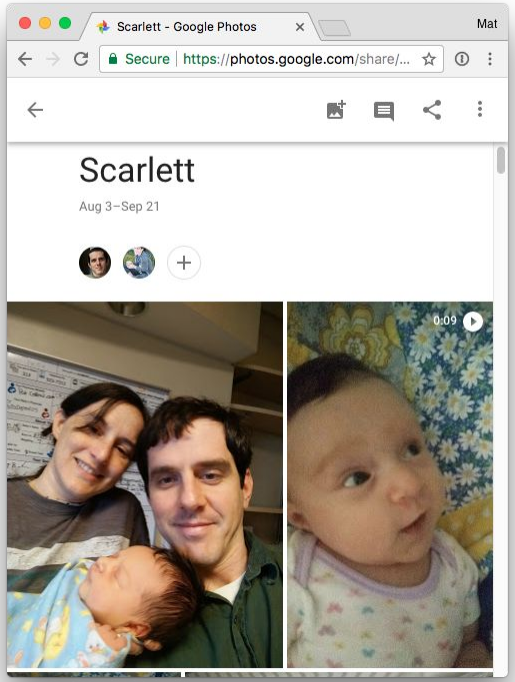
Seminar, Penn State University
February 14, 2019

WSDL Old Dominion University

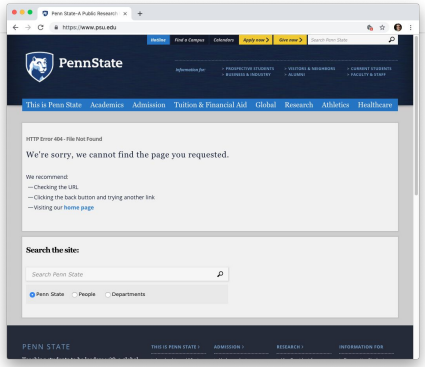
The Web

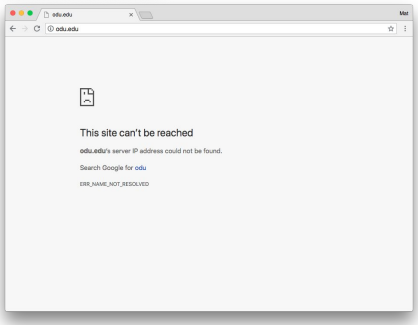


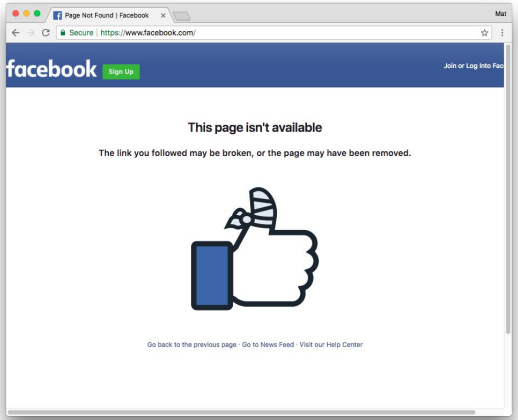


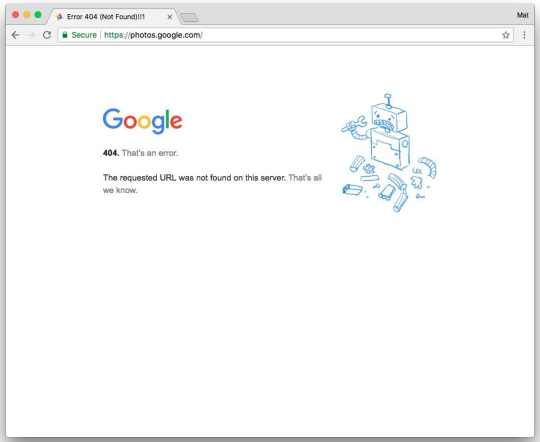


The Web is Ephemeral

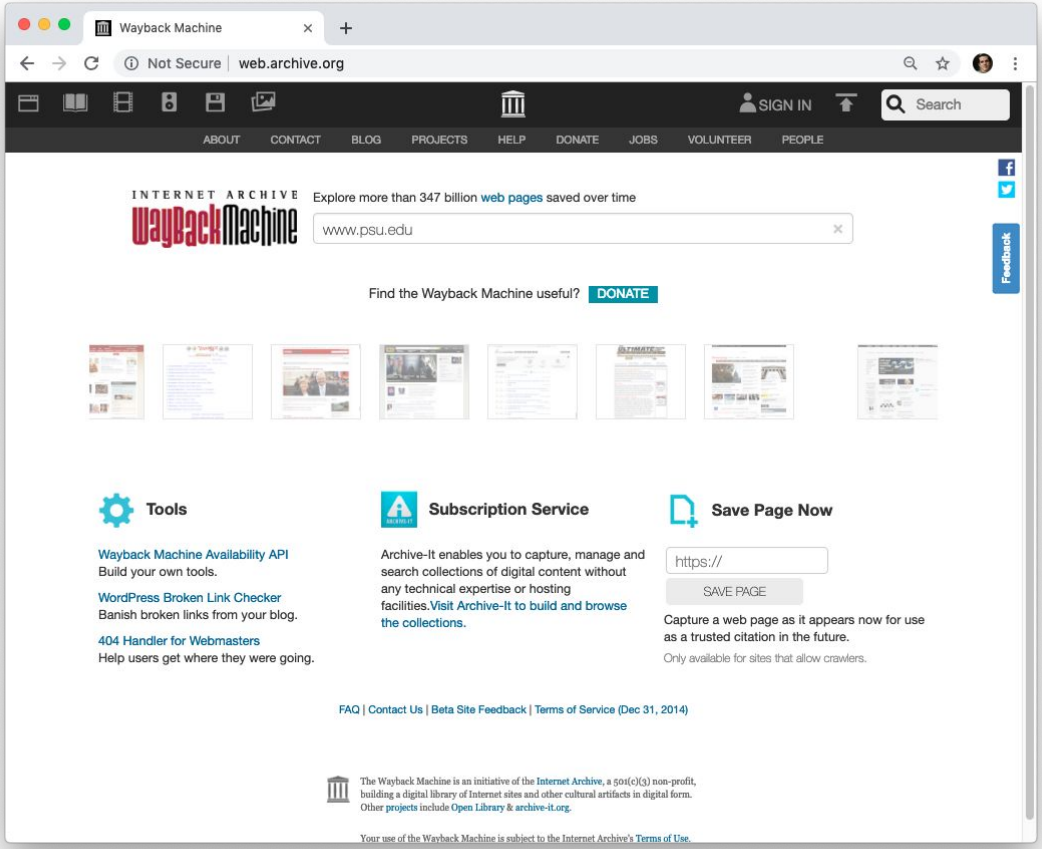






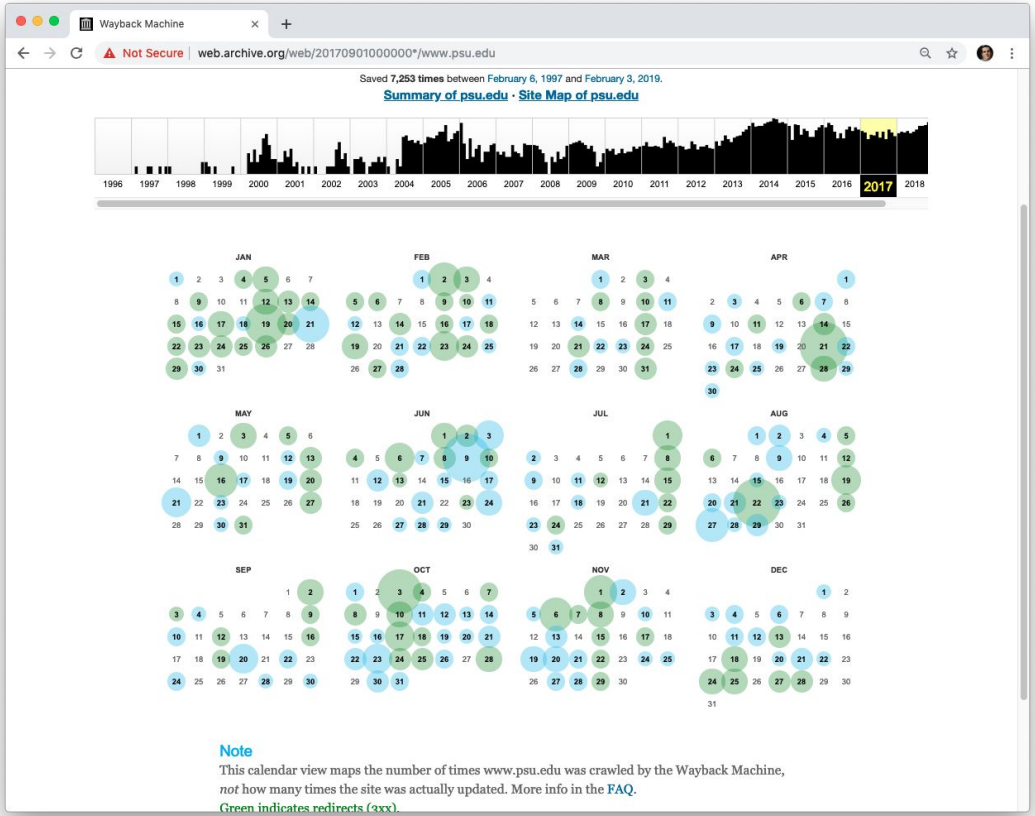


Web Archives to the Rescue: Typical Access



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

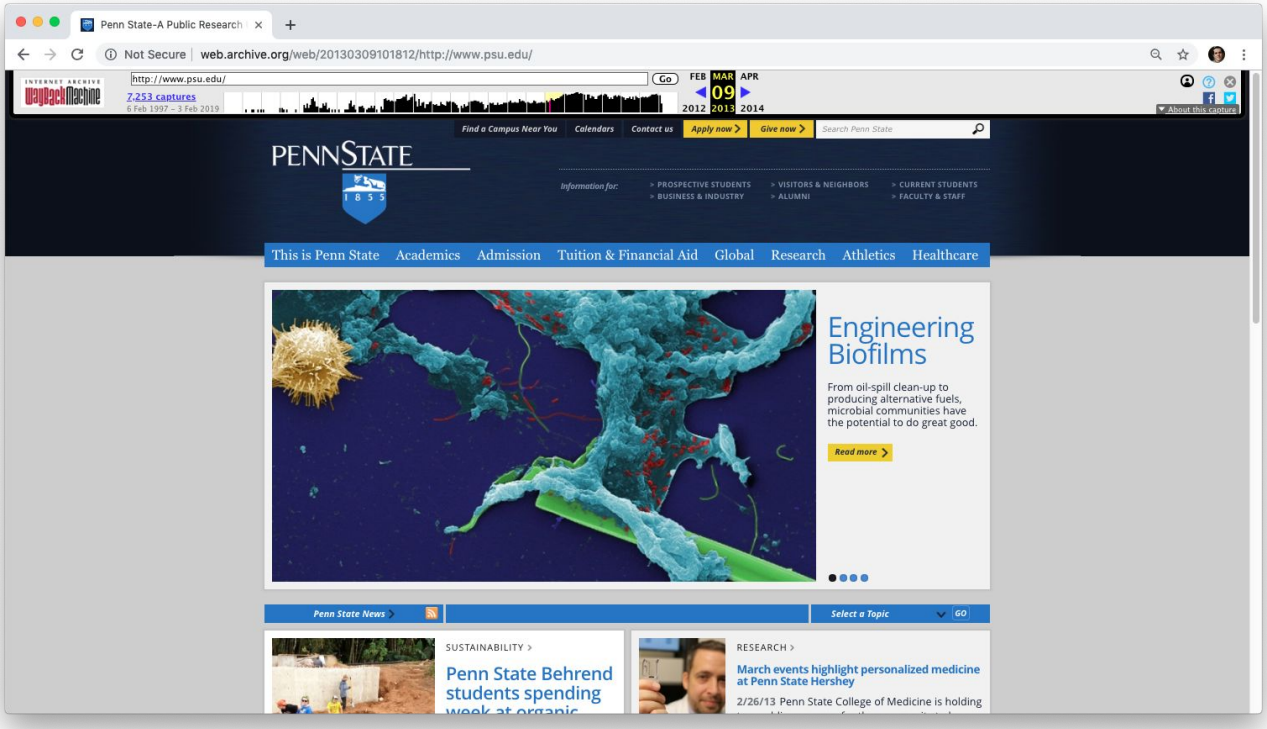


4. Locate the capture on the calendar or histogram view

5. Select the year/capture for the day

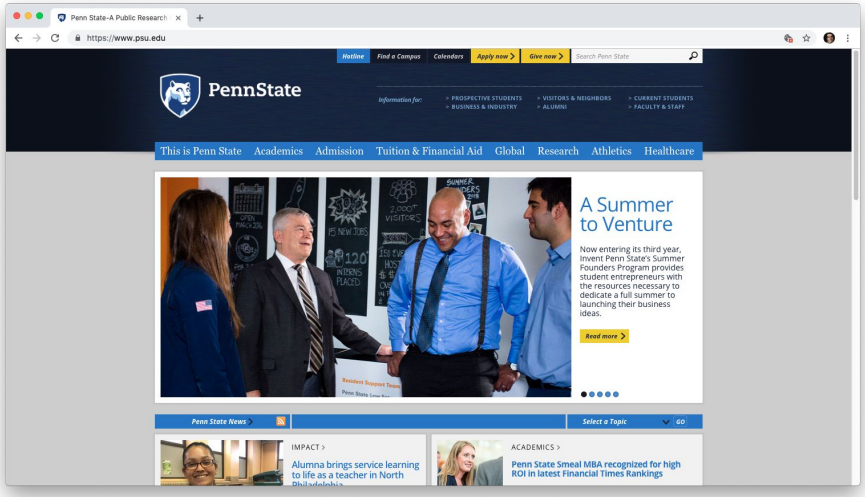
6. Repeat until you find the closest date and time

7. Finally, view the capture



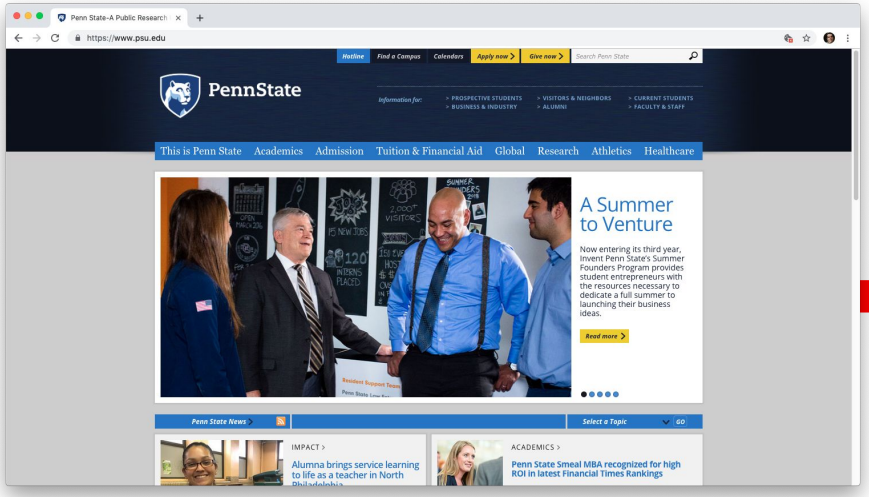
from **March 9, 2013**

Web Archiving - Live Web psu.edu

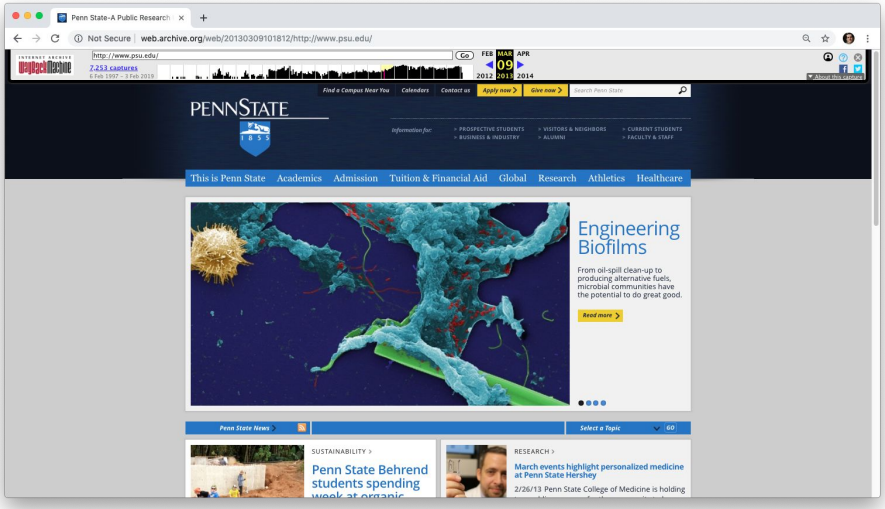


Now

Web Archiving - Archival Capture

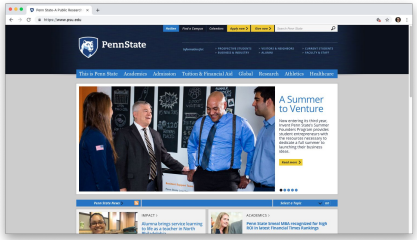


Now

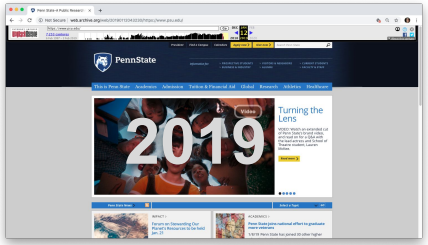


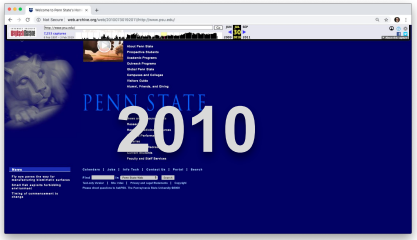
March 9, 2013

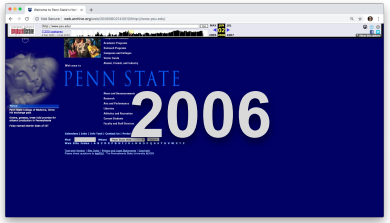
Web Archiving



Associate live Web URIs

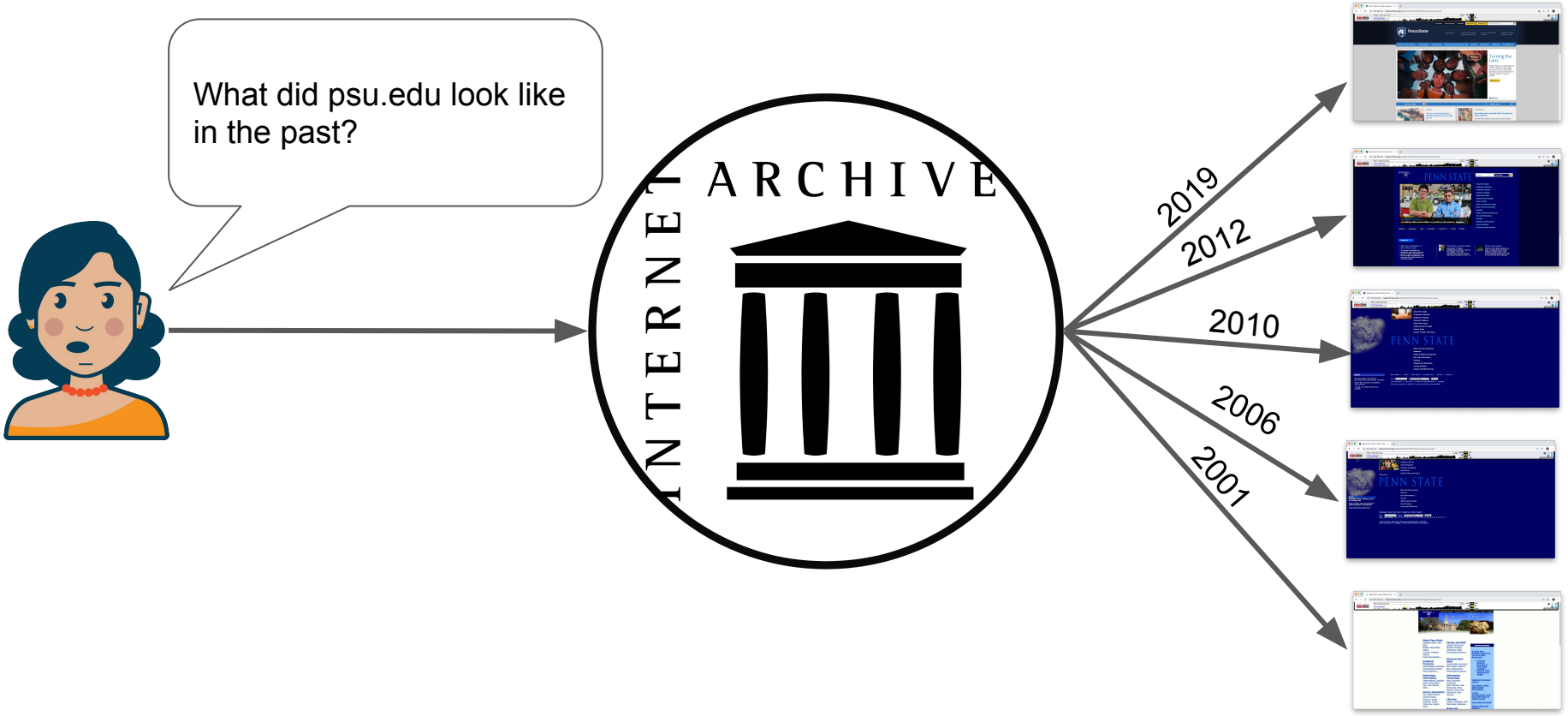






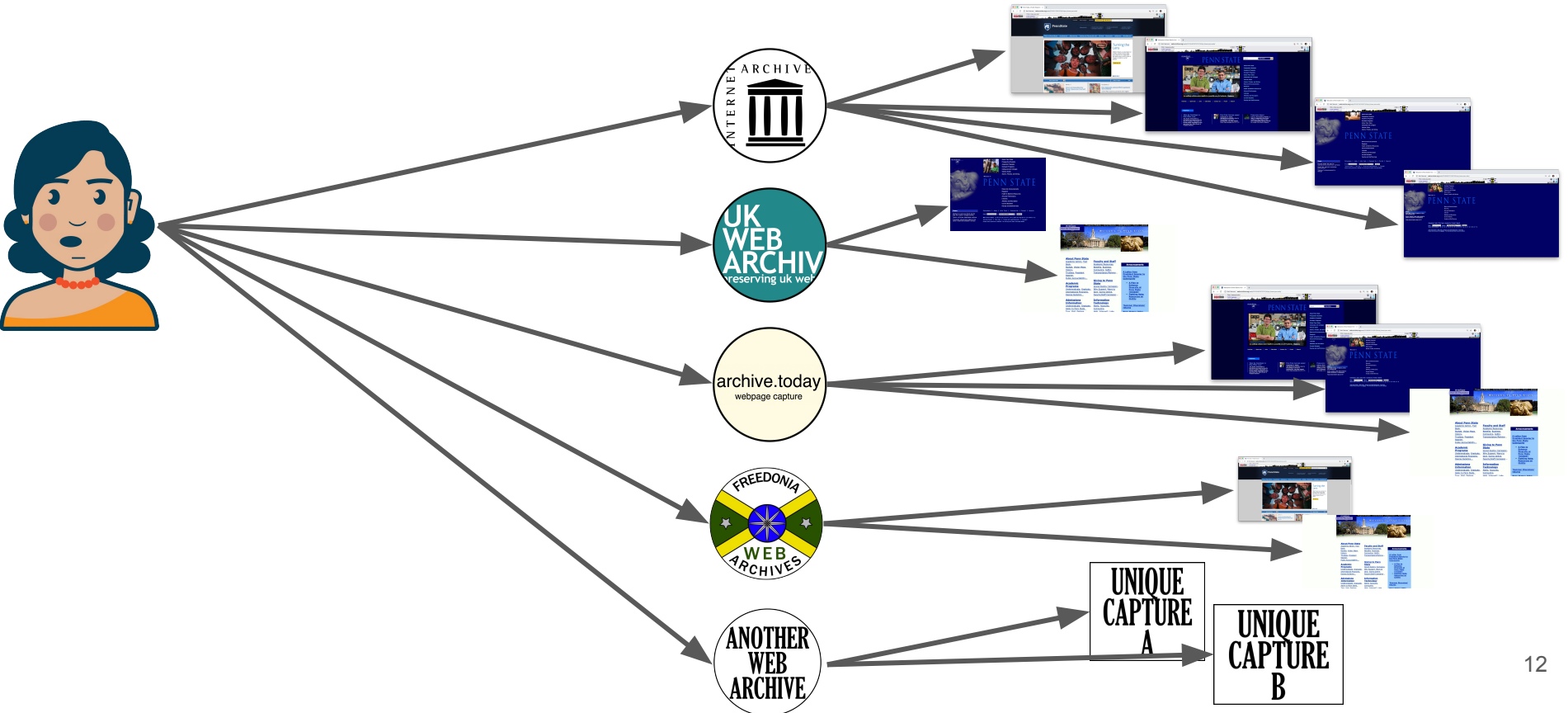
With their archived representations

Web Archives provides access to the **Web** that *was*

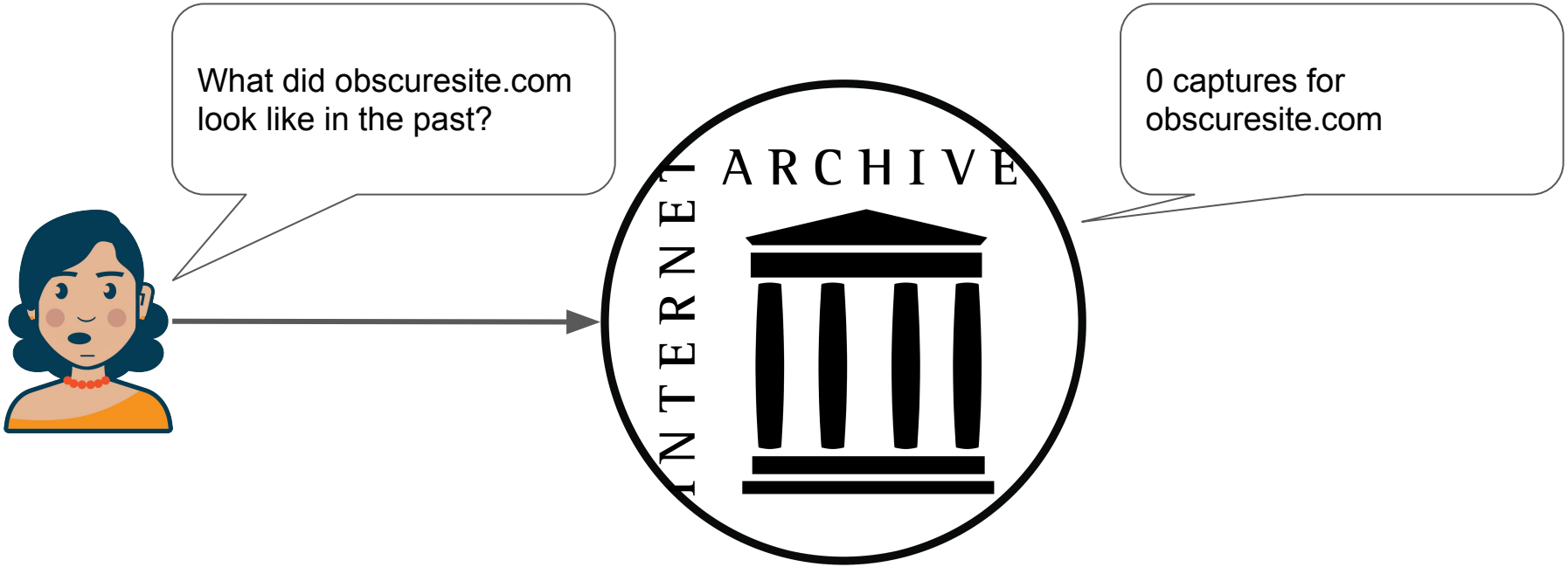


Multiple archival efforts (3 of many)

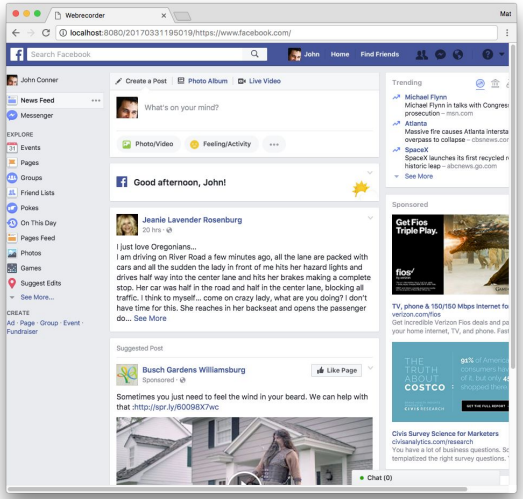
More archives produces a more comprehensive picture

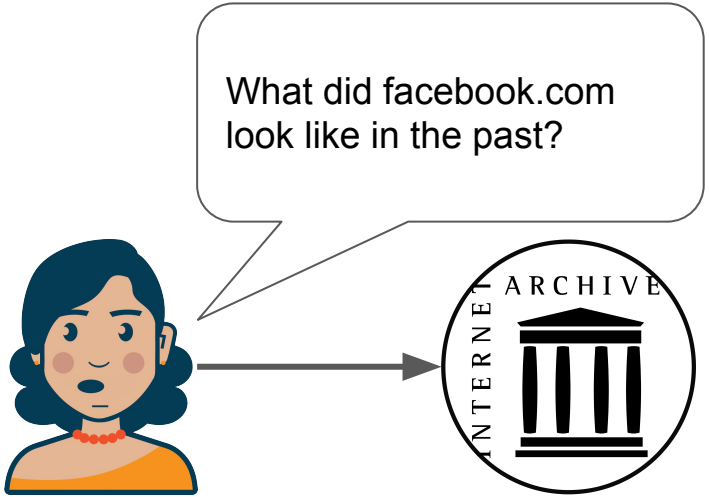


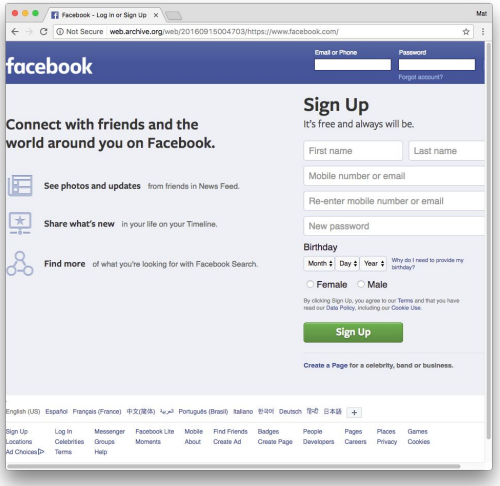
Even then, not everything is preserved



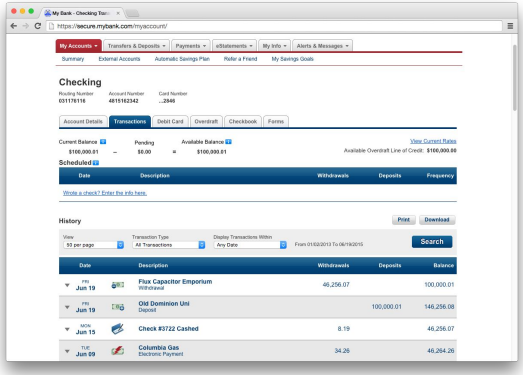
User sees on live Web may not be what is captured

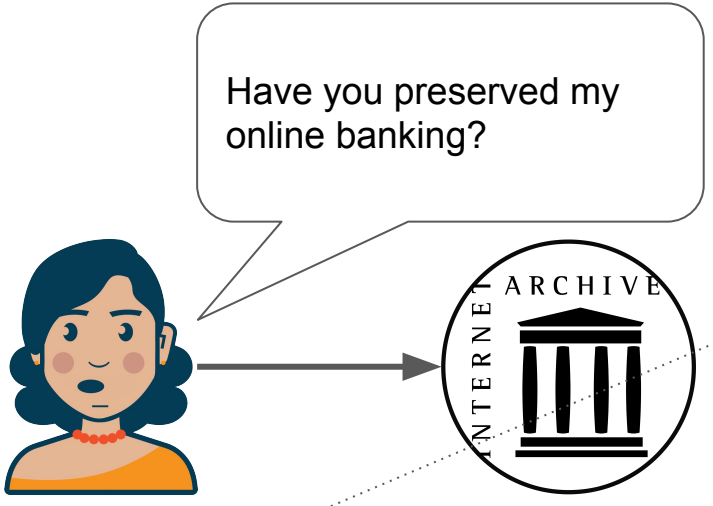


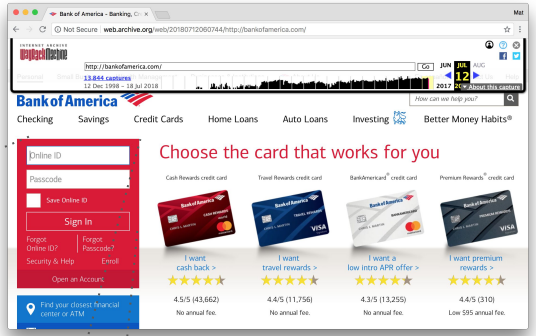


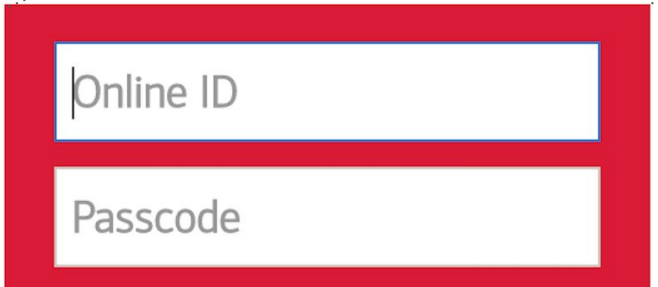


...And oftentimes that is for the best

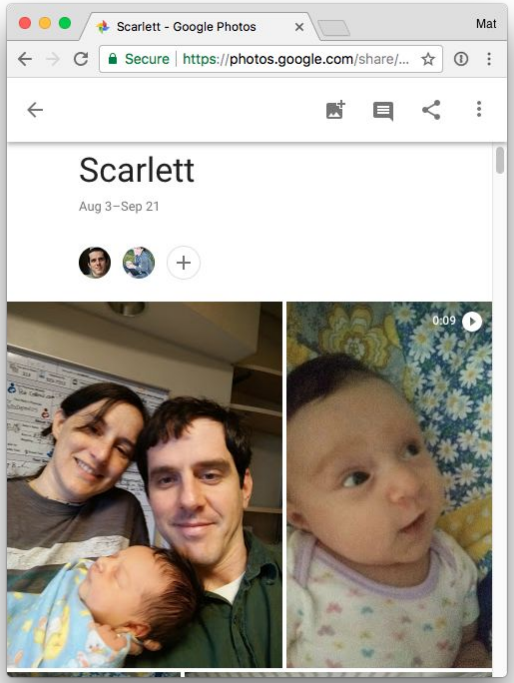


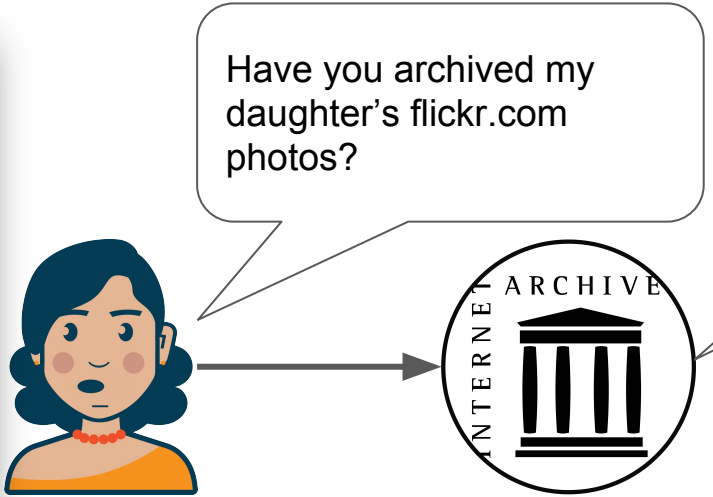




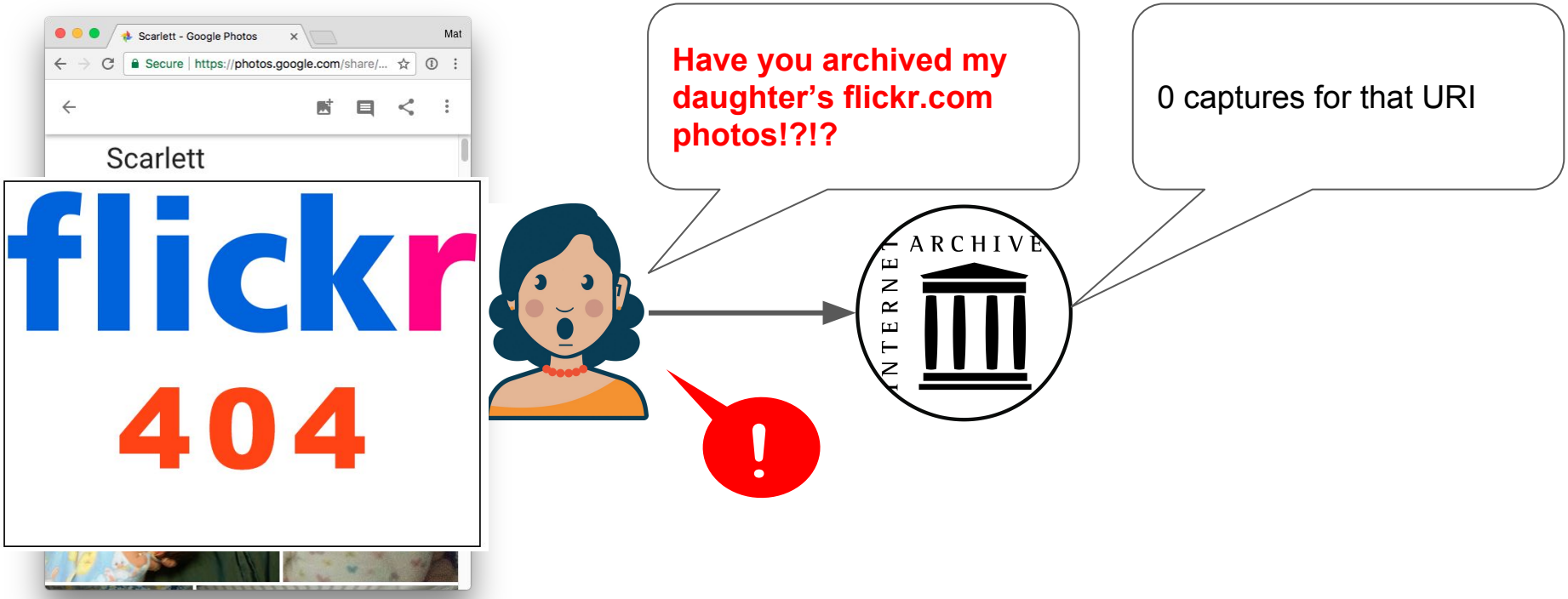


Other times, we may want our content archived





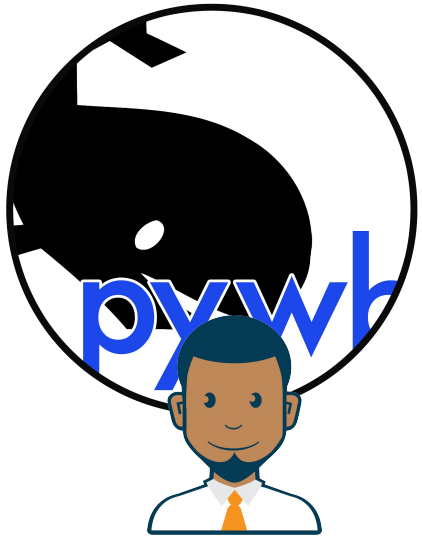
...especially when it has disappeared



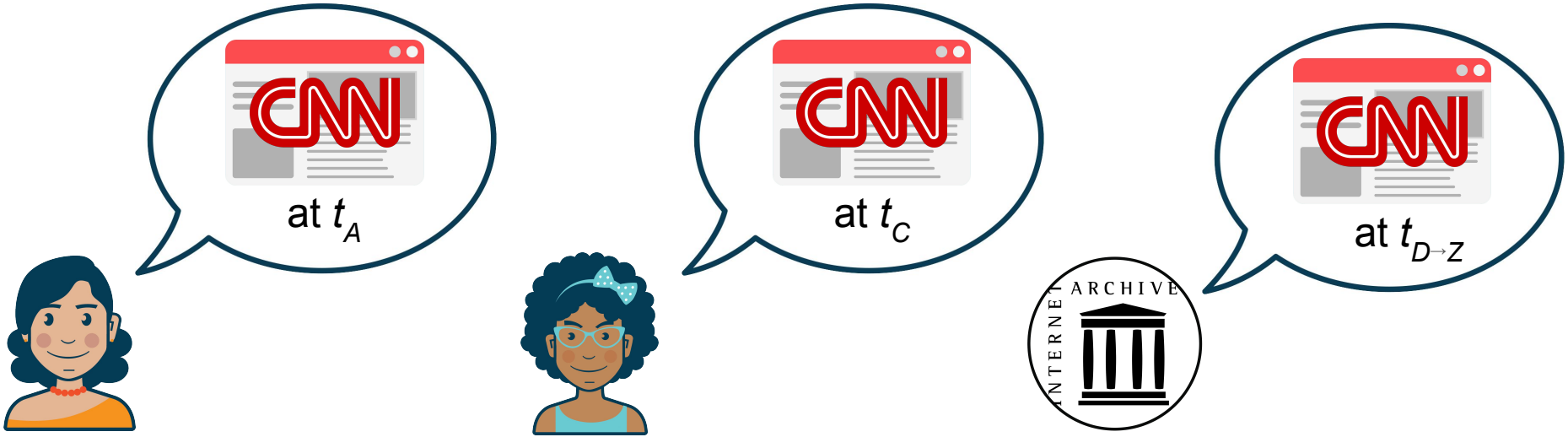
“Save this, but only for me.”

- **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



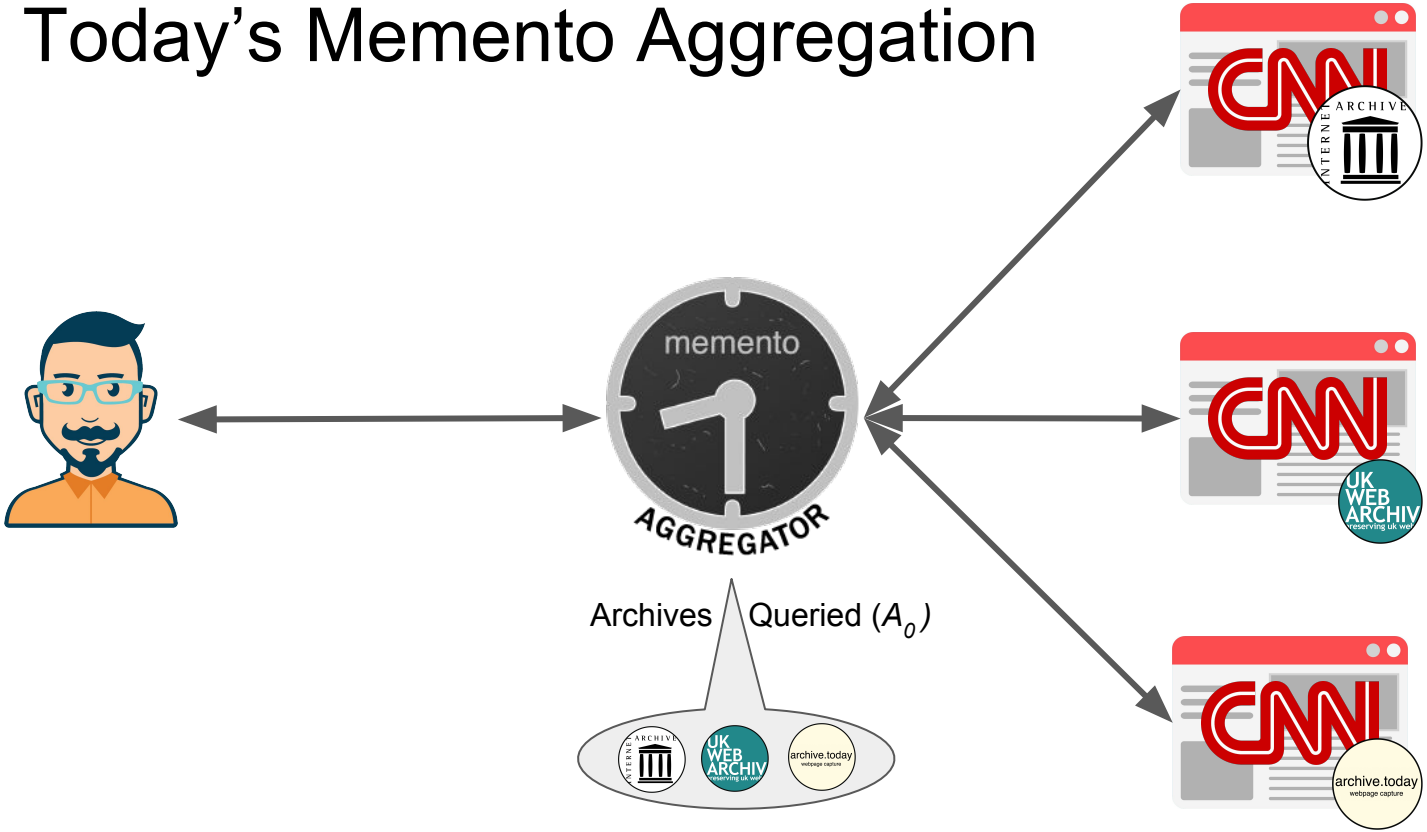
Captures from Institutional and Personal Sources



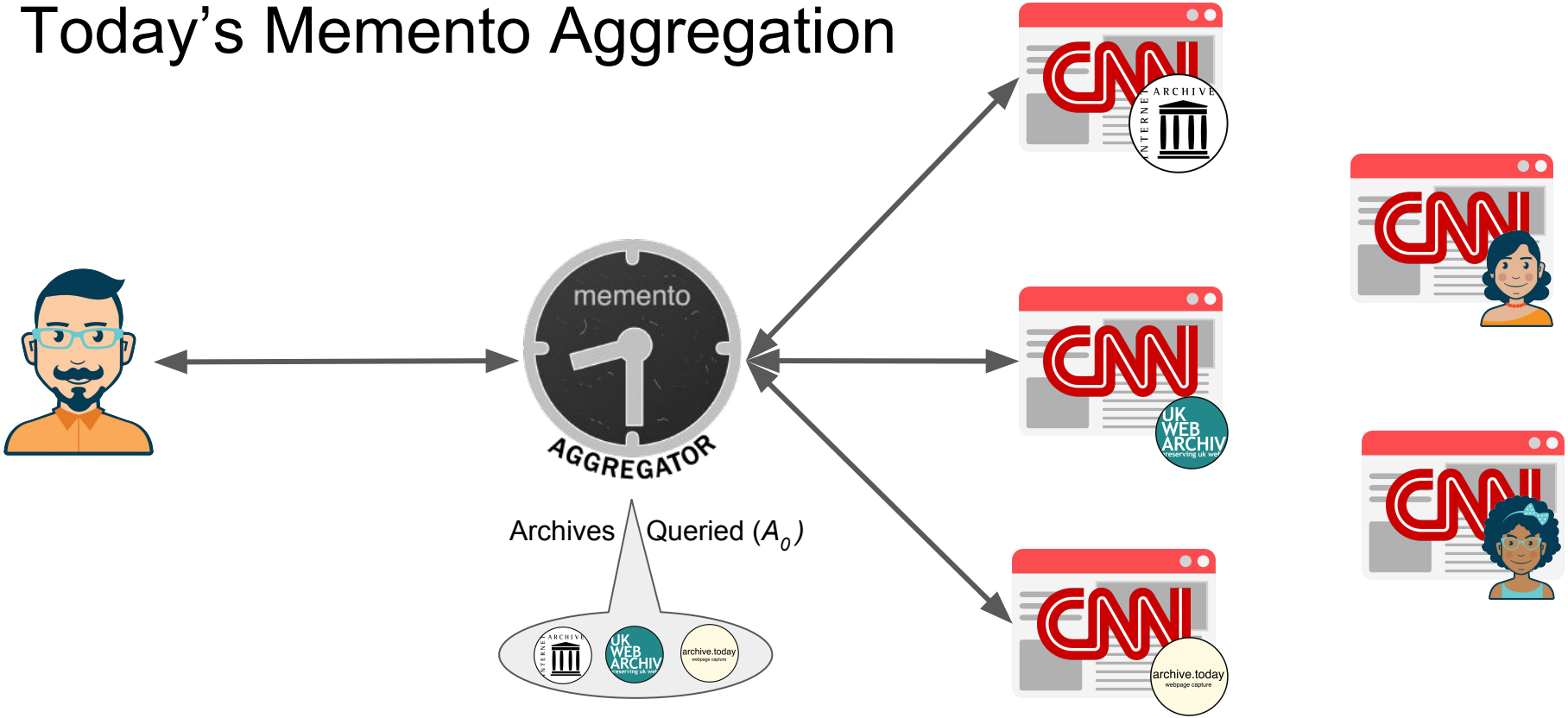
Memento Facilitates this Aggregation

RFC7089

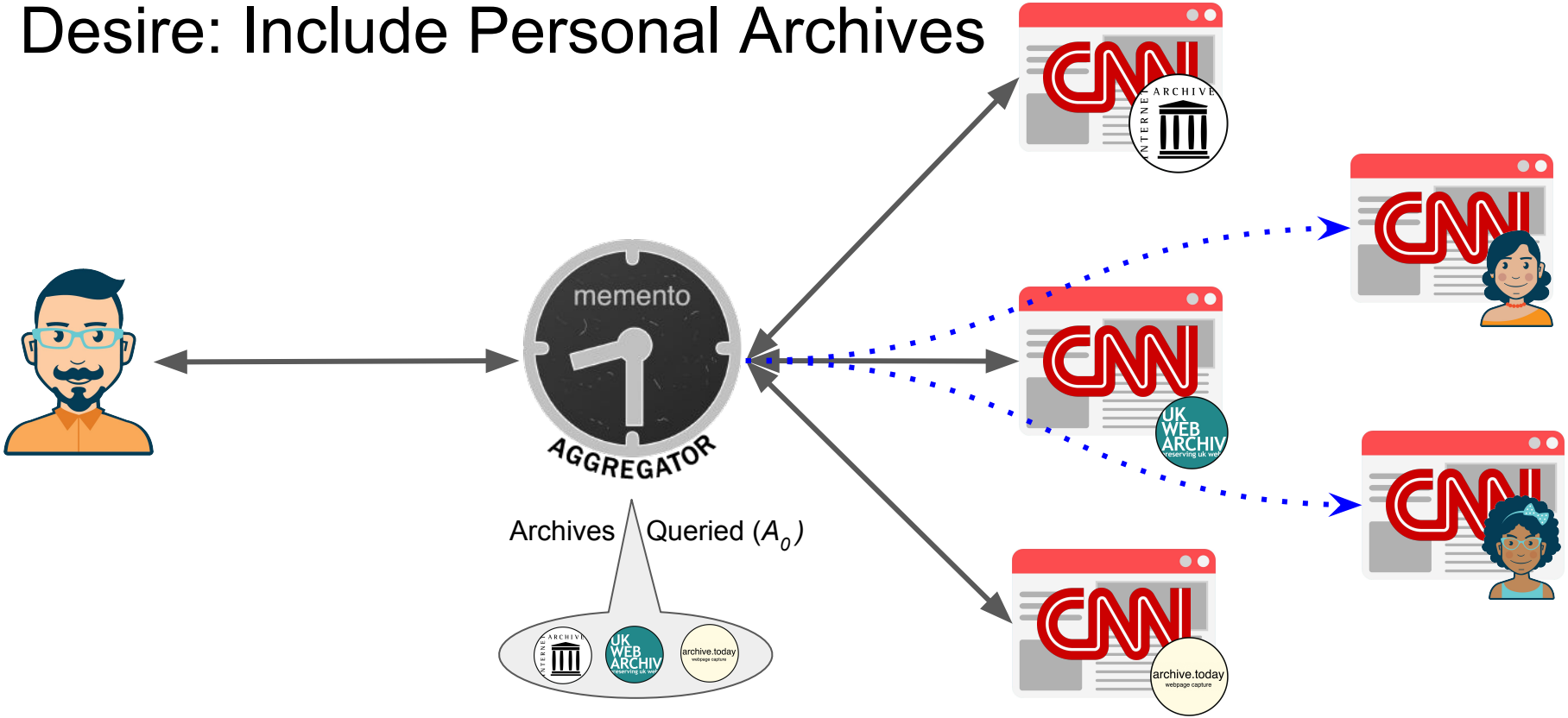
Today's Memento Aggregation



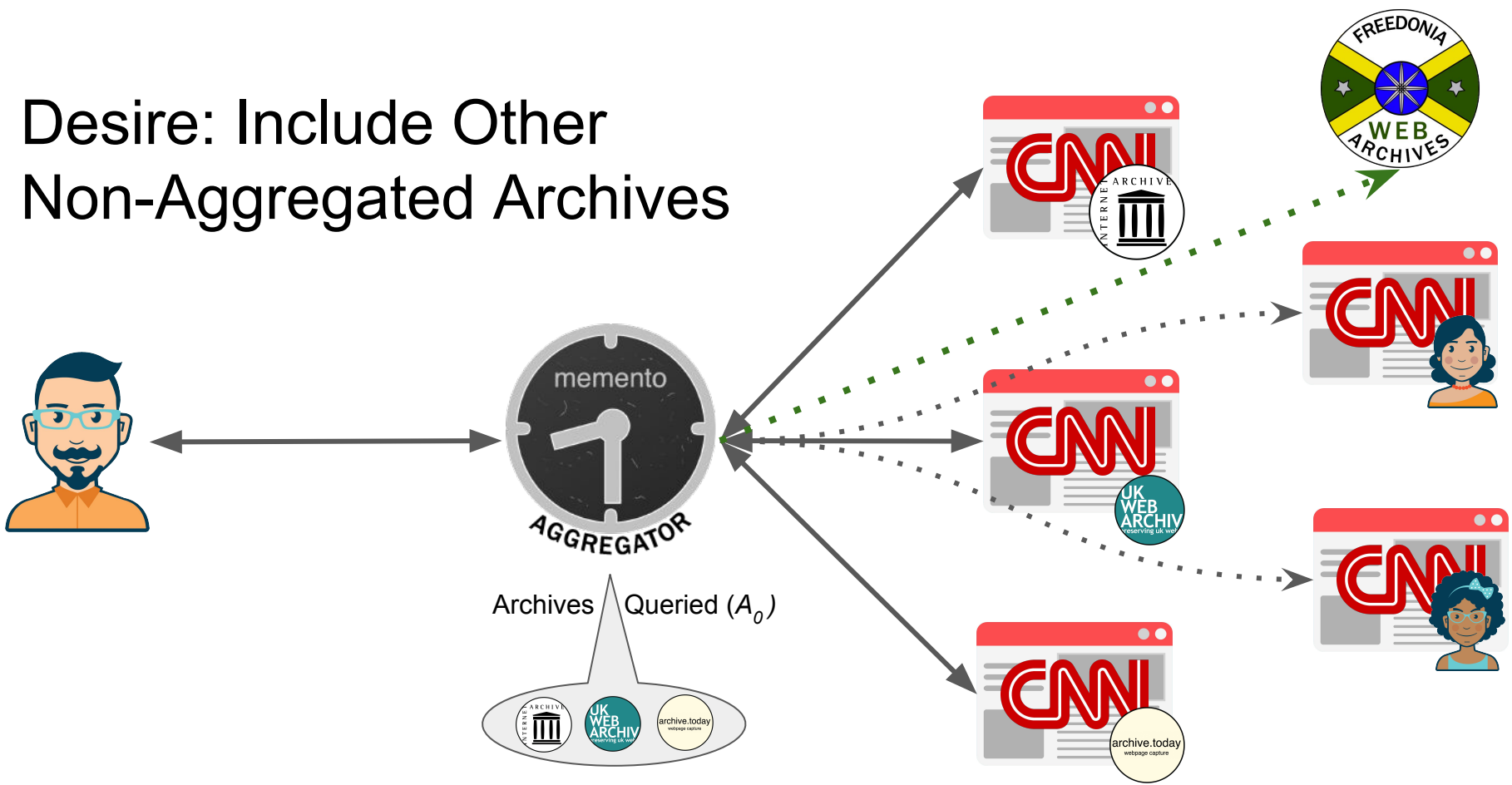
Today's Memento Aggregation



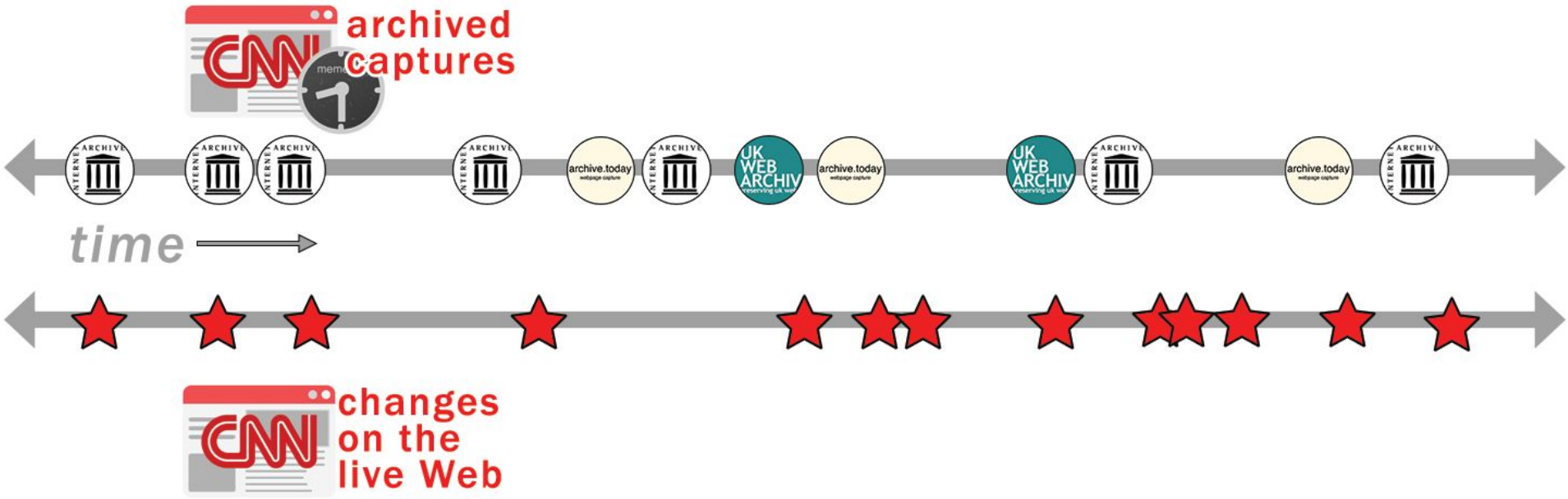
Desire: Include Personal Archives



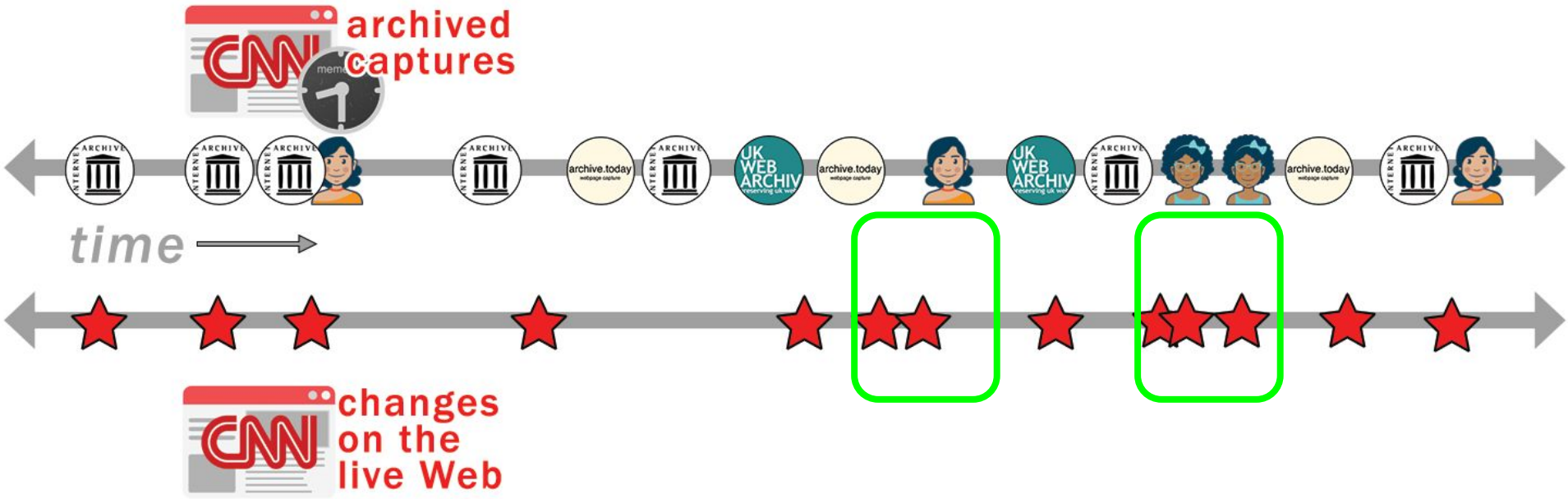
Desire: Include Other Non-Aggregated Archives



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

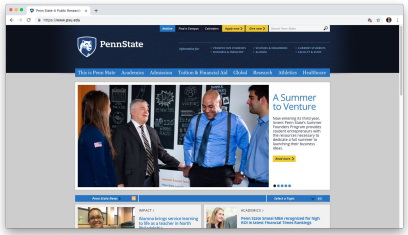
- Introduction/Motivation
- Background
- Preliminary Research
- Proposed Framework
- Evaluation Plan

Outline

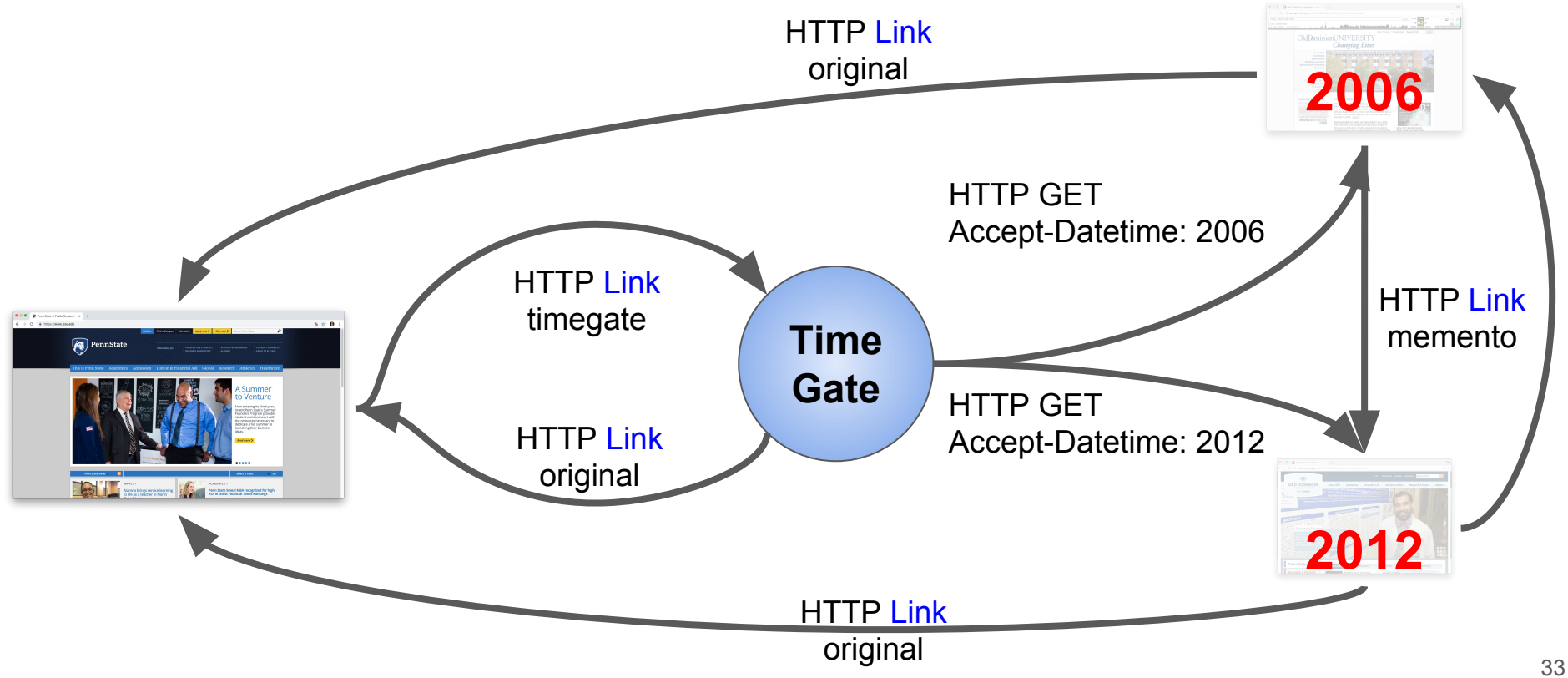
- Introduction/Motivation
- Background
- Preliminary Research
- Proposed Framework
- Evaluation Plan

Needed Association of Live-to-Archived Web

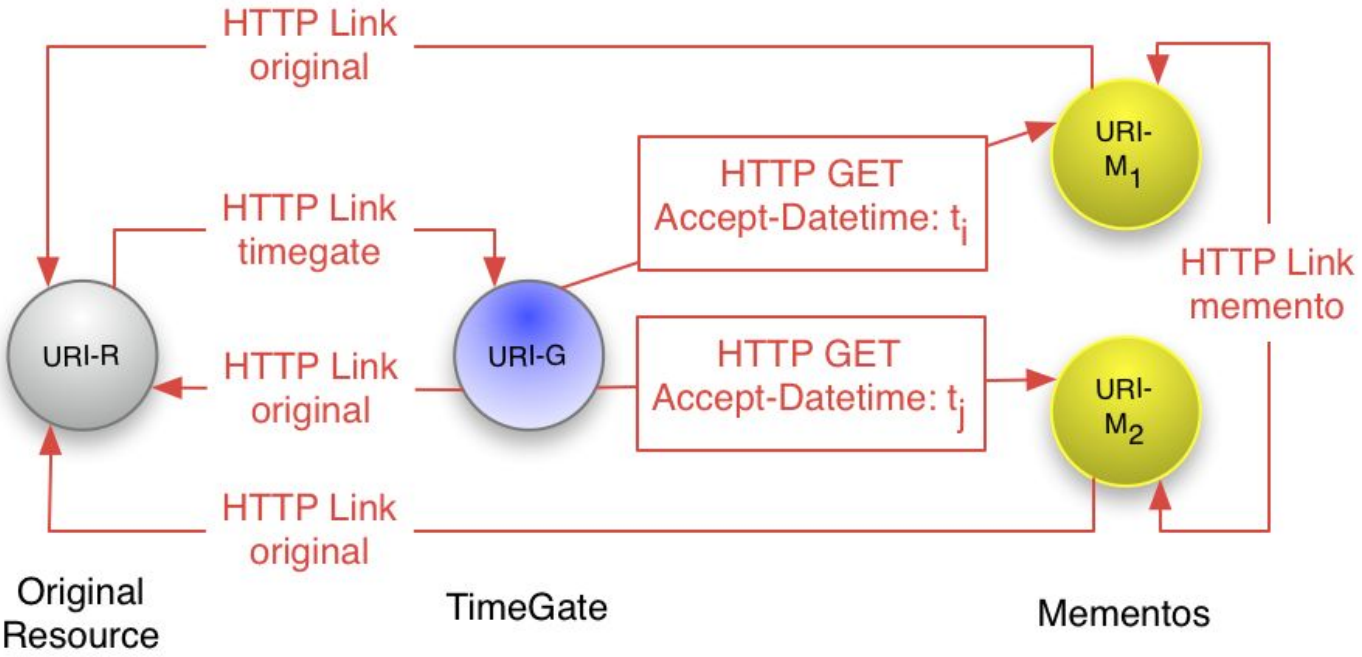
- Archived captures (mementos) are “aware” of what they represent from the live Web
- ...but we don't have a good way to establish the bidirectional linkage



Representations can be Linked in time



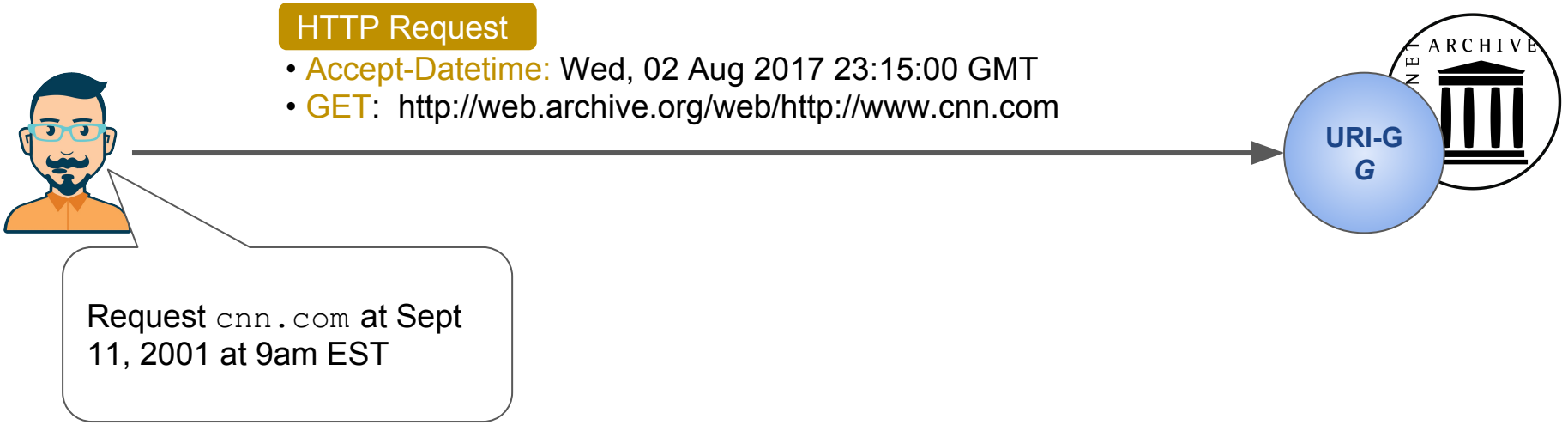
Background: Memento



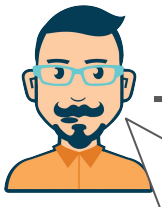
Memento Guide: Introduction. http://www.mementoweb.org/guide/quick-intro/, January 2015.

* 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



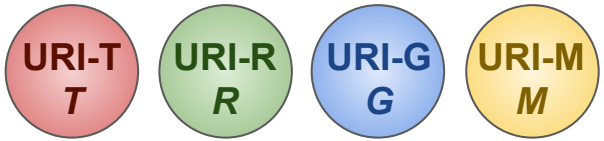
Request `cnn.com` at Sept 11, 2001 at 9am EST

HTTP Request

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

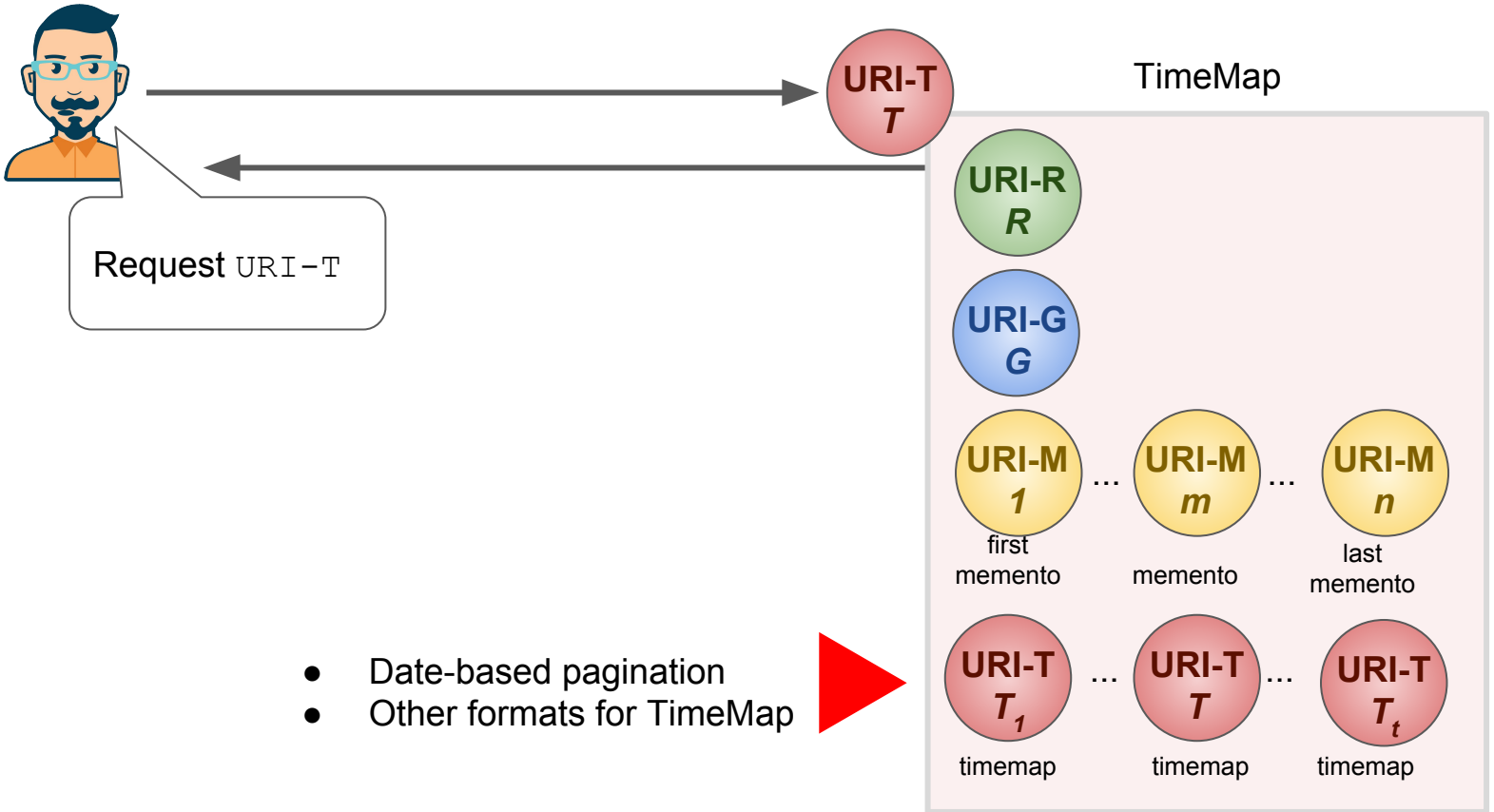
HTTP Response (302)

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



timemap original timegate memento

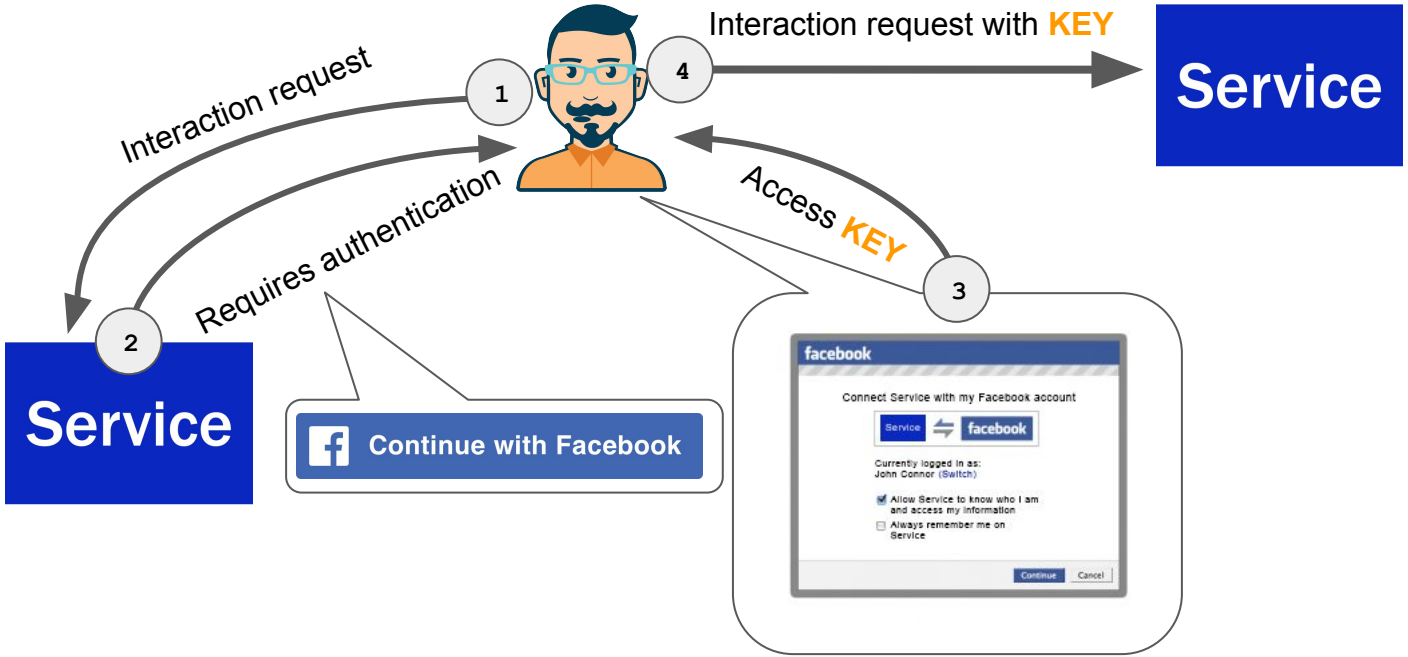
Background: Dereferencing a TimeMap at URI-T



- Date-based pagination
- Other formats for TimeMap

Role-based delegation and authentication

A familiar paradigm used for authentication on the live Web

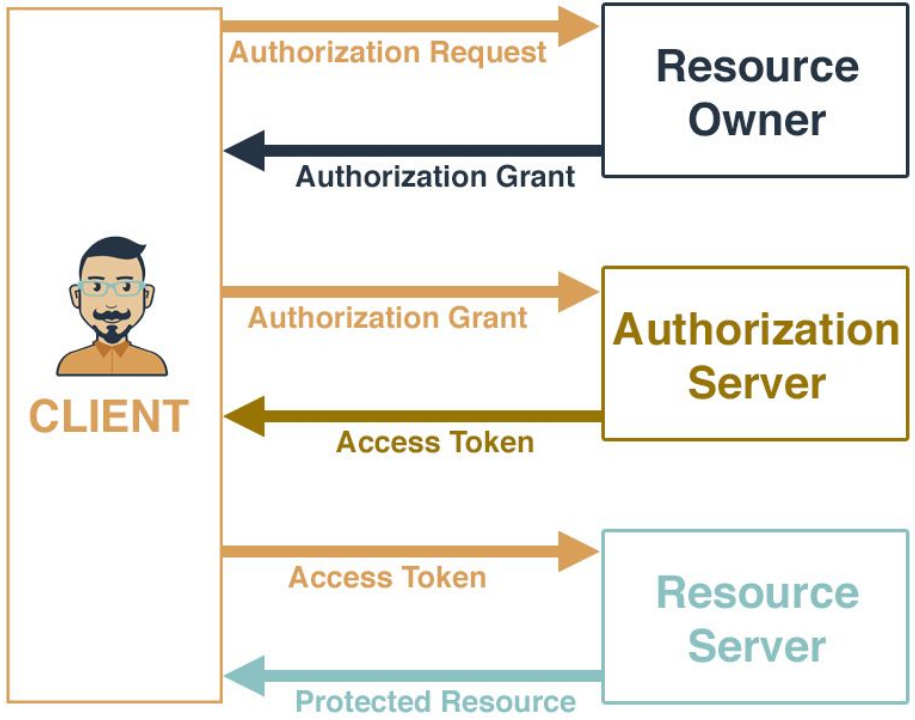


Background - Privacy and Security

- Web users question trusting institutions to preserve private Web contents[1]
- OAuth 2.0[2] 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?

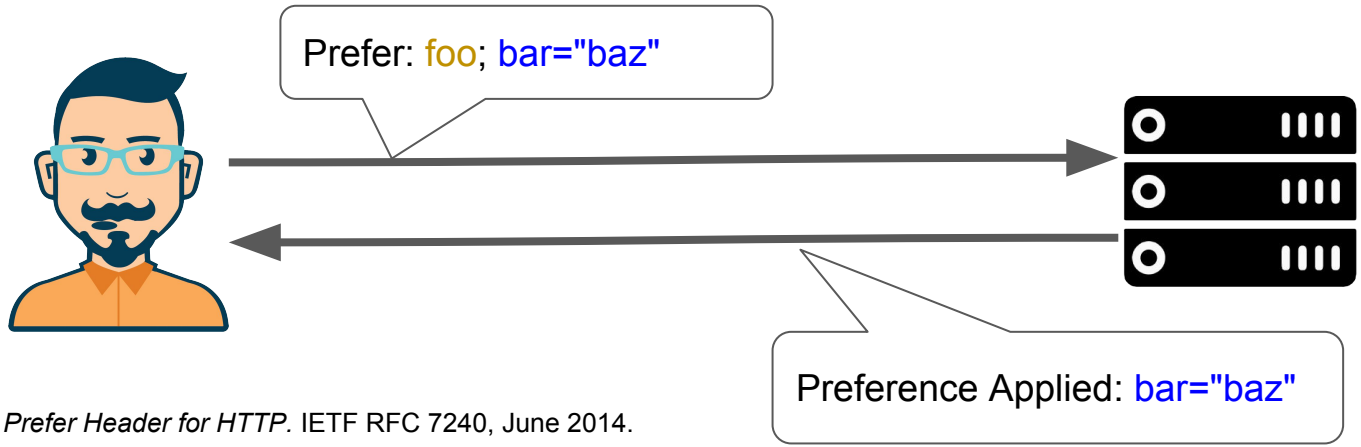


[1] Marshall and Shipman., "*On the Institutional Archiving of Social Media*", JCDL 2012

[2] D. Hardt. *The OAuth 2.0 Authorization Framework*. IETF RFC 6749, October 2012.

HTTP Prefer

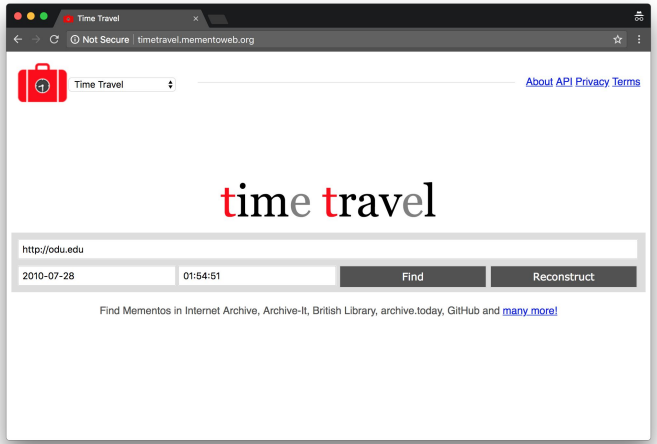
- HTTP negotiation already available via Accept-* headers
- *Prefer* syntax provide mechanism for client to specify preferences
  - ...with which servers may not comply

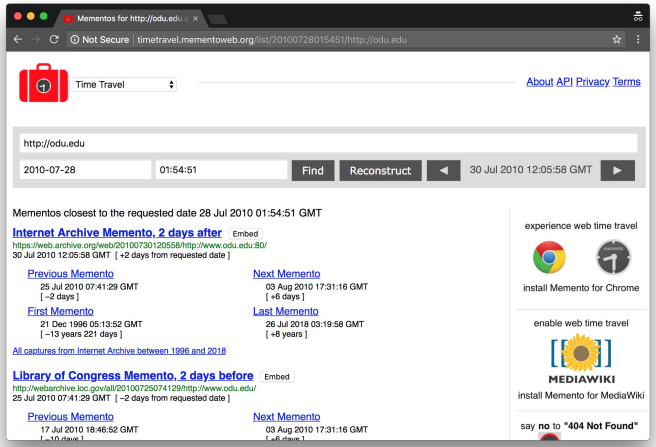


* J. Snell. *Prefer Header for HTTP.* IETF RFC 7240, June 2014.

Memento Aggregation State of the Art

Memento Aggregation - MementoWeb



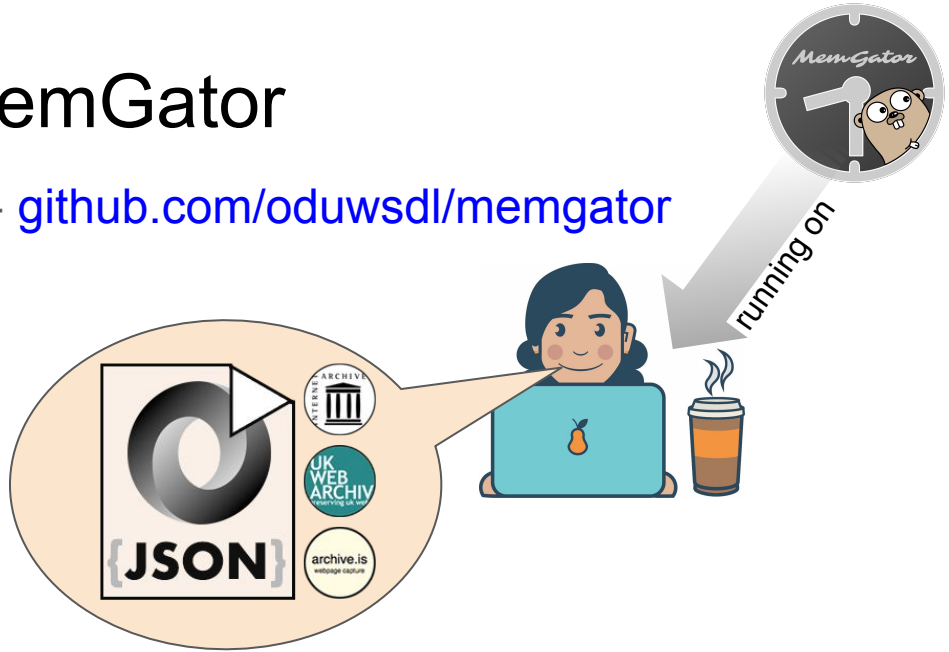


Also available via CLI:

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

Memento Aggregation - MemGator

- Open Source Memento Aggregator - github.com/oduwsdl/memgator
- 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"}
```

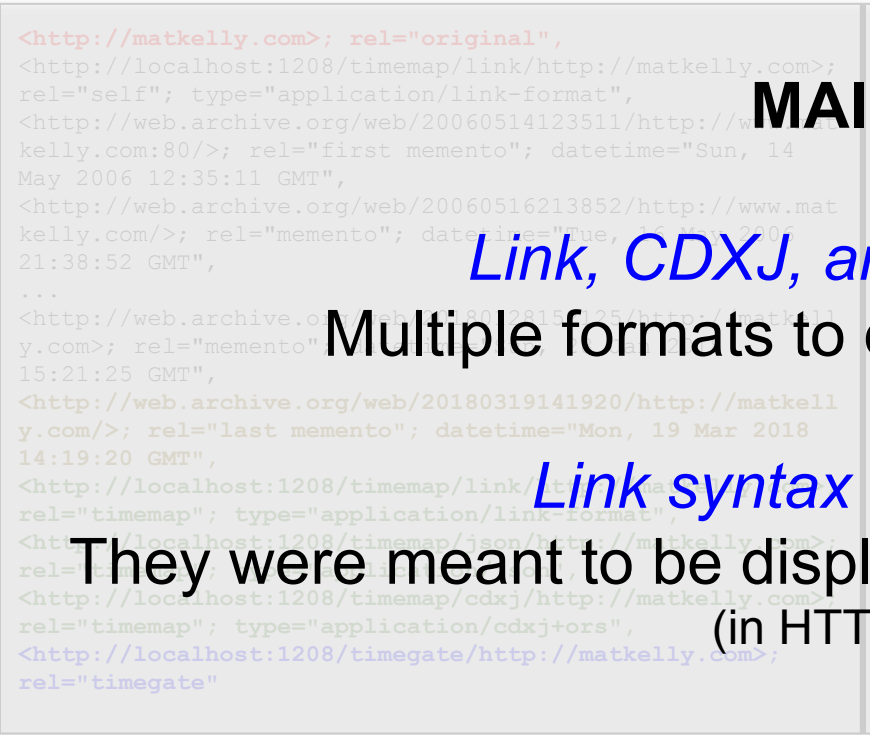
Link (RFC 7089) TimeMap

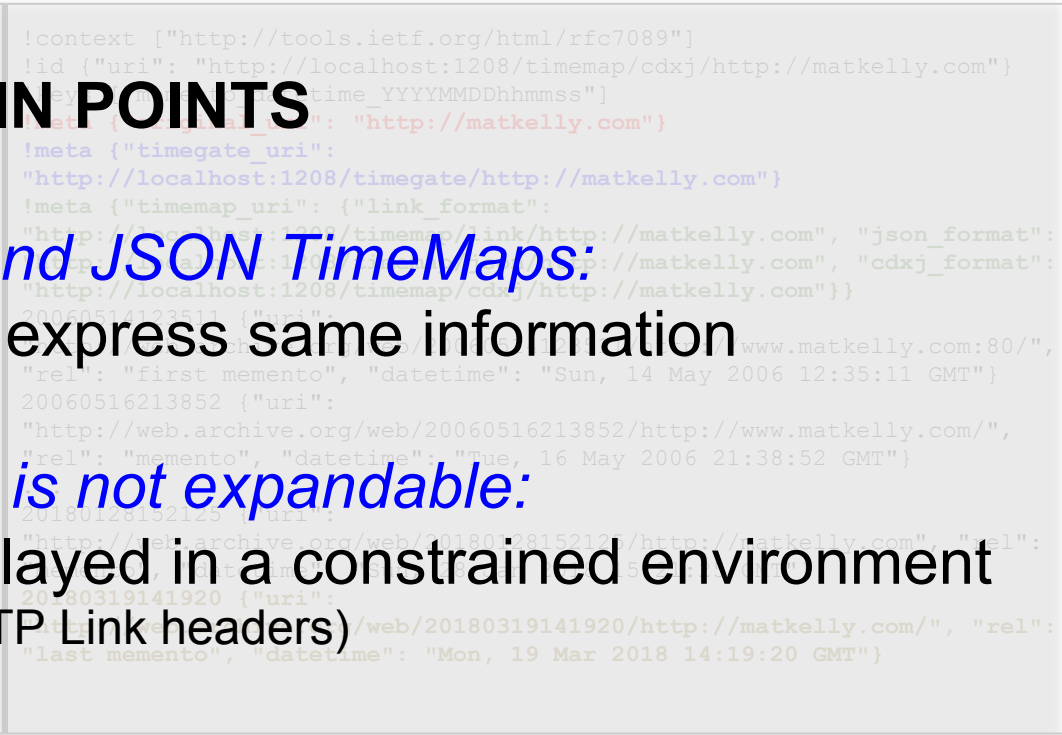
CDXJ TimeMap

Original URI (URI-R) | Other TimeMaps (URI-Ts) | TimeGate (URI-G) | Relative Relations

See Alam, "CDXJ: An Object Resource Stream Serialization Format", 2015

CDXJ: An Alternative TimeMap Format





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)

Link (RFC 7089) TimeMap

CDXJ TimeMap

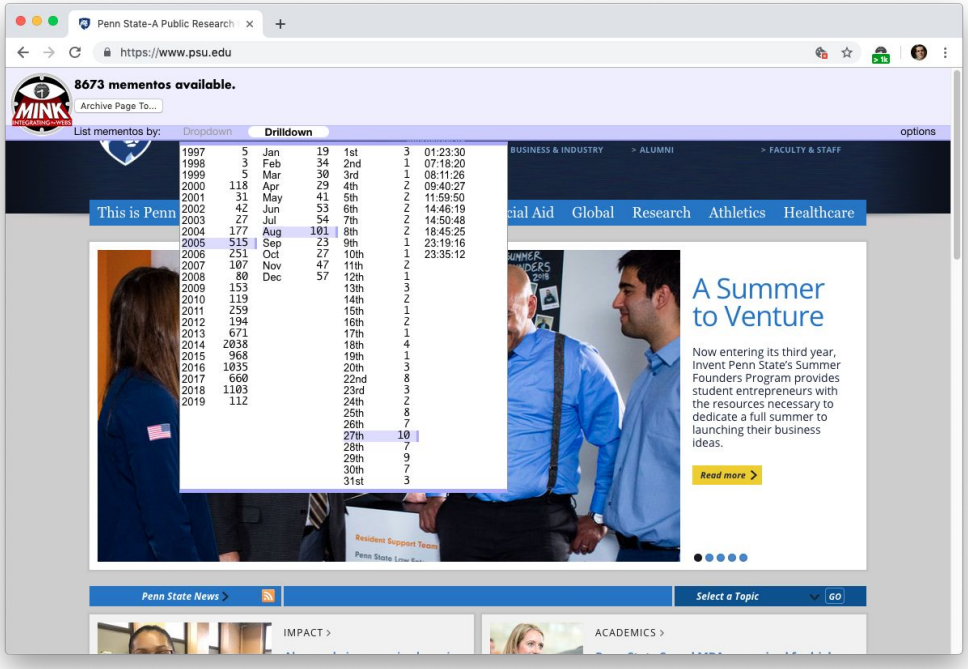
Original URI (URI-R) | Other TimeMaps (URI-Ts) | TimeGate (URI-G) | Relative Relations

See Alam, "CDXJ: An Object Resource Stream Serialization Format", 2015

: 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



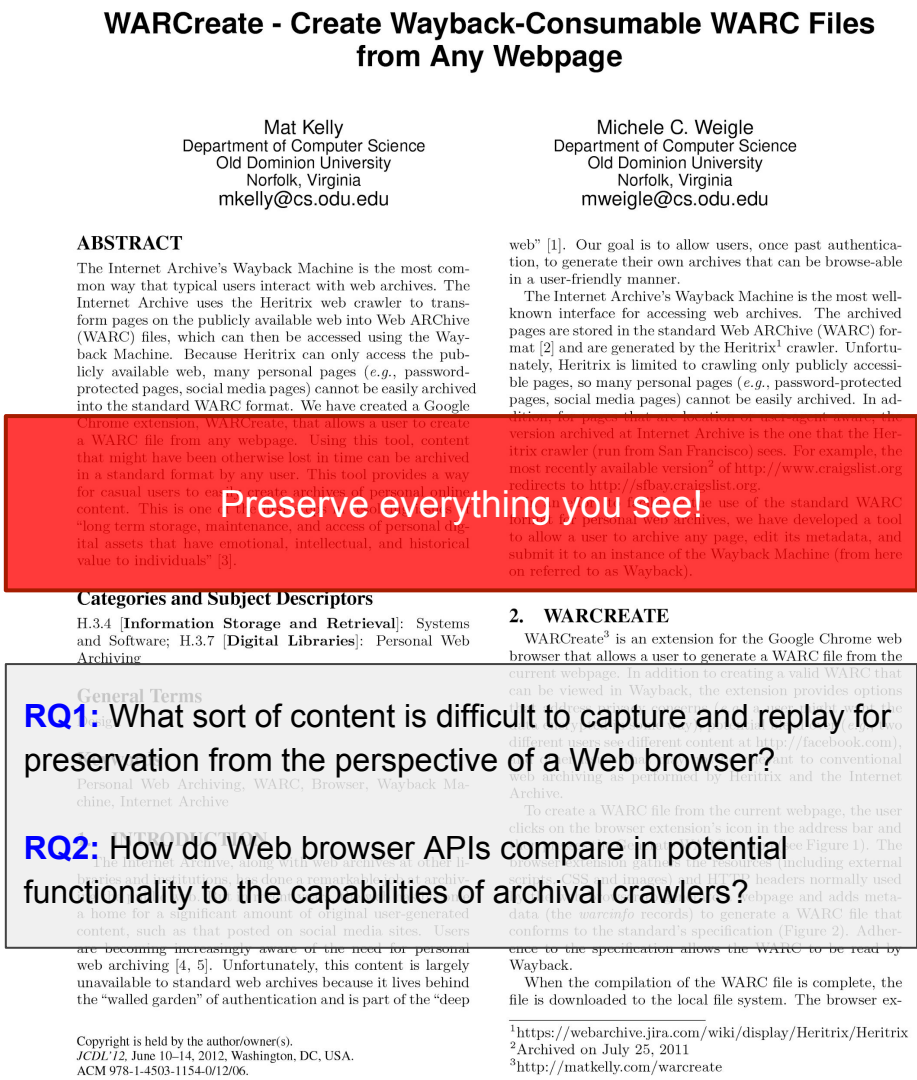
* Kelly et al., "Mink: Integrating the Live and Archived Web Viewing Experience Using Web Browsers and Memento", JCDL 2014

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...



WARCreate - Create Wayback-Consumable WARC Files from Any Webpage

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

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

ABSTRACT

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

Categories and Subject Descriptors

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

General Terms

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

1. INTRODUCTION

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

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

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

2. WARCREATE

WARCreate[3] is an extension for the Google Chrome web browser that allows a user to generate a WARC file from the current webpage. In addition to creating a valid WARC that can be viewed in Wayback, the extension provides options [illegible] different users (different content at http://facebook.com), [illegible] unt to conventional web archiving as performed by Heritrix and the Internet Archive.

To create a WARC file from the current webpage, the user clicks on the browser extension's icon in the address bar and [illegible] (see Figure 1). The [illegible] including external scripts, CSS and images) and HTTP headers normally used [illegible] webpage and adds metadata (the *warcinfo* records) to generate a WARC file that conforms to the standard's specification (Figure 2). Adherence to the specification allows the WARC to be read by Wayback.

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

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

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

Preserve everything you see!

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?

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

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

Keywords: Web Archiving, Digital Preservation

Which things are hard to preserve?

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

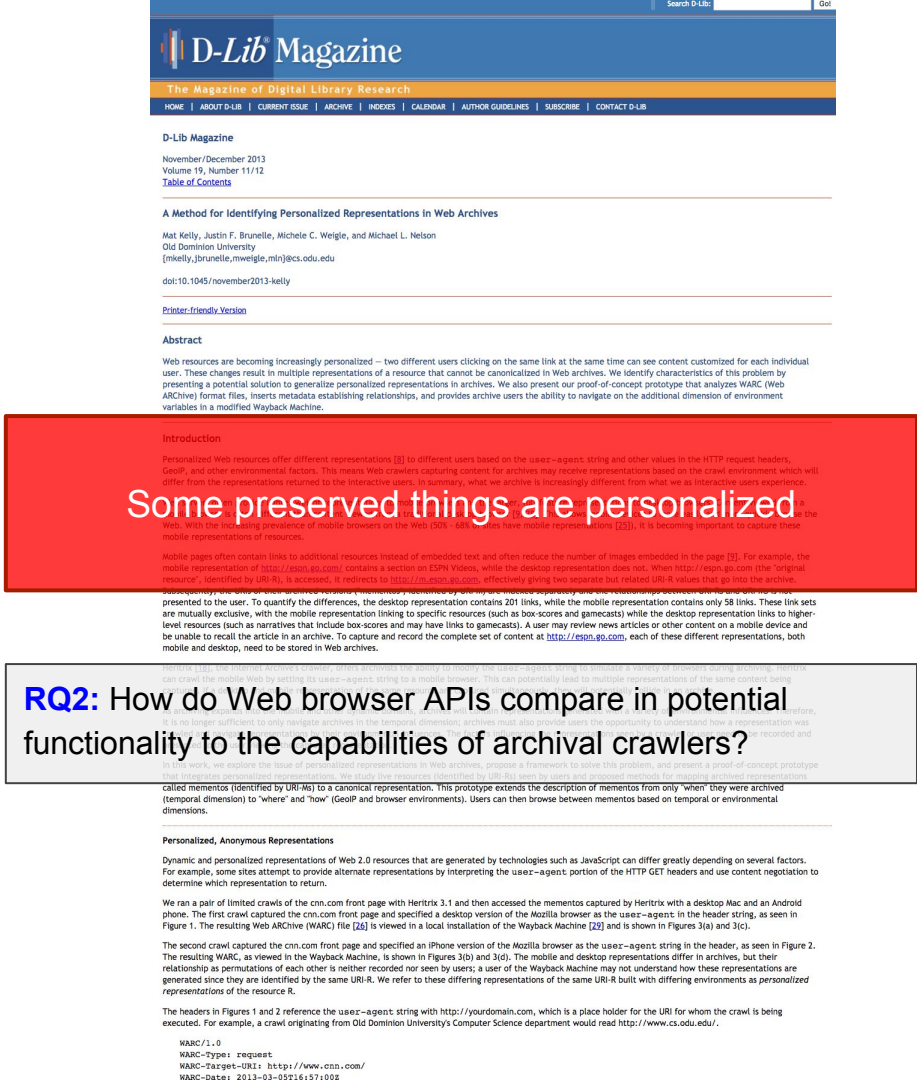
interactivity has become more fluid to the end-user. Early websites were static. Adoption of JavaScript allowed the components on a web page to respond to users' actions or be manipulated in ways that made the page more usable. Ajax [9] combines multiple web technologies to give web pages the ability 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

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...



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 Services: Web-based services] and Archives

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 called *mementos*. Several Memento clients exist, including [illegible] iOS and Android [2]. While Memento support for mobile contexts of the native browsers gives 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 [illegible] requires the user to either resort to the past with each page

web. We have developed a new browser extension, Mink[2], 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 [illegible]. Bombarding the Memento proxy-[illegible] requests for content negotiation is computationally expensive. We have implemented a method, [illegible][3] that utilizes the Memento TimeGate and TimeMap resources to a URI, 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 [illegible] queries a memento's TimeGate and expects a TimeMap in return. While processing the response, Mink displays a [illegible] 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 [illegible] is executed, a badge is set atop the extension icon indicating how many mementos [illegible] 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 compar-

[2]Named for Minkowski Space
[3]Available at `https://github.com/machawk1/mink`

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

Bridge live and archived Web
Provides a seamless viewing experience

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

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?

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 Descriptors

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

General Terms

Experimentation, Standardization, Verification

Keywords

Web Crawler, Web Archiving, Digital Preservation

1. INTRODUCTION

Web archiving is necessary for posterity. The goal of web archiving is to capture 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" [...] allowed each browser to render a web page and produce an evaluation of how well the browser conformed to the standards. In much the same way, we have created an "Archival Acid Test" to implement features of web browsers in a web page. While all standards-compliant browsers will correctly 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.

2. RELATED WORK

[...] variety of tools in a variety of formats. An ISO standard format utilized by institutional and personal web archivists alike is the Web ARChive (WARC) format [1]. WARC files allow HTTP communication that occurred during a crawl as well as payload, metadata and other archival features to be encoded in a single or an extensibly defined set of WARC files.

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

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

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

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

How well do archiving tools perform?

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

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

ABSTRACT

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 ...

... aged. 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 em- ...

Not all missing resources are created equal

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

1. INTRODUCTION

Web archives are valuable cultural repositories that capture and store Web content. Users make use of archives like the Internet Archive [16, 25] to retrieve archived material [11, 14] for a variety of purposes and in a variety of ways [3]. However, the resources being requested by Web users may not be complete; embedded resources are sometimes missing from an archived Web page [4]. Missing embedded resources return a non-200 HTTP status (e.g., 404, 503) when their ...

... 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 ...

... 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 ...

BEST STUDENT PAPER AWARD at JCDL 2014

... 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

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

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...

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

CrossMark

The impact of JavaScript on archivability

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

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

Abstract As web technologies evolve, web archivists work to adapt so that digital history is preserved. Recent advances in web technologies have introduced client-side executed scripts (Ajax) that, for example, load data without a change in top level Universal Resource Identifier (URI) or require user interaction (e.g., content loading via Ajax when the [illegible] ing method for capturing web pages more difficult. 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 [illegible], and the Heritrix crawler. We found that only 4.2 % of the Twitter collection is perfectly archived by all of the tools, while 34.3 % of the Archive-It collection is perfectly archived. After studying the quality of these mementos, [illegible] (Ajax) as the source of archival difficulty. Further, we show that resources are increasing their use of JavaScript to load embedded resources. By 2012, over half (54.5 %) of pages use JavaScript to load embedded resources. The number of embedded resources loaded via JavaScript has increased by 12.0 % from 2005 to 2012. We also show that JavaScript is responsible for 33.2 % more missing resources in 2012 than in 2005. This shows that JavaScript is responsible for an increasing proportion of the embedded resources unsuccessfully loaded by mementos. JavaScript is also responsible for 52.7 % of all missing embedded resources in our study.

Keywords Web architecture · Web archiving · Digital preservation

Missing JavaScript has big ramifications

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

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 [40,47] and commercial and government compli-

The web has gone through a gradient of changes fueled by [illegible]. Early websites were relatively static, while continued adoption of web technologies has made the pages personalized and more interactive. 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-

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: mkelly@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...

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

CrossMark

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

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

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

Abstract 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 to archived resources is a measure of their impact on the Web page. Some embedded resources are more important than others. In this paper, we propose a method to assign a damage rating to missing 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 an overrating 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 missing over time. Finally, we investigate the impact of JavaScript on archives, showing that a crawler that can execute JavaScript-dependent representations will reduce memento damage by 13.5 %.

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

Is the metric for missing resources applicable across Web?

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

1 Introduction

17 % and an improvement by 51 % if the mementos have a damage in the Internet Archive, showing that it is getting better at mitigating damage over time (going from a damage

and store Web content. People (and robots) use archives like archived material [20, 25] for a variety of purposes and in a variety of ways [2]. However, the resources being requested by Web users may not be 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
mkelly@cs.odu.edu

Hany SalahEldeen
hany@cs.odu.edu

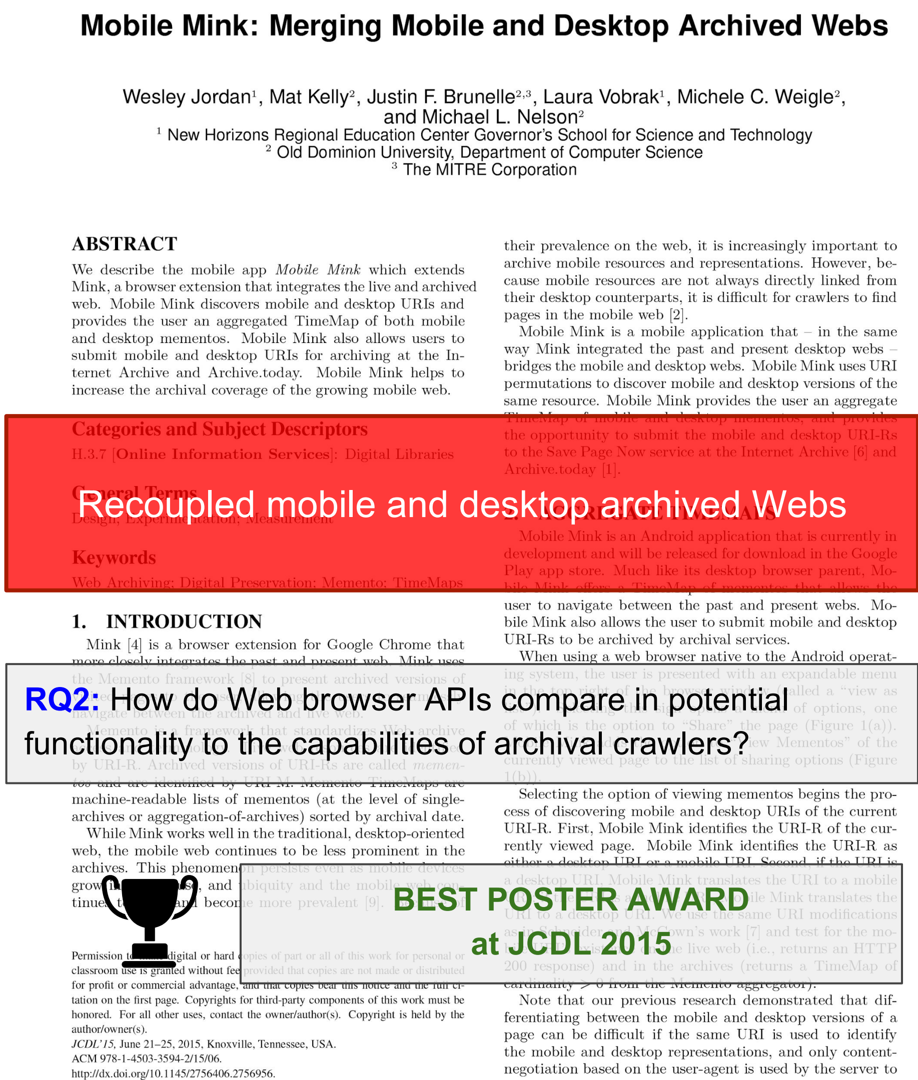
Michele C. Weigle
mweigle@cs.odu.edu

Michael L. Nelson
mln@cs.odu.edu

[1] 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...



Mobile Mink: Merging Mobile and Desktop Archived Webs

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

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

Recoupled mobile and desktop archived Webs

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

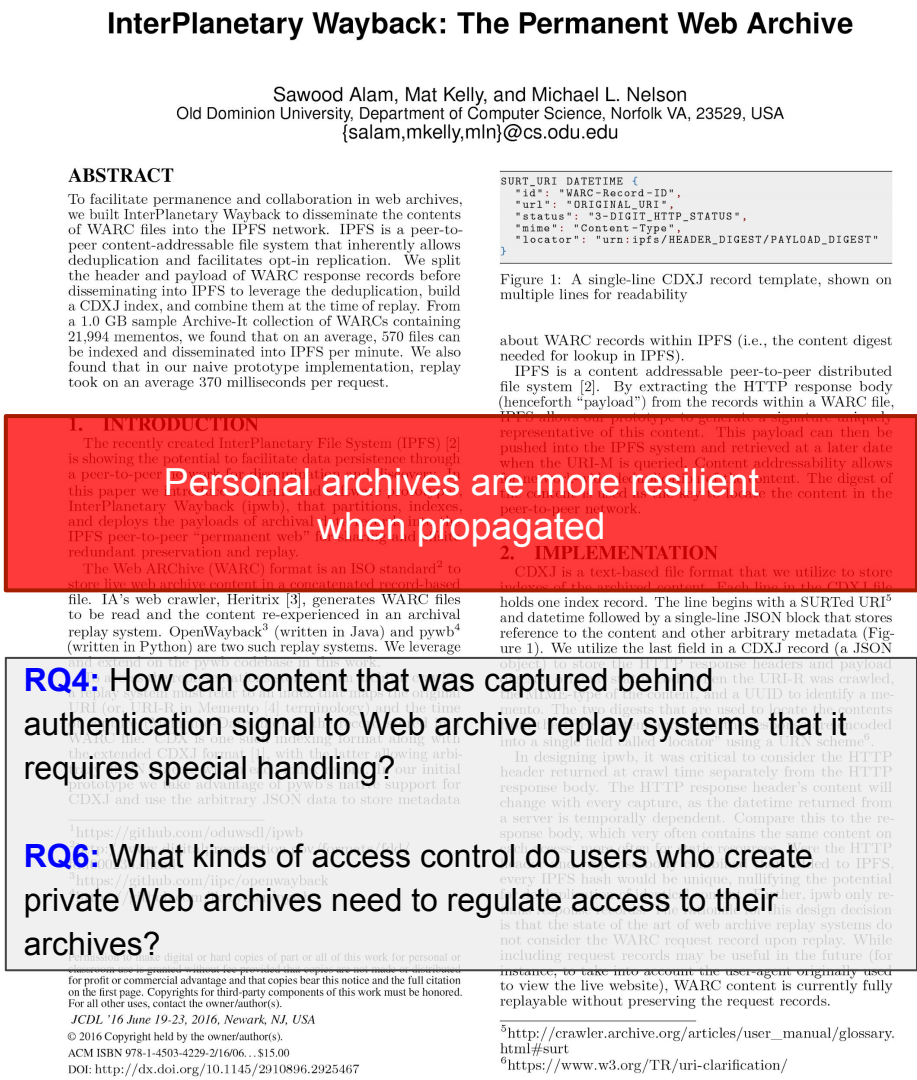
BEST POSTER AWARD at JCDL 2015

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

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...



InterPlanetary Wayback: The Permanent Web Archive

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 [illegible] this paper we introduce InterPlanetary Wayback (ipwb), that partitions, indexes, and deploys the payloads of archival [illegible] IPFS peer-to-peer "permanent web" [illegible] redundant preservation and replay.

The Web ARChive (WARC) format is an ISO standard[2] to store live web archive content in a concatenated record-based file. IA's web crawler, Heritrix [3], generates WARC files to be read and the content re-experienced in an archival replay system. OpenWayback[3] (written in Java) and pywb[4] (written in Python) are two such replay systems. We leverage [illegible] on the pywb codebase in this [illegible] URI (or URI-R in Memento [4] terminology) and the time [illegible] the extended CDXJ format [3], with the latter allowing arbi- [illegible] our initial prototype we take advantage of pywb's native support for CDXJ and use the arbitrary JSON data to store metadata

[1]https://github.com/oduwsdl/ipwb
[illegible]https://github.com/ipfc/openwayback

[illegible] digital or hard copies of part or all of this work for personal or [illegible] 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 '16 June 19-23, 2016, Newark, NJ, USA

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_URI",
  "status": "3-DIGIT_HTTP_STATUS",
  "mime": "Content-Type",
  "locator": "urn:ipfs/HEADER_DIGEST/PAYLOAD_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 [illegible] representative of this content. This payload can then be pushed into the IPFS system and retrieved at a later date when the URI-M is queried. Content-addressability allows [illegible] content. The digest of [illegible] the content in the peer-to-peer network.

2. IMPLEMENTATION

CDXJ is a text-based file format that we utilize to store [illegible] holds one index record. The line begins with a SURTed URI[5] 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 [illegible] object, to store the HTTP response headers and payload [illegible] in the URI-R was crawled, [illegible] a UUID to identify a memento. The two digests that are used to locate the contents [illegible] into a single field called "locator" using a URN scheme[6].

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 [illegible] the HTTP [illegible] to IPFS, every IPFS hash would be unique, nullifying the potential [illegible] ipwb only re- [illegible] this 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.

[5]http://crawler.archive.org/articles/user_manual/glossary.html#surt
[6]https://www.w3.org/TR/uri-clarification/

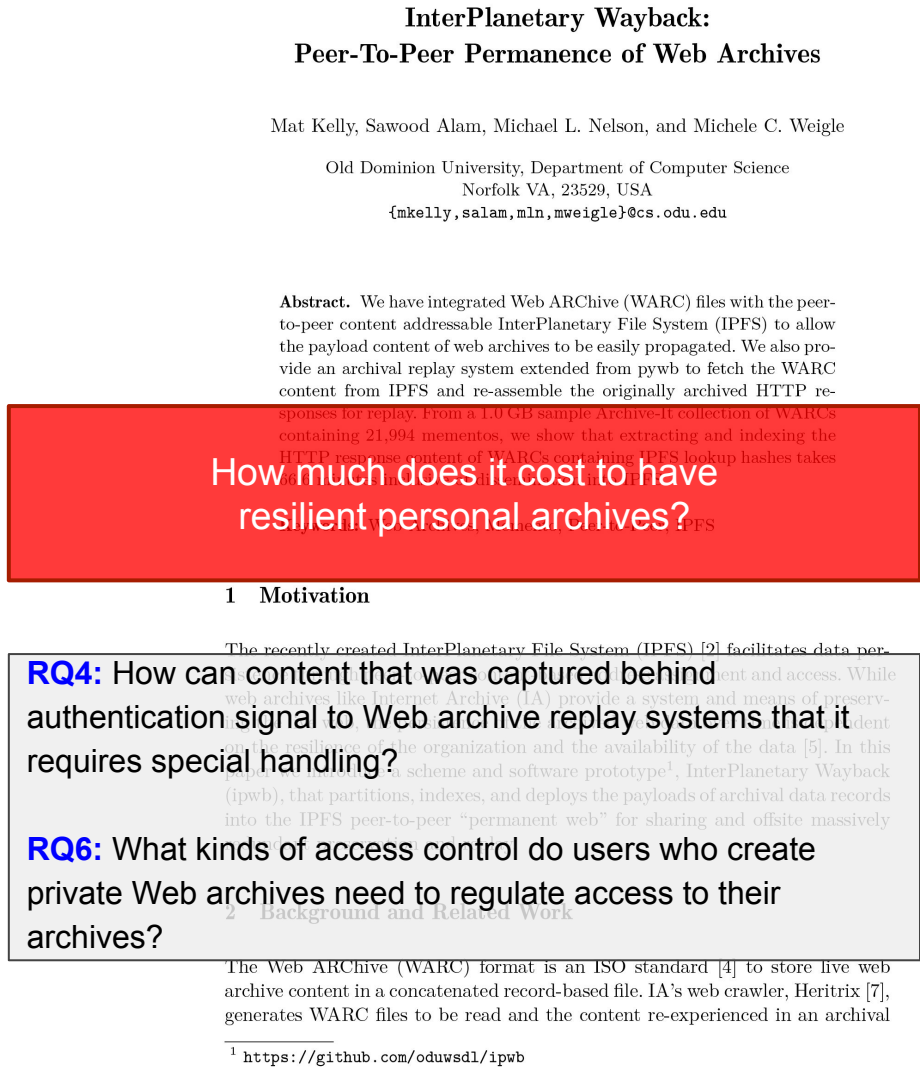
Personal archives are more resilient when propagated

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?

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...



InterPlanetary Wayback:
Peer-To-Peer Permanence of Web Archives

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

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?

1 Motivation

The recently created InterPlanetary File System (IPFS) [2] facilitates data per-

web archives like Internet Archive (IA) provide a system and means of preserv-

on the resilience of the organization and the availability of the data [5]. In this

(ipwb), that partitions, indexes, and deploys the payloads of archival data records into the IPFS peer-to-peer "permanent web" for sharing and offsite massively

2 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

[1] https://github.com/oduwsdl/ipwb

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?

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...

WAIL: Collection-Based Personal Web Archiving

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

ABSTRACT

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

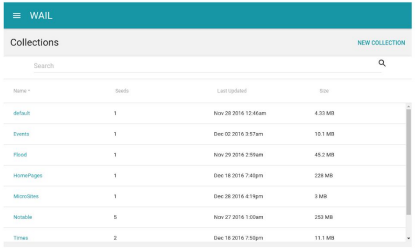


Figure 1: Collections screen

KEYWORDS

Personal Web Archiving

ACM Reference format:

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

Archive from the desktop
With higher fidelity than institutions

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

1 INTRODUCTION

Subscription-based Web archiving services like Archive-it allow users with limited technical knowledge to create and replay personalized collections with a [illegible] archival crawl. Similar to Archive-It is Webrecorder[1], 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 skill level. Even if a user is able to successfully set up these tools, [illegible] Heritrix and come up with their own means of associating the Web archives to each other for [illegible] access 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-Election with the software taking care of the details in managing the collections, crawls, and replay. We have integrated a native Chromium[2] browser (the core of Google's Chrome Web browser) into the archival process in order to surface content specific to sites like Twitter for more accurate and comprehensive preservation.

2 WAIL

[illegible] 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

[1]https://webrecorder.io/

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

JCDL'17, Toronto, Ontario, Canada

DOI: 10.XXX/XXXX

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 Internet Archive reduces 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 — Archival storage; 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 Using ServiceWorker. In *Proceedings of JCDL '17, Toronto, Ontario, Canada, June 19–23, 2017 (JCDL'17),* 4 pages.

Figure 1: Live Ad Zombie Leaks into Archived Page

Replay the archived Web using modern URI resolution

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 with all the embedded resources that are necessary to render the page correctly. Web archival replay systems rewrite embedded URI references to point to their archival versions (e.g., a reference to `external.example.net/logo.png` is changed to `archive.example.com/external.example.net/logo.png`). The browser can 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 16th, 2008, but when [illegible] an ad from the [illegible] look at the 2012 presidential candidates [5]. Client-side rerouting also saves [illegible]ting of the content on the client side, such as to include archival banners, hence, there is no need to send extra data with each HTML response. Client-side solutions such as Memento for Chrome[1] 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.

[1] 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...

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

Impact of URI Canonicalization on Memento Count

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

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

ABSTRACT

Memento TimeMaps [5] list identifiers for archival web captures (URI-Ms). When some URI-Ms are dereferenced, they redirect to a different URI-M instead of a unique representation at the datetime. This suggests that confidently obtaining an accurate count quantifying the number of non-forwarding captures for an Original Resource URI (URI-R) is not possible using a TimeMap alone and that the magnitude of a TimeMap is not equivalent to the number of representations it identifies. This work represents an abbreviated version of the full technical report describing this phenomena in depth [3]. For google.com we found that 84.9% of the URI-Ms in a TimeMap result in an HTTP redirect when dereferenced. The full study applies this technique to seven other URI-Rs of large Web sites and 13 academic institutions. Using a [illegible] 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 dereferenced.

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

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 [illegible] redirecting mementos.

URI coalescence considered harmful for archives

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

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 archived for a URI-R. However, TimeMaps may include URI-Ms for archived representations [illegible] URI-M is 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 cannot be completed without [illegible]

Redirection in the web archive can be attributed to a variety of canonicalization rules [3]. Preserving and replaying these redirections 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 (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 [illegible] 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.

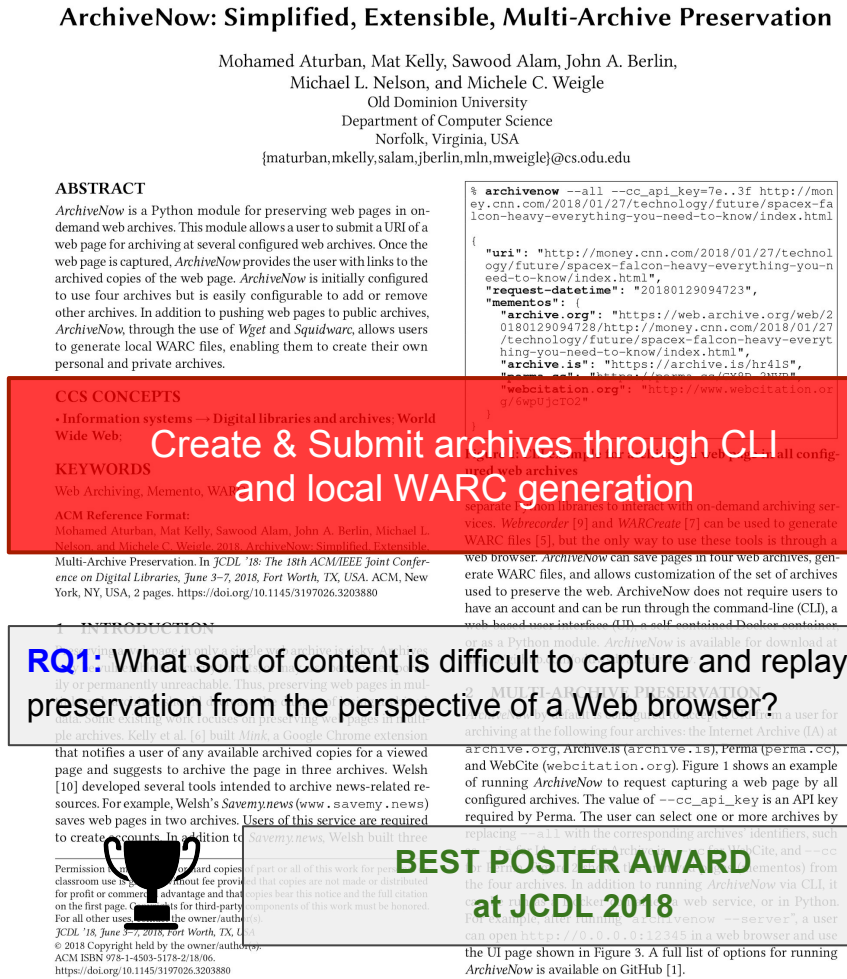
2 BACKGROUND AND RELATED WORK

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 [illegible] critical for retaining a cohesive, comprehensive listing of the mementos available [illegible] memento redirection patterns relating to HTTP redirects to supply the user with the correct memento when a redirect is encountered in the archives. They introduced the notion of "URI stability" to give a quantitative measure of the presence of HTTP 3XX status codes that result when URI-Ms in TimeMaps are dereferenced.

BEST POSTER AWARD at JCDL 2017

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.

```
% 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/hr4lS",
    "perma.cc": "http://perma.cc/CY8D-2WVB",
    "webcitation.org": "http://www.webcitation.org/6wpUjcTO2"
  }
}
```

Create & Submit archives through CLI and local WARC generation

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

BEST POSTER AWARD at JCDL 2018

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

Preliminary Research

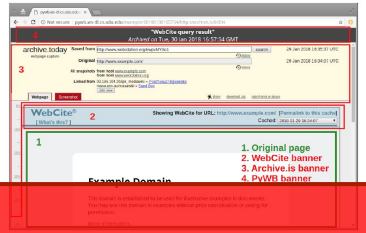
1. JCDL 2012 - WARCreate
2. TPDL 2013 - Change in Archivability
3. DLib 2013 - Method for Identifying
4. JCDL 2014 - Mink
5. JCDL 2014 - Archival Acid Test
6. JCDL 2014 - Not All Mementos
7. IJDL 2015 - Impact of JavaScript
8. IJDL 2015 - Not All Mementos
9. JCDL 2015 - Mobile Mink
10. JCDL 2016 - InterPlanetary 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...

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 [illegible]

ACM Reference Format:

Sawood Alam, Mat Kelly, Michele C. Weigle, [illegible]
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 [illegible] 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

JCDL '18, June 3–7, 2018, Fort Worth, TX, USA

ACM ISBN 978-1-4503-5178-2/18/06.
https://doi.org/10.1145/3197026.3203881

2 METHODOLOGY

There are 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 [illegible] and *Archive.is* use this method. *OpenWayback*, a commonly used [illegible] approach, [illegible] 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.

[illegible] banners. This can be implemented by 1) serving both the banner and memento inside an *iframe*. For example, *WebCite* uses the first approach, while many archives, such as the *Portuguese Web Archive*, 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

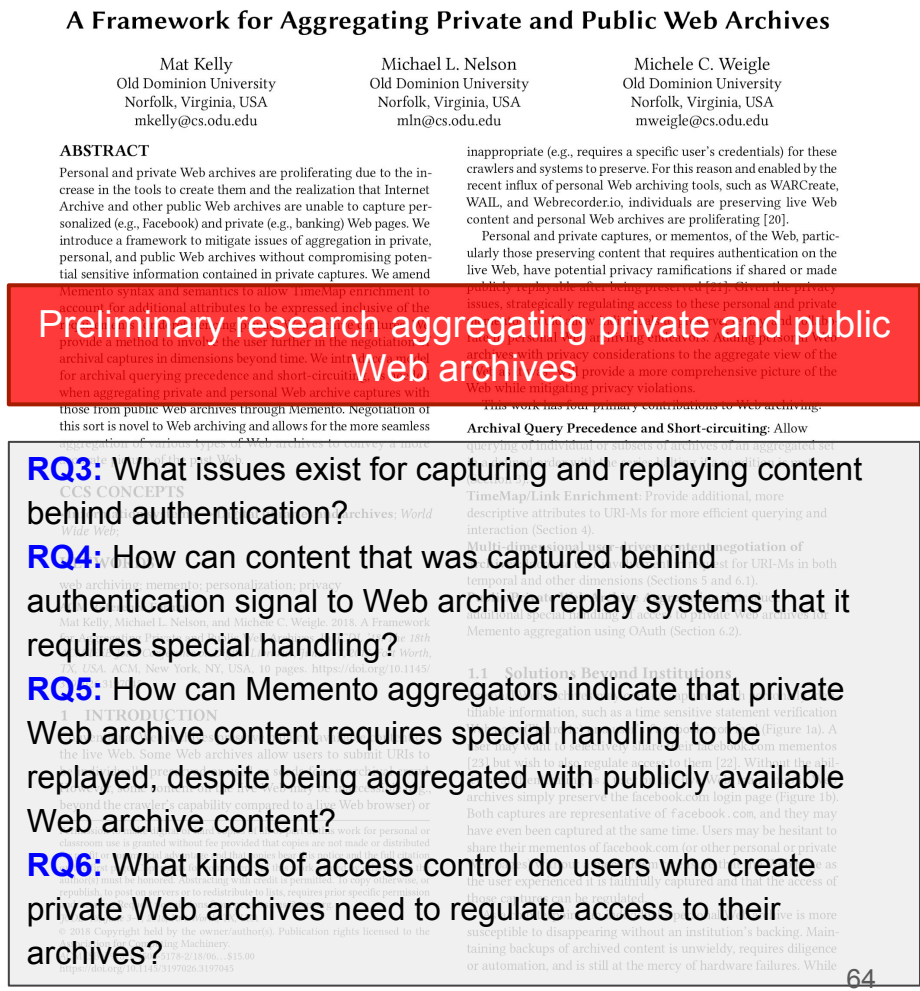
RQ1: What sort of content is difficult to capture and replay for preservation from the perspective of a Web browser?

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

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

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

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

ABSTRACT

Personal and private Web archives are proliferating due to the increase in the tools to create them and the realization that Internet Archive and other public Web archives are unable to capture personalized (e.g., Facebook) and private (e.g., banking) Web pages. We introduce a framework to mitigate issues of aggregation in private, personal, and public Web archives without compromising potential sensitive information contained in private captures. We amend Memento syntax and semantics to allow TimeMap enrichment to account for additional attributes to be expressed inclusive of the provide a method to involve the user further in the negotiation archival captures in dimensions beyond time. We introduce for archival querying precedence and short-circuiting 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 WARCreate, WAIL, and Webrecorder.io, individuals are preserving live Web content and personal Web archives are proliferating [20].

Personal and private captures, or mementos, of the Web, particularly those preserving content that requires authentication on the live Web, have potential privacy ramifications if shared or made

Preliminary research aggregating private and public Web archives

Archival Query Precedence and Short-circuiting: Allow

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

Memento aggregation using OAuth (Section 6.2).

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

Proposed Framework

(for aggregating private and public Web archives)

Proposed Framework

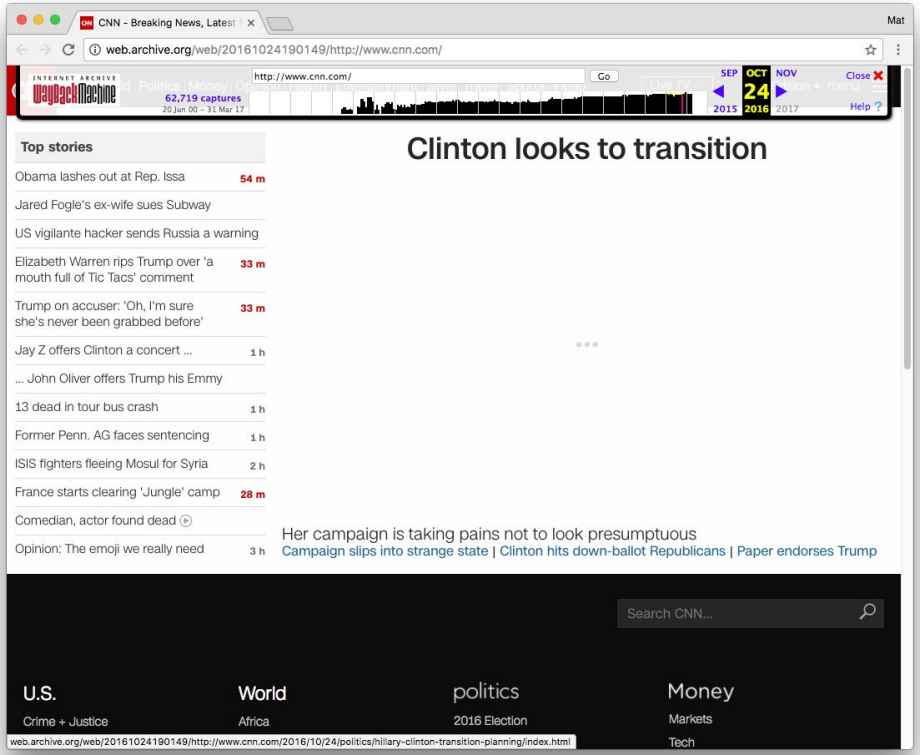
- Archival negotiation beyond time
- Query precedence & short-circuiting
- Mementities

PROPOSED FRAMEWORK

Archival Negotiation Beyond Time

More Expressive TimeMaps

- Memento Quality (e.g., Damage)[1]
- How Many Captures?[2]
- How Many Are Identical?[2,3]
- Other Attributes of Mementos...



[1] Brunelle *et al.*, JCDL 2014, IJDL 2015
[2] Kelly *et al.*, JCDL 2017
[3] AlSum and Nelson, ECIR 2014

Additional TimeMap Attributes

Content-based Attributes

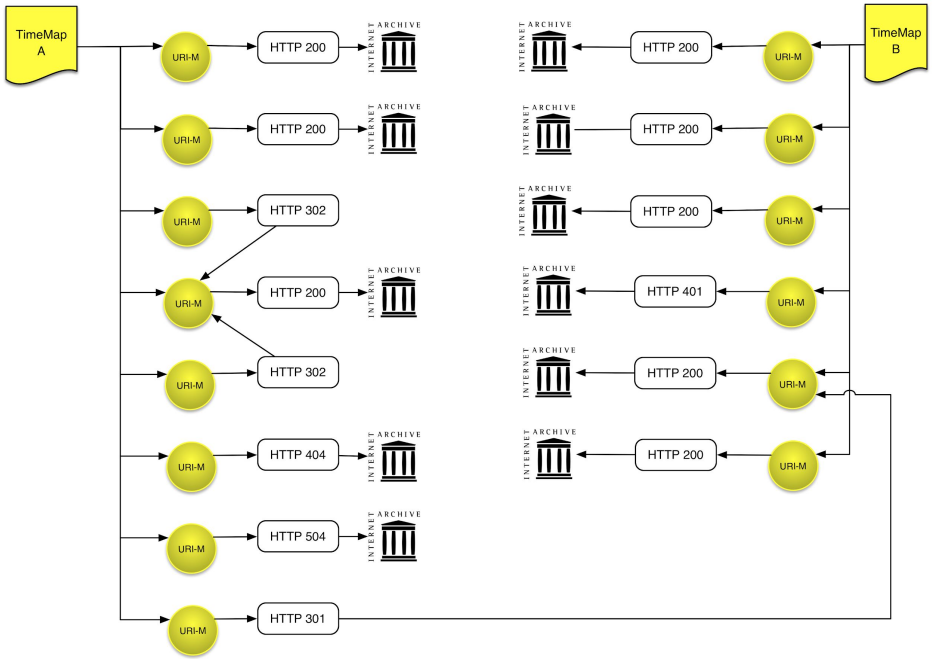
Derived Attributes

Access Attributes

TimeMap Enrichment: Content-Based Attributes

- Status Code[1]
- Content-Digest
  - In WARC & CDX
  - Not all archives expose CDX

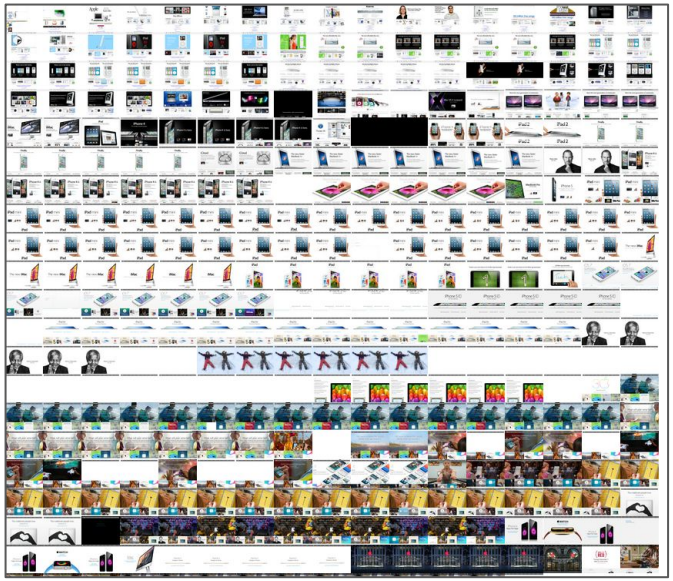
- Would allow more info about mementos without requiring comprehensive dereferencing



[1] Kelly *et al.,* "Impact of URI Canonicalization on Memento Count", JCDL 2017, arXiv 1703.03302

TimeMap Enrichment: Derived Attributes

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



apple.com, many duplicate mementos!

[1] AlSum and Nelson, Thumbnail Summarization Techniques for Web Archives, ECIR 2014, pp. 299-310.
[2] Brunelle *et al.,* "The Impact of JavaScript on Archivability," IJDL, 17(2), pp. 95-117. January 2016.

TimeMap Enrichment: Access Attributes

How to distinguish

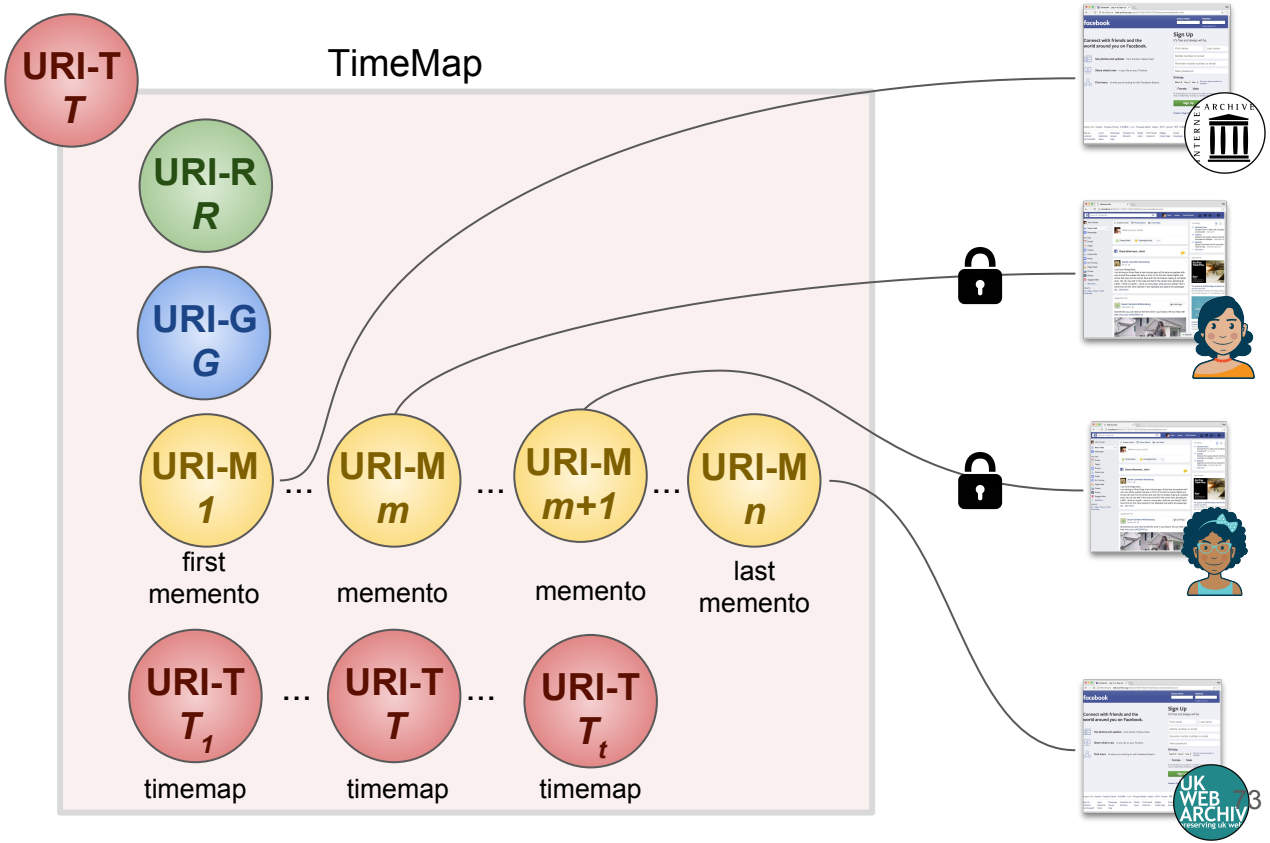
Private captures

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:LK26DRRQJ4WATC6LBVF3B3Z4P2CP5ZZ7",
  "damage": 0.24,
  "simhash": "65511106224221534 88",
  "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

1

2

3

3

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

See Atkins', "Paywalls in the Internet Archive", March 2018

Query Precedence

- More control of querying in series and parallel

1

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

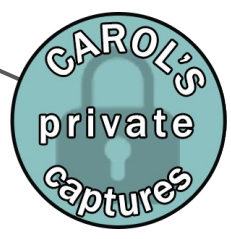
See Atkins', "Paywalls in the Internet Archive", March 2018

Query Precedence

- More control of querying in series and parallel

1

2



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

See Atkins', "Paywalls in the Internet Archive", March 2018

Query Precedence

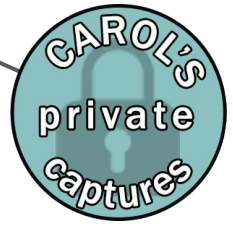
- More control of querying in series and parallel

1

2

3

3



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

See Atkins', "Paywalls in the Internet Archive", March 2018

Query Short-Circuiting

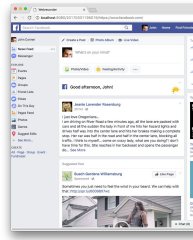
- **May give priority to archive relevancy.**
- Series halt when threshold met.

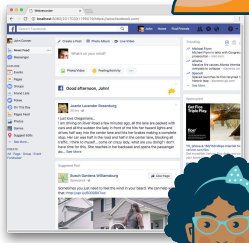
1

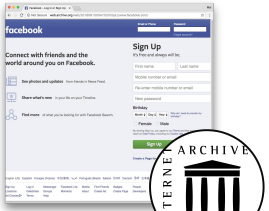
1

2

2



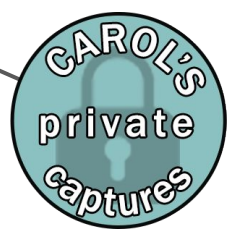




"Check private archives first. **Iff** you find no captures, only *then* check the public archives.

Query Short-Circuiting

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



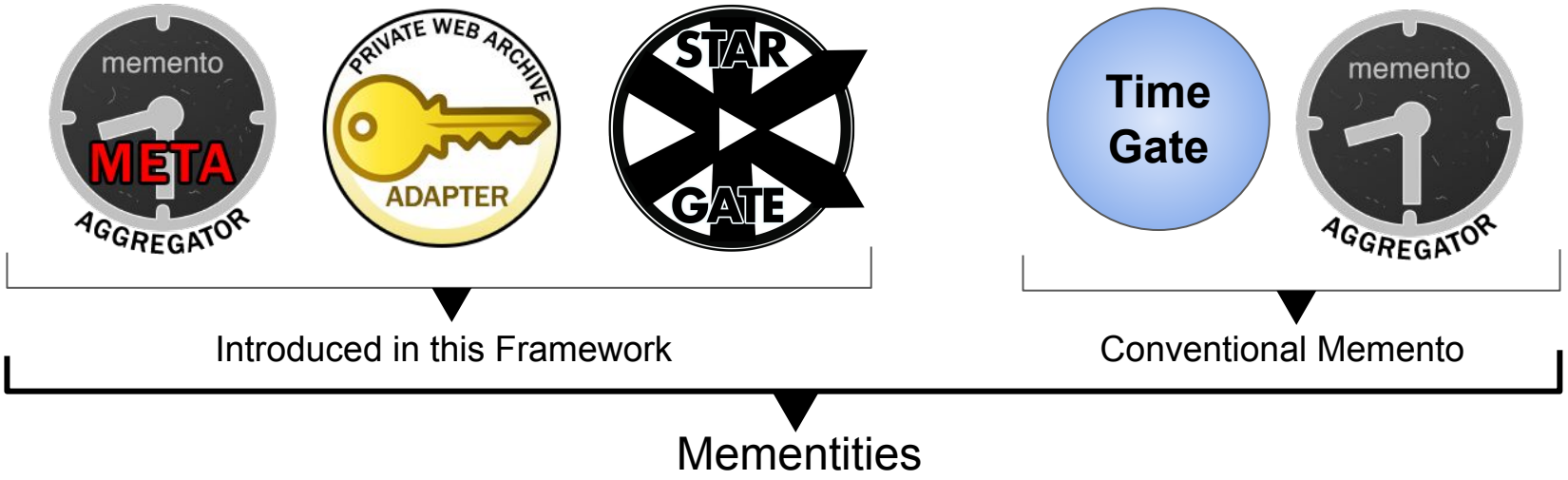
"Check private archives first. **Iff** you find no captures, only *then* check the public archives.

PROPOSED FRAMEWORK

Mementies

Mementities

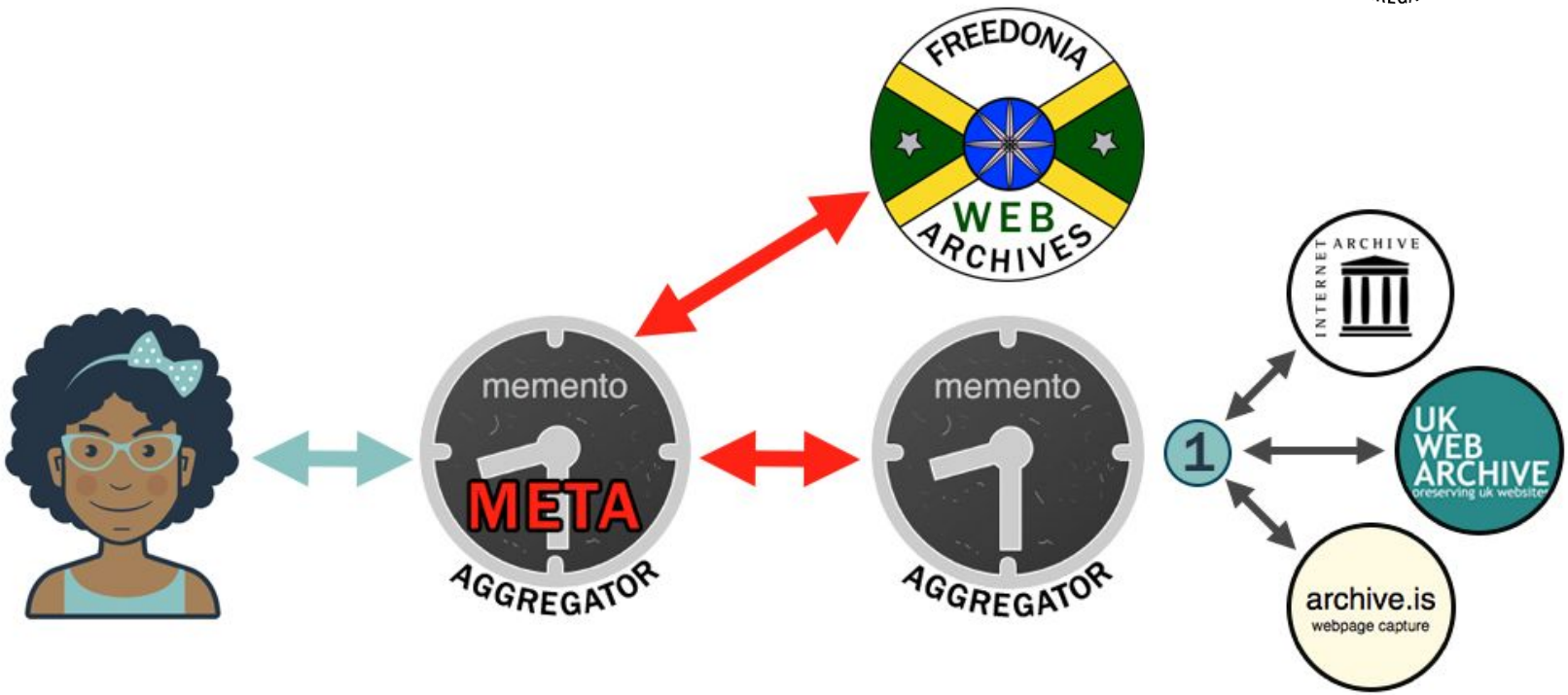
- Memento + Entity (*entity* term already overused)



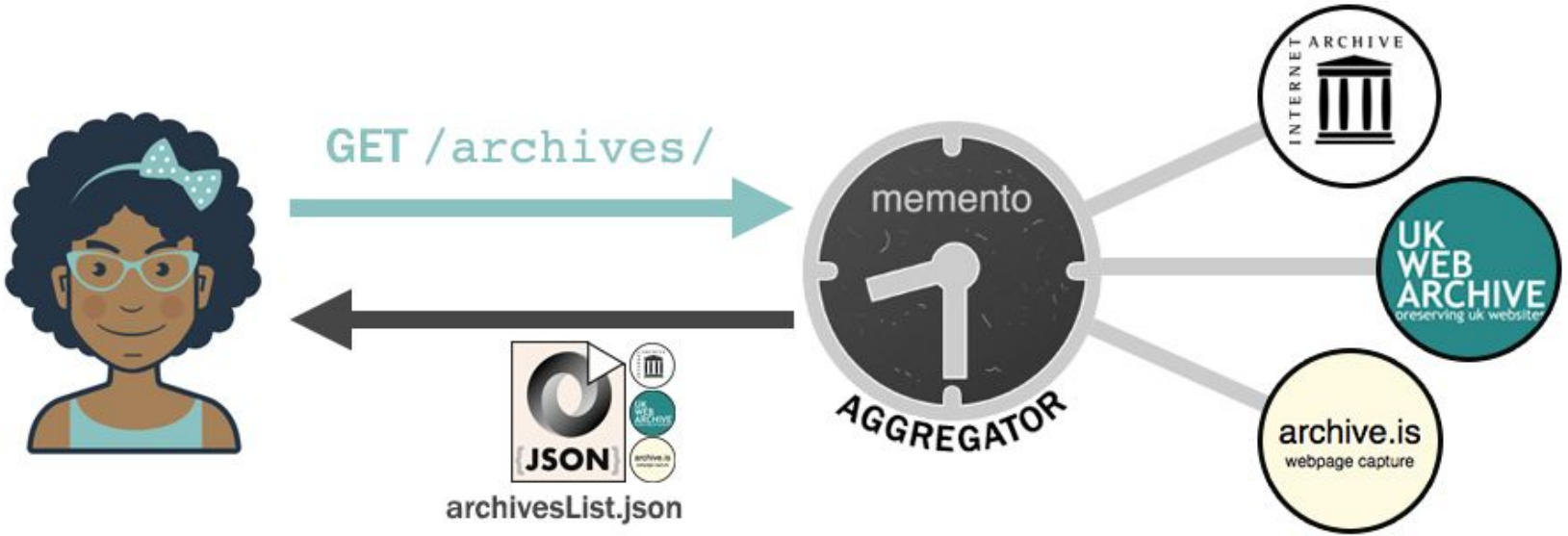
PROPOSED FRAMEWORK

Mementities

Memento Meta-Aggregator (MMA)



MMA: Archive Selection



MMA: User-Driven Archival Specification

MMA Aggregation sources

MMA_{α}:

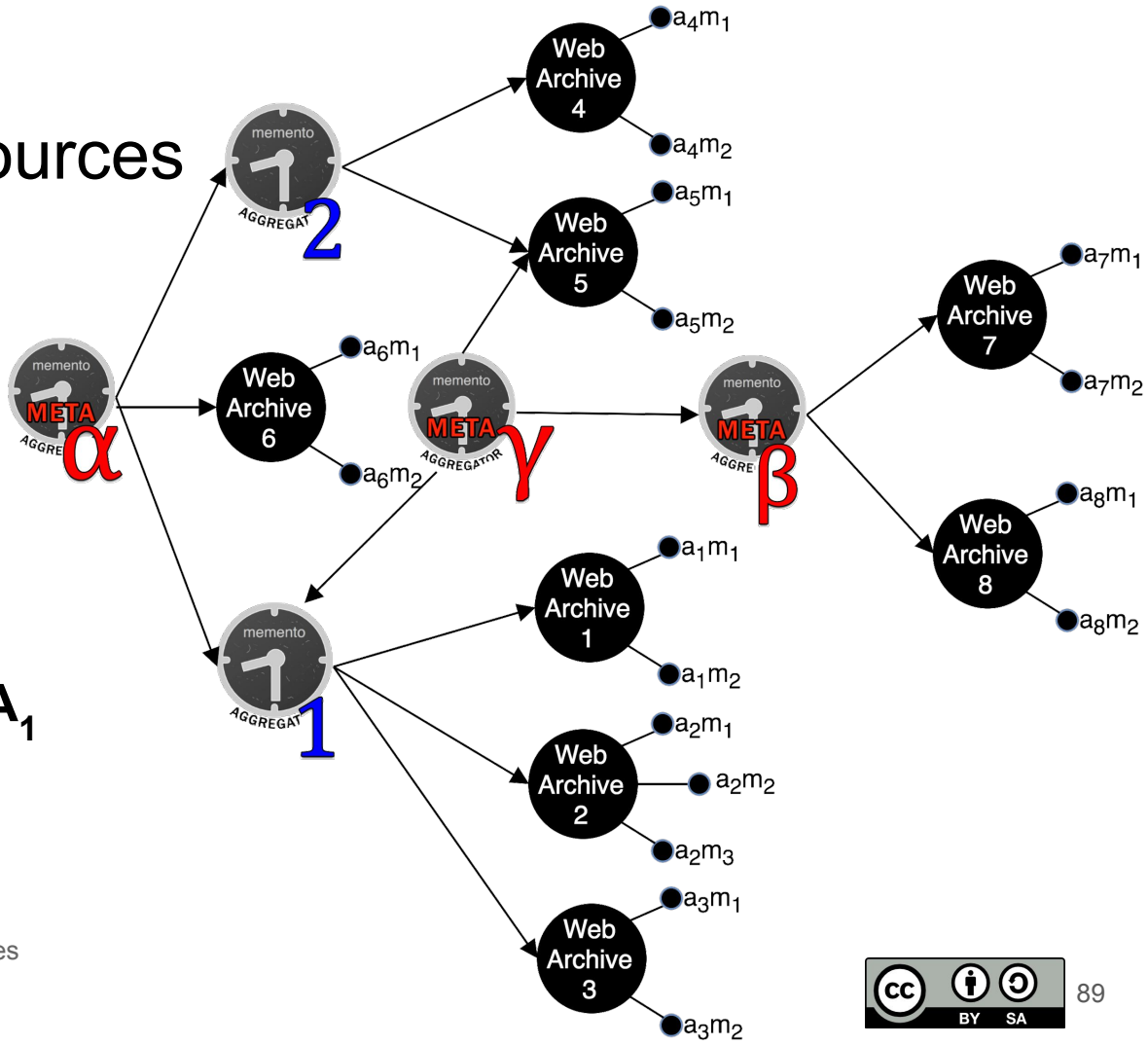
from MA_2, MA_1 and $\mathbf{WA_6}$

MMA_{β}:

from $\mathbf{WA_7}$ and $\mathbf{WA_8}$

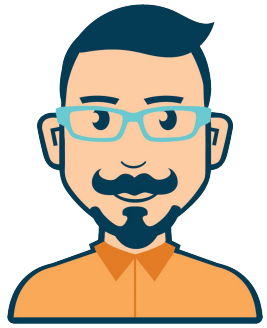
MMA_{γ}:

from MMA_{β}, MA_5, and $\mathbf{WA_1}$



MMA Dynamics By-Example

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



MMA Dynamics - Personal Archive Aggregation

bbc | homepage

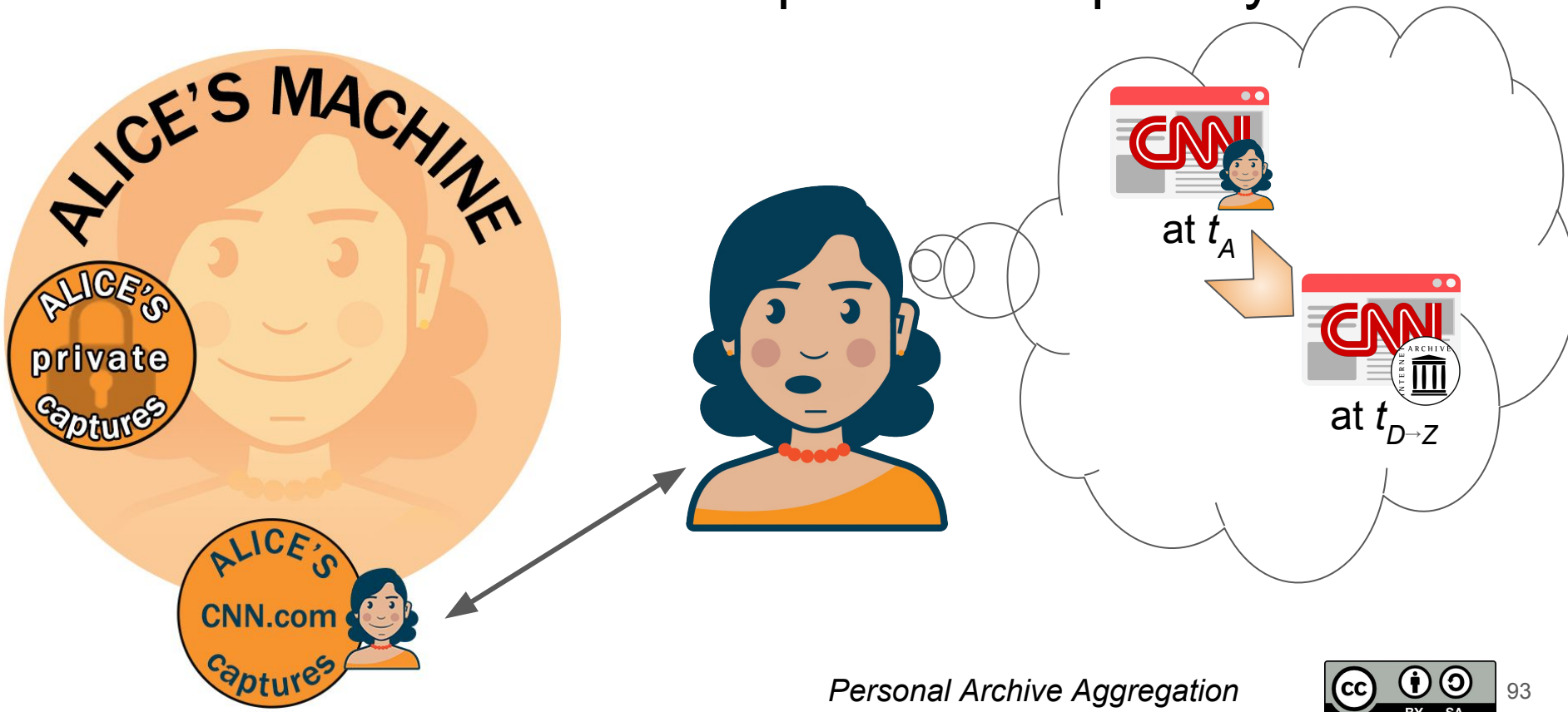
Public videos

FB | bank | flickr

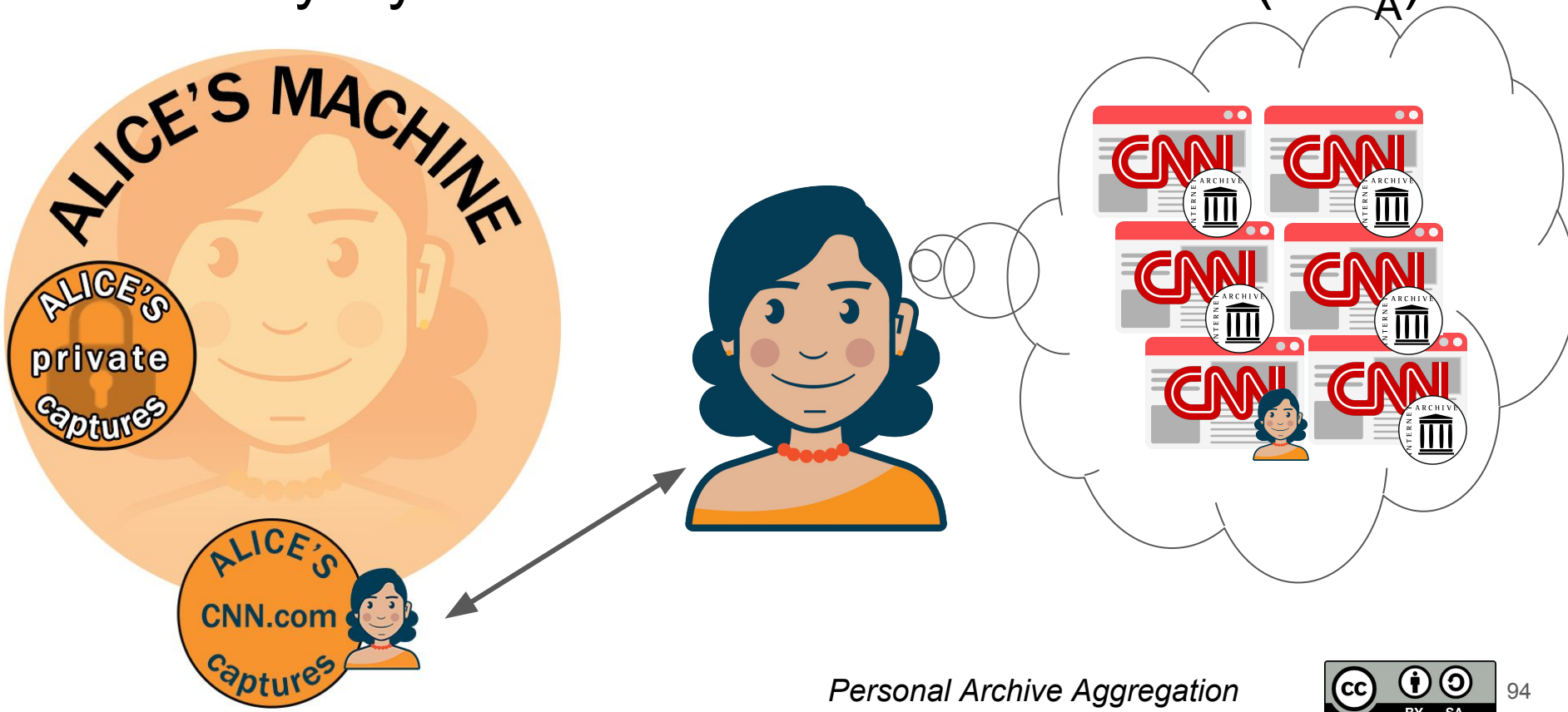
Personal Archive Aggregation

Alice Saves the Web

Alice Wants to See Her Captures Temporally Inline

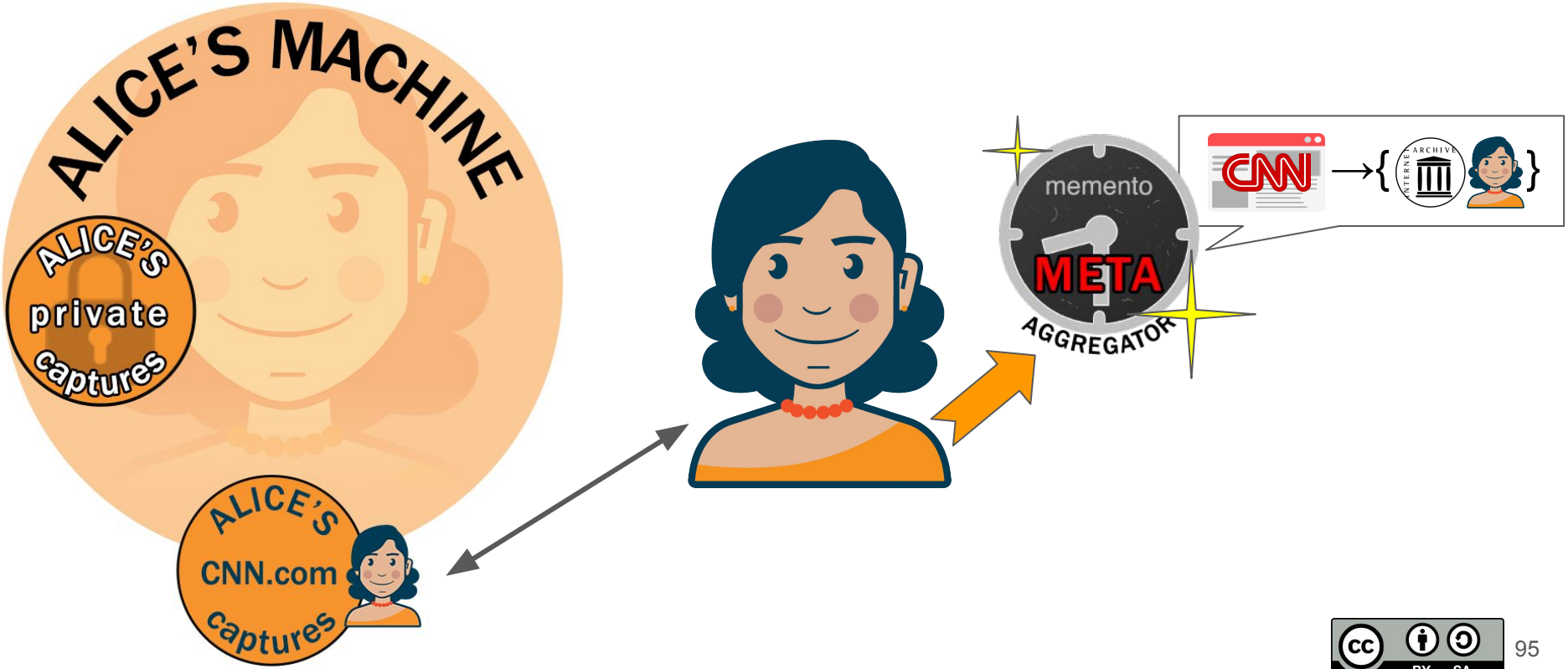


Mementity Dynamics - Alice & Her Archives (WA_A)

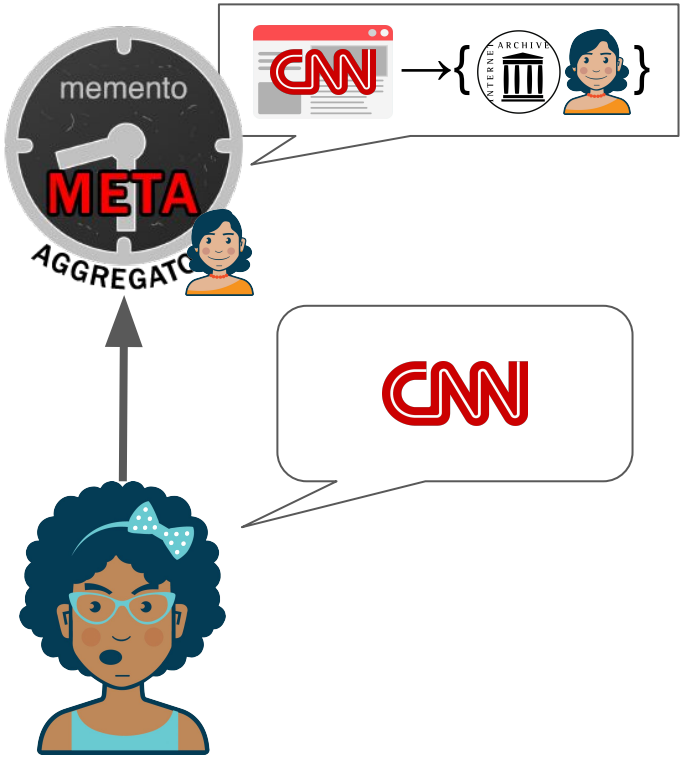


Personal Archive Aggregation

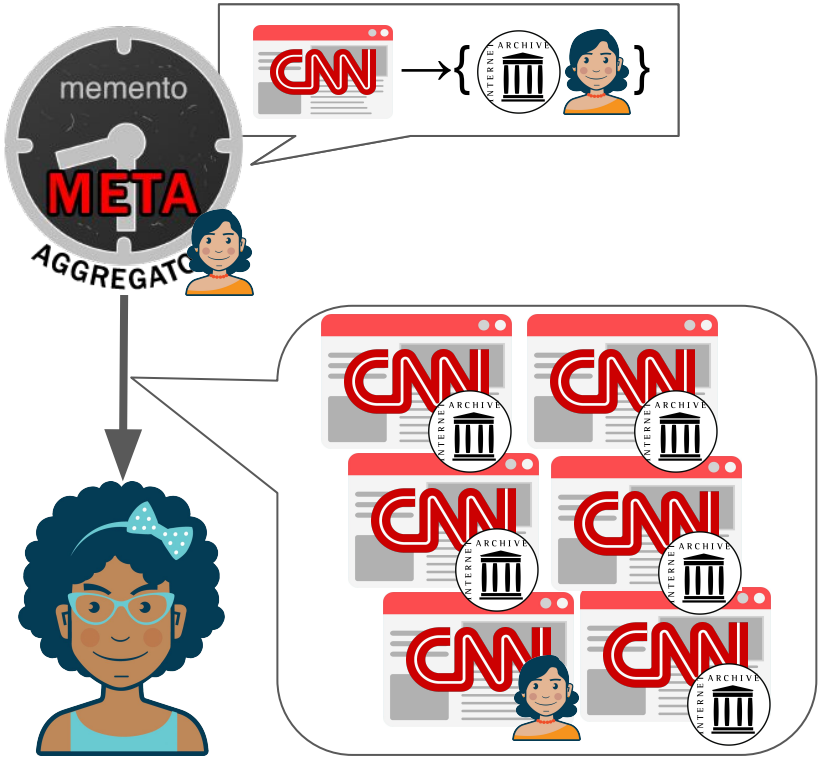
Alice Deploys MMA_A



Carol Asks MMA_A for CNN

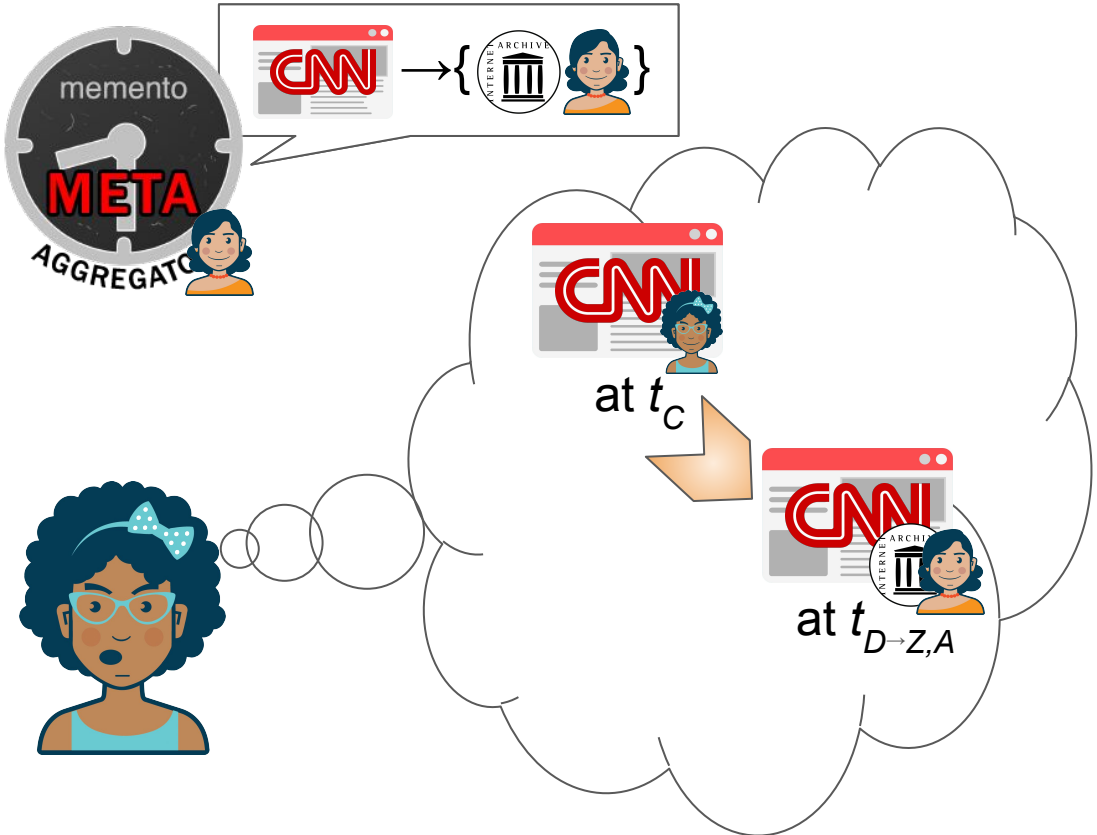


MMA_A returns CNN Memento $\{M_A, M_{IA}\}$

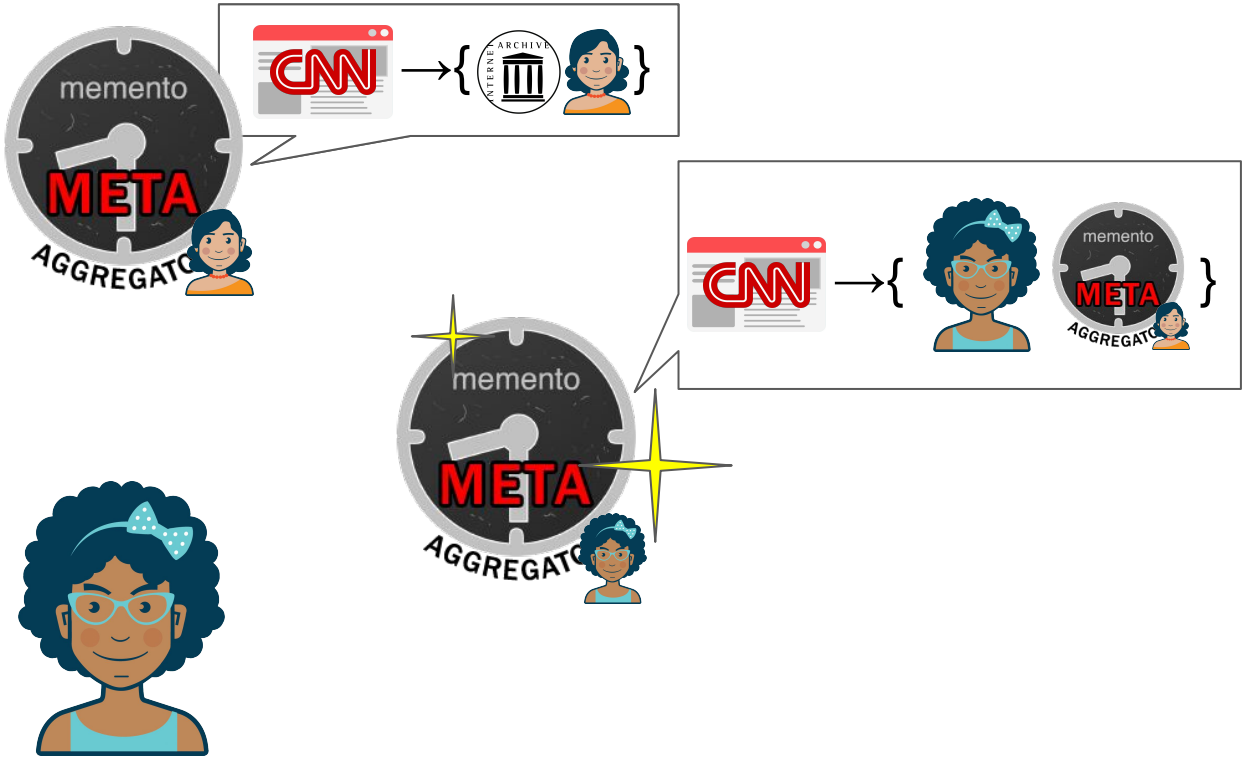


Carol Wants to Aggregate Her Own Captures

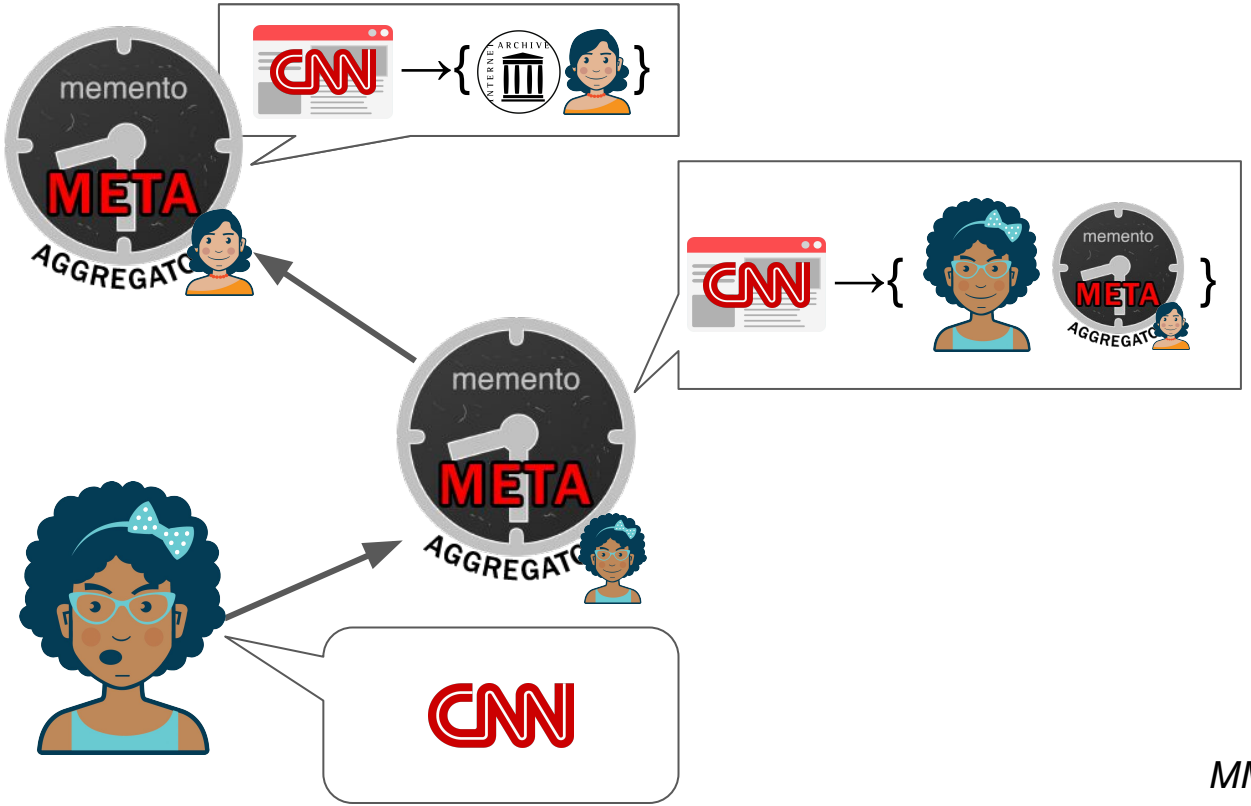
CNN(M(WA$_C$))



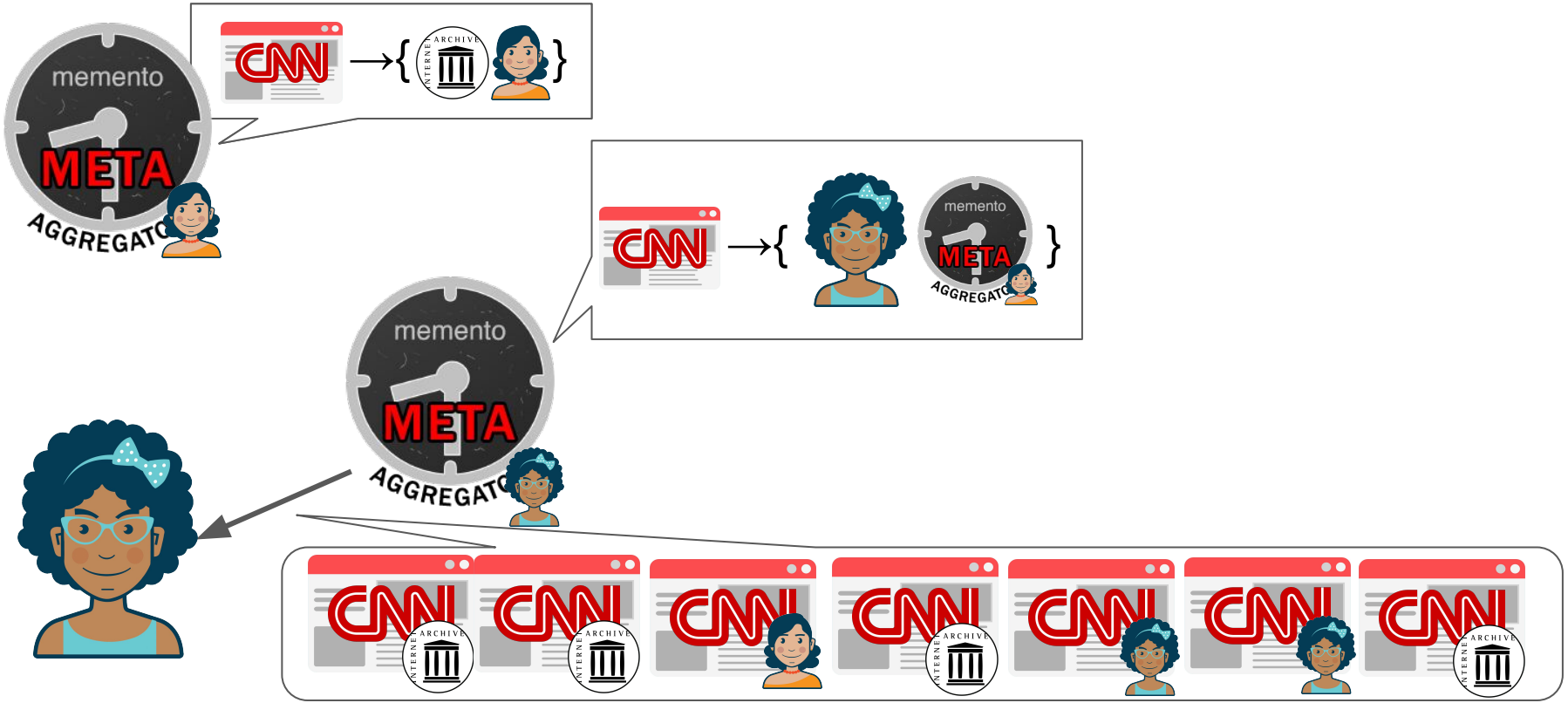
Carol Creates MMA_C to Access WA_C and MMA_A



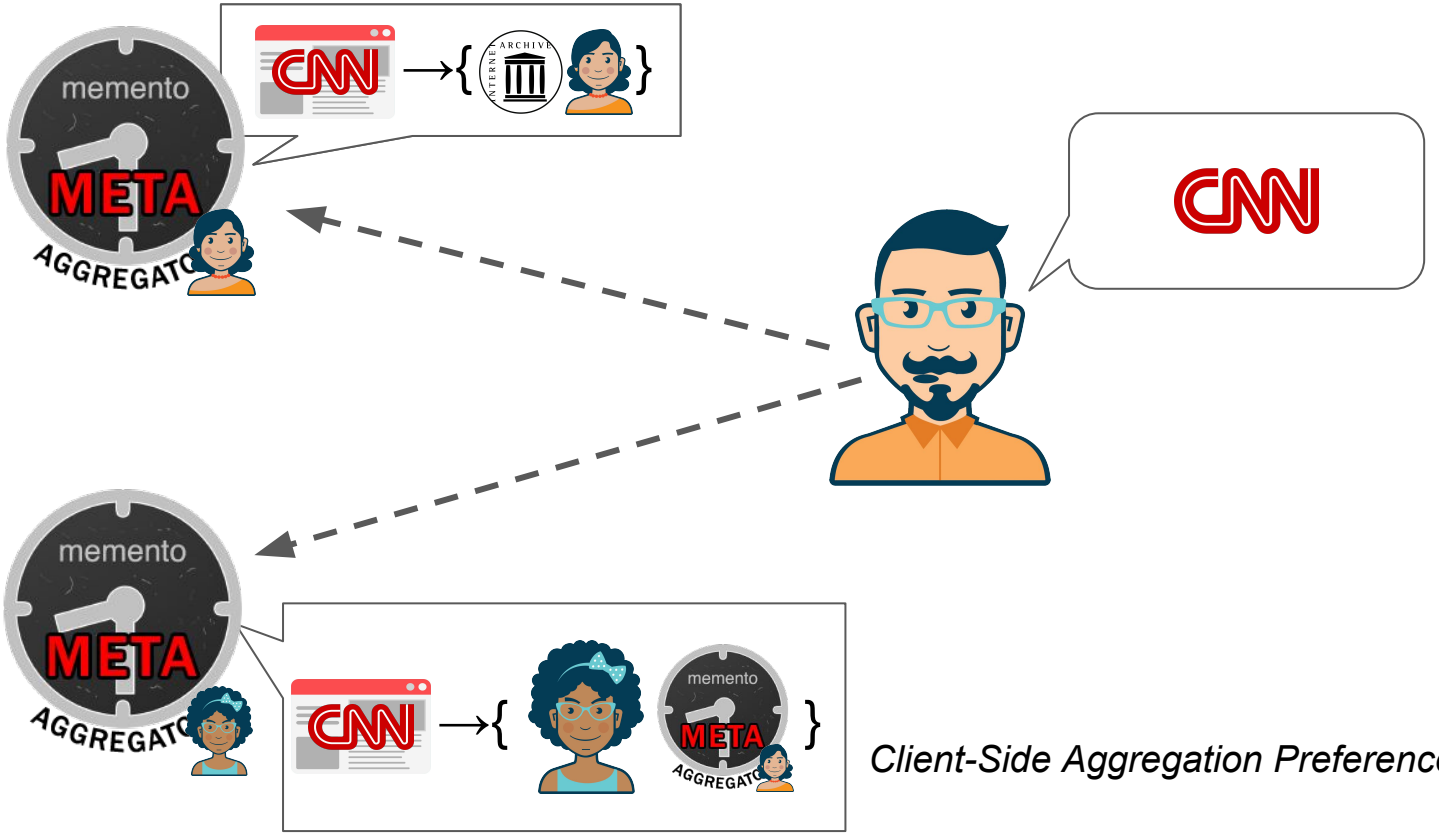
Carol Asks MMA_C For CNN



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



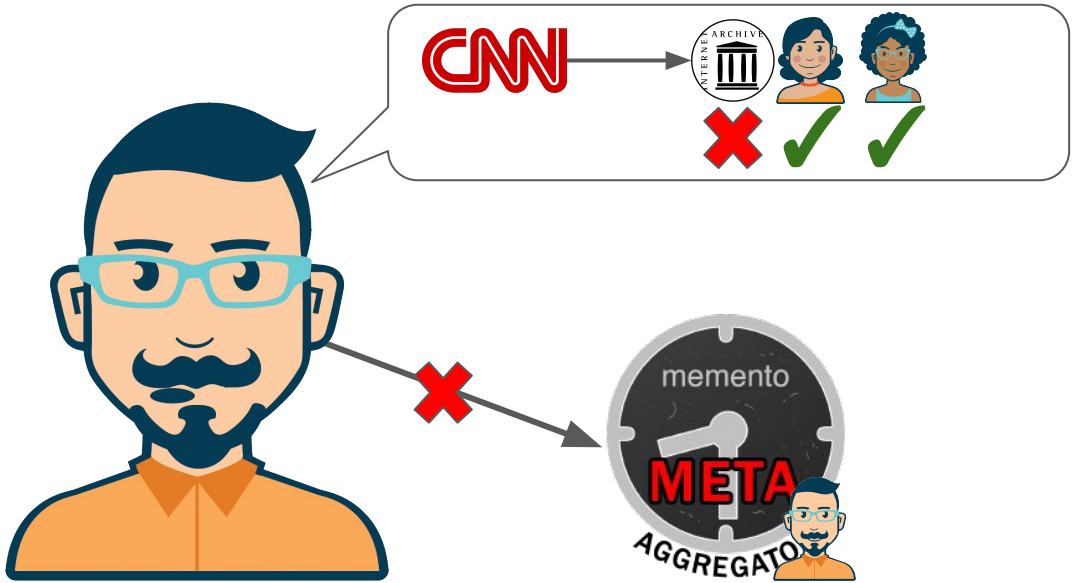
Bob May Request M(CNN) From MMA_A or MMA_C



Client-Side Aggregation Preference

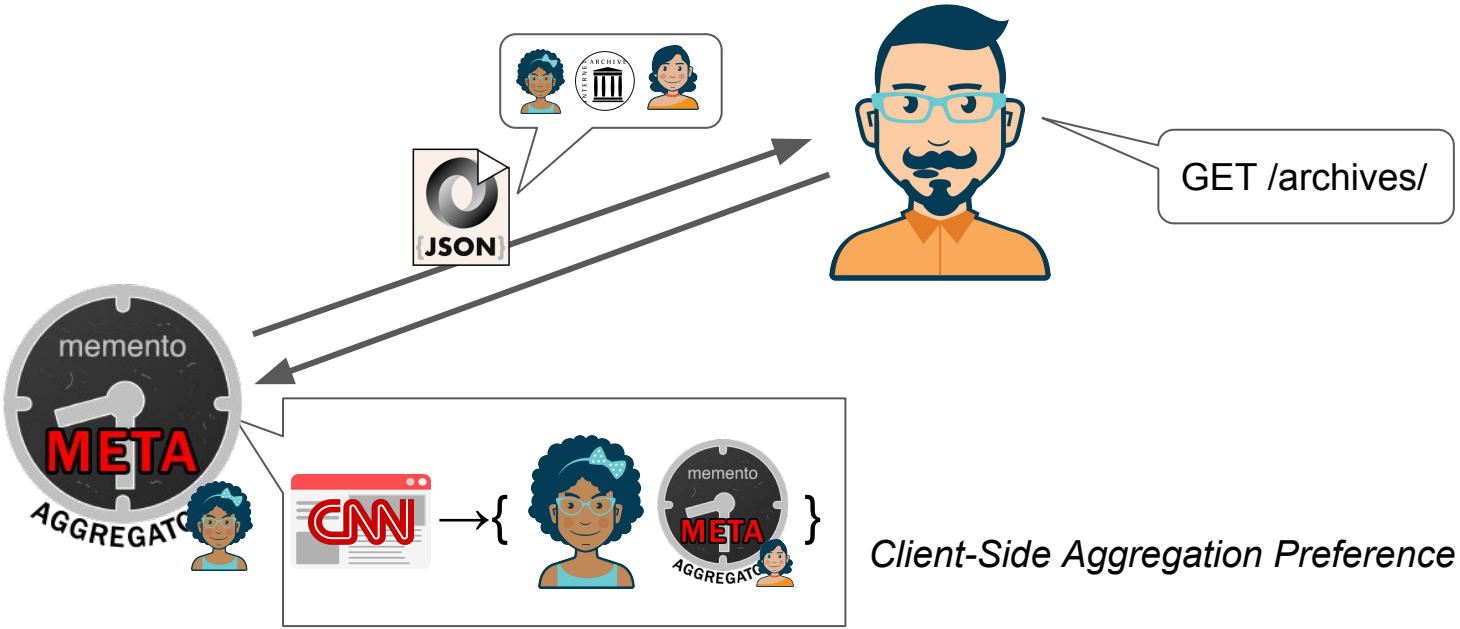
Bob Prefers to Exclude IA Captures

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



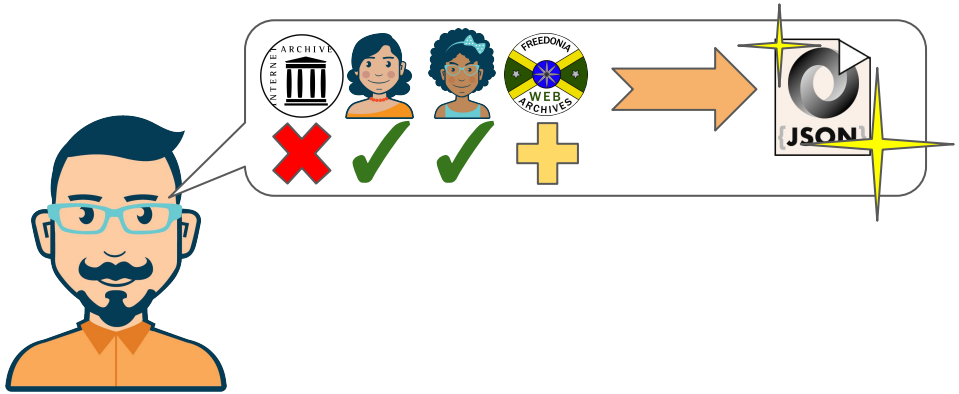
Client-Side Aggregation Preference

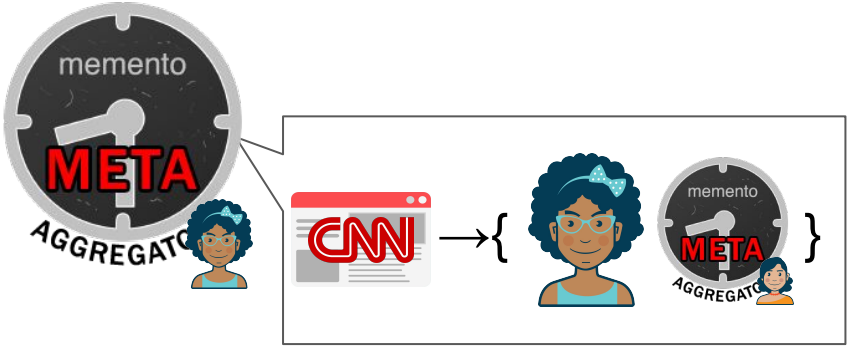
Bob Requests Supported Archives



Client-Side Aggregation Preference

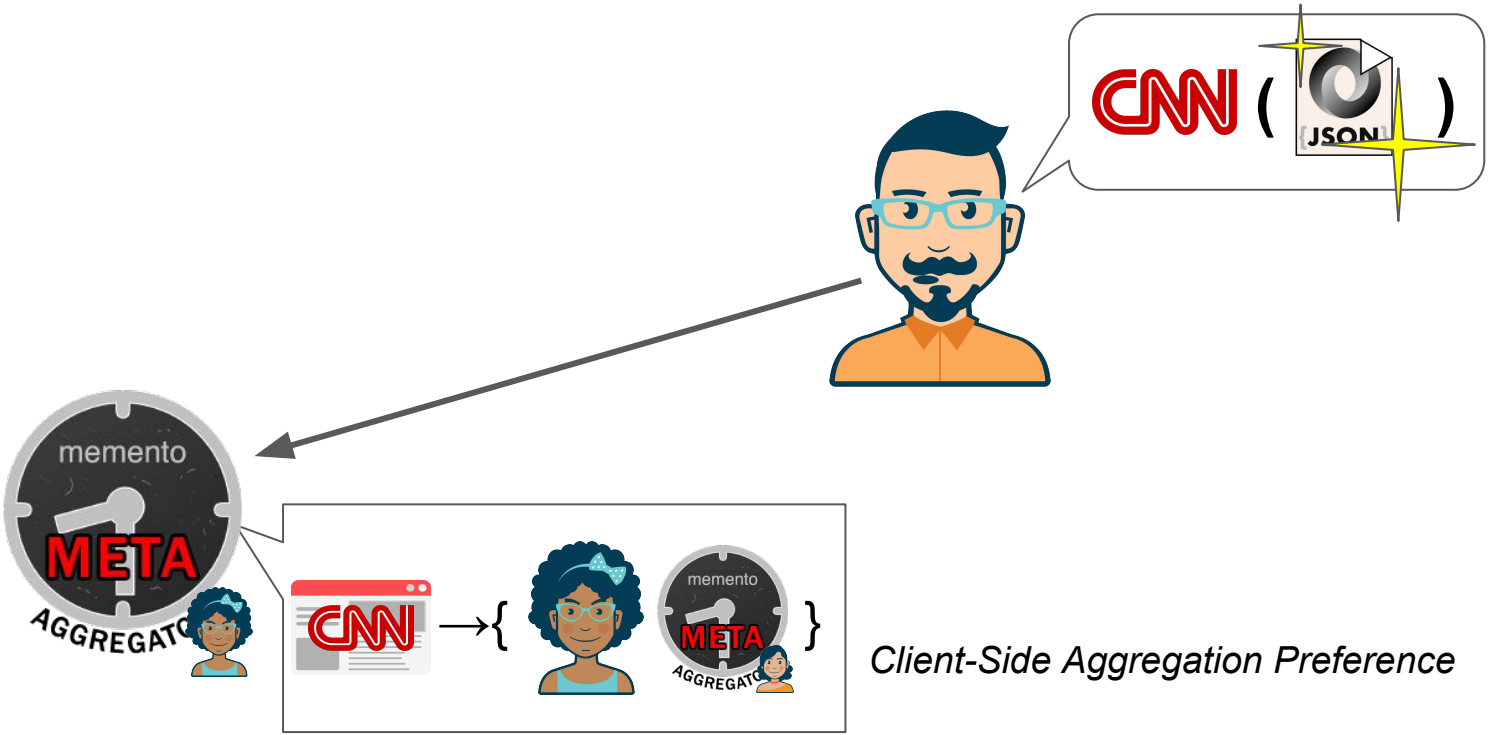
Bob Customizes the Set in the JSON





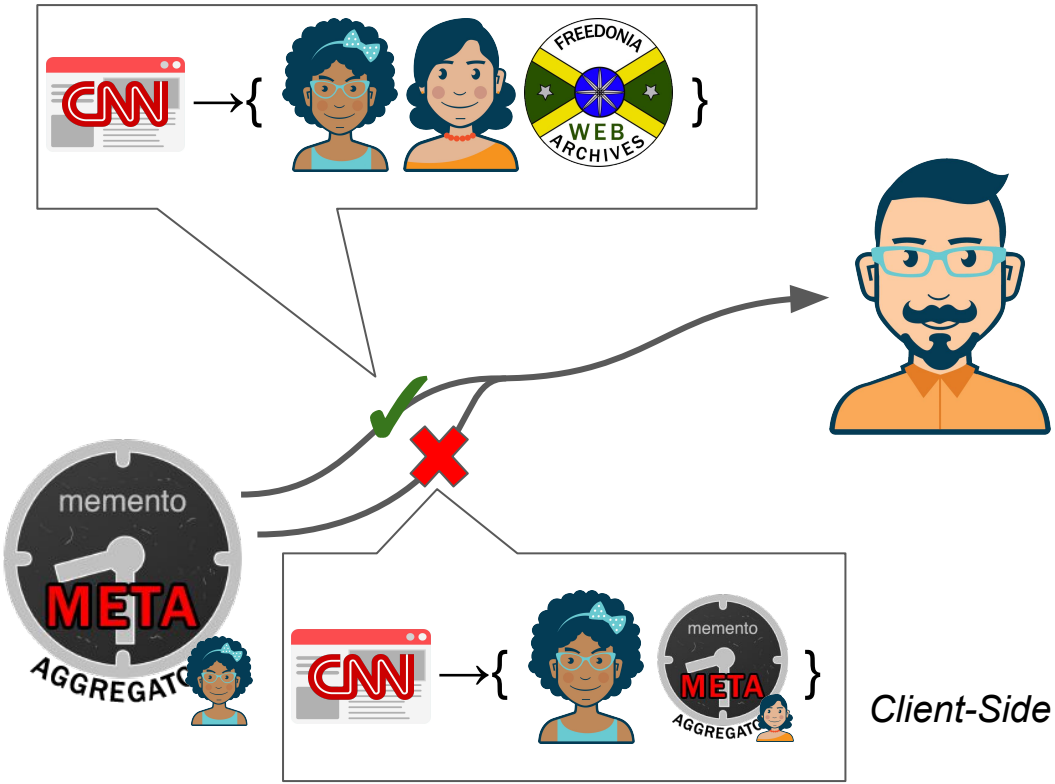
Client-Side Aggregation Preference

Bob Requests CNN for His Custom Set



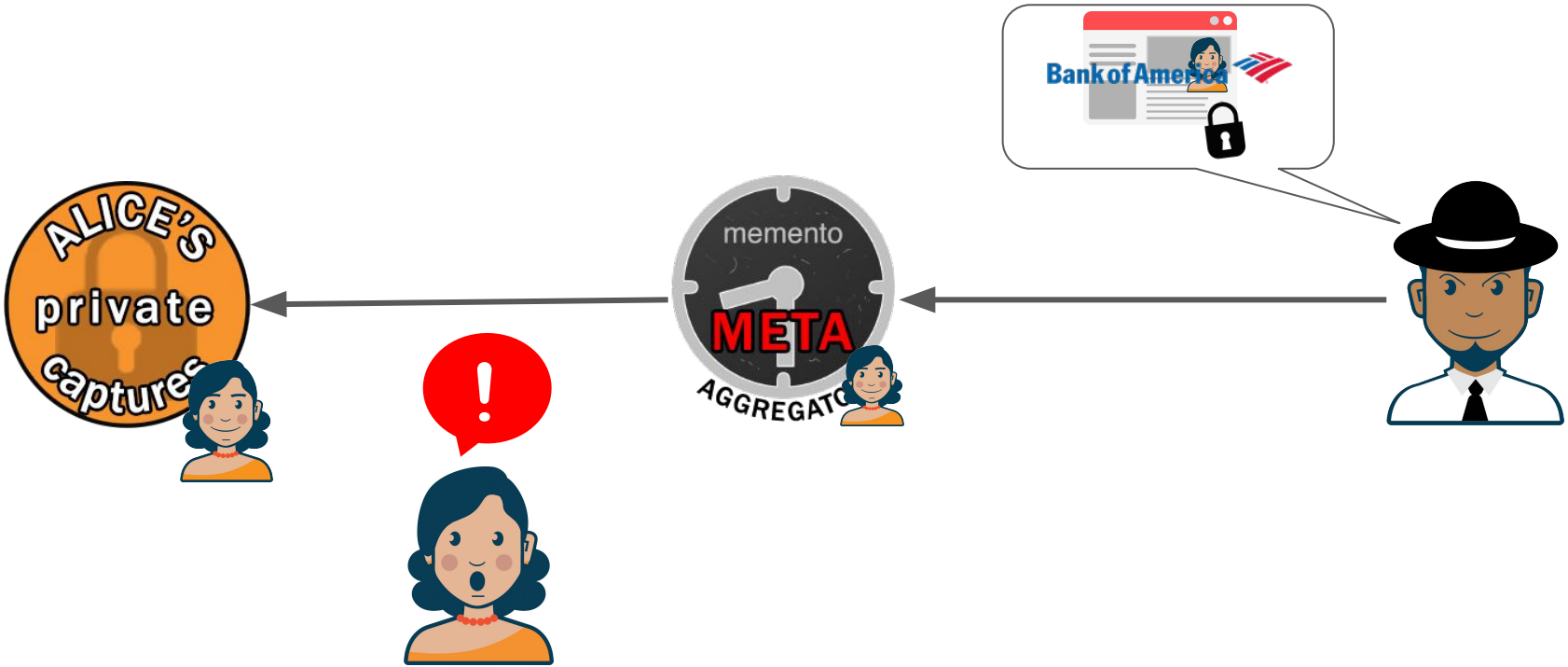
Client-Side Aggregation Preference

MMA Complies or Ignores Preference



Client-Side Aggregation Preference

Hooray, Aggregation!

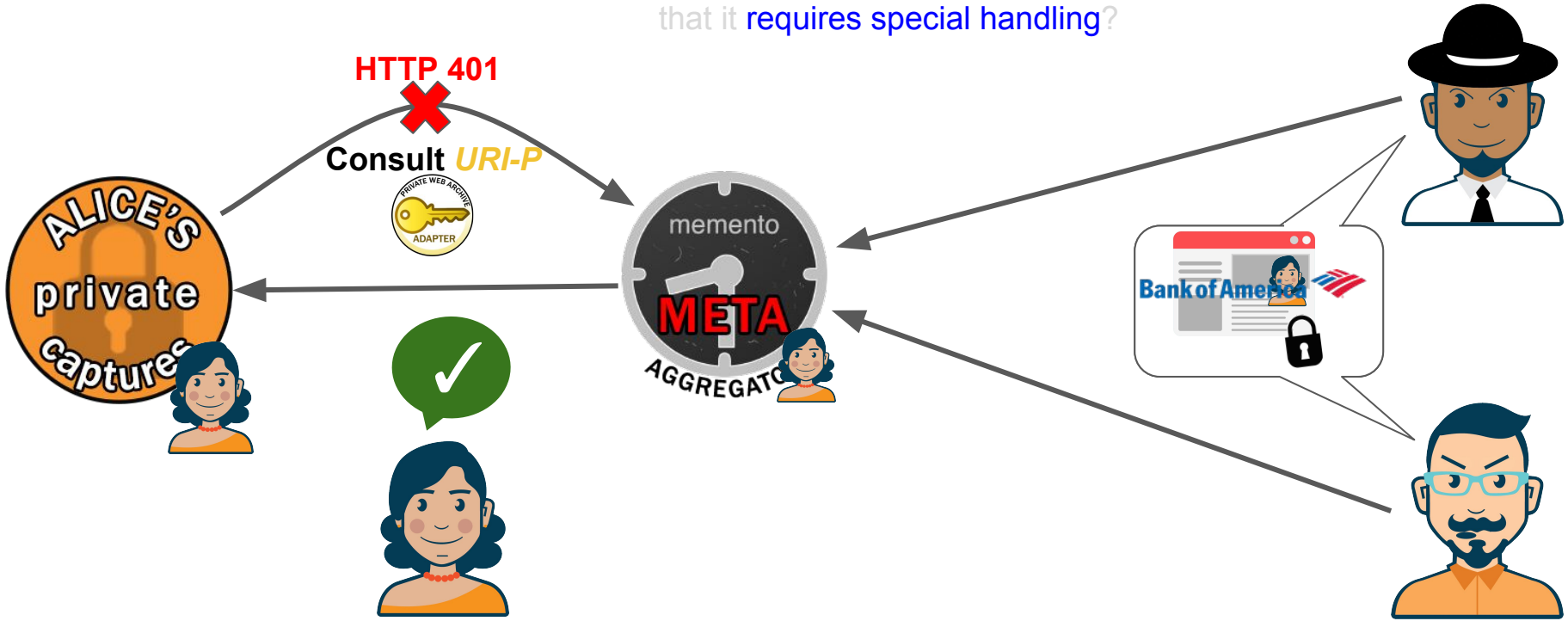


PROPOSED FRAMEWORK

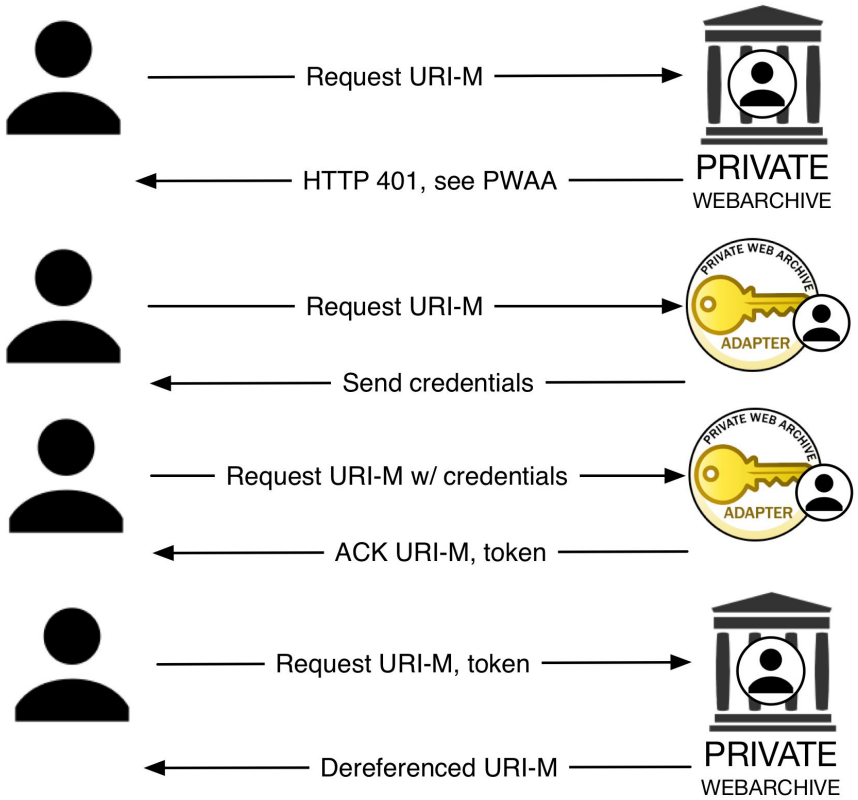
Mementities

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)

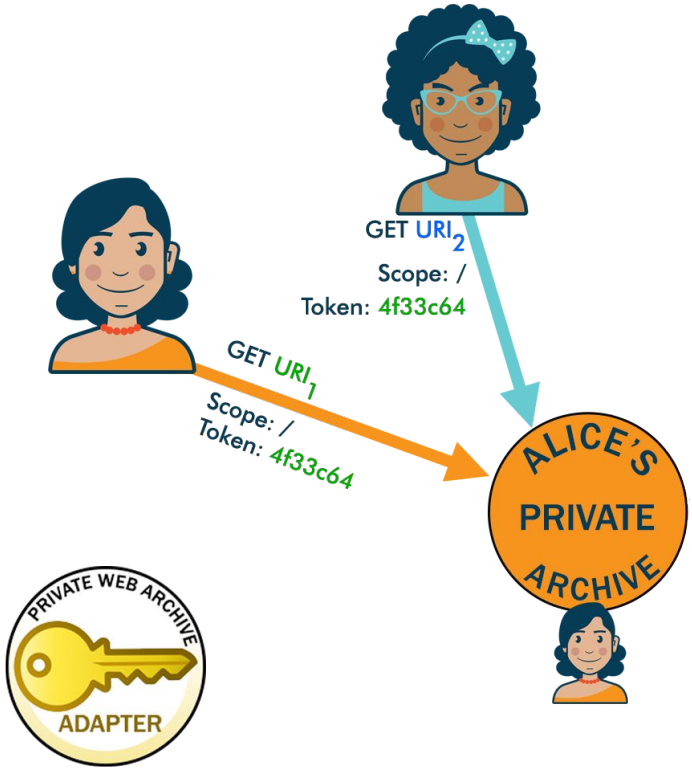


- Auth Layer for to encourage Private Web archive aggregation
- Typical OAuth 2.0 flow
- Auth role cohesive to PWAA
- Persistent access through tokenization

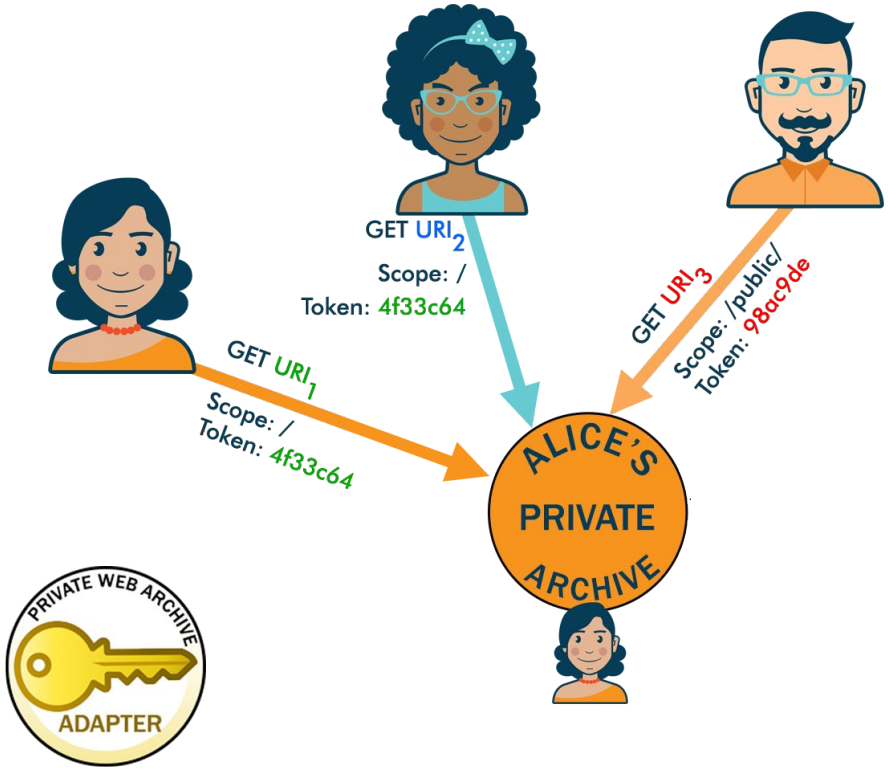
RQ6: What kinds of access control do users who create private Web archives need to regulate access to their archives?

PWAA - Sharing Tokens

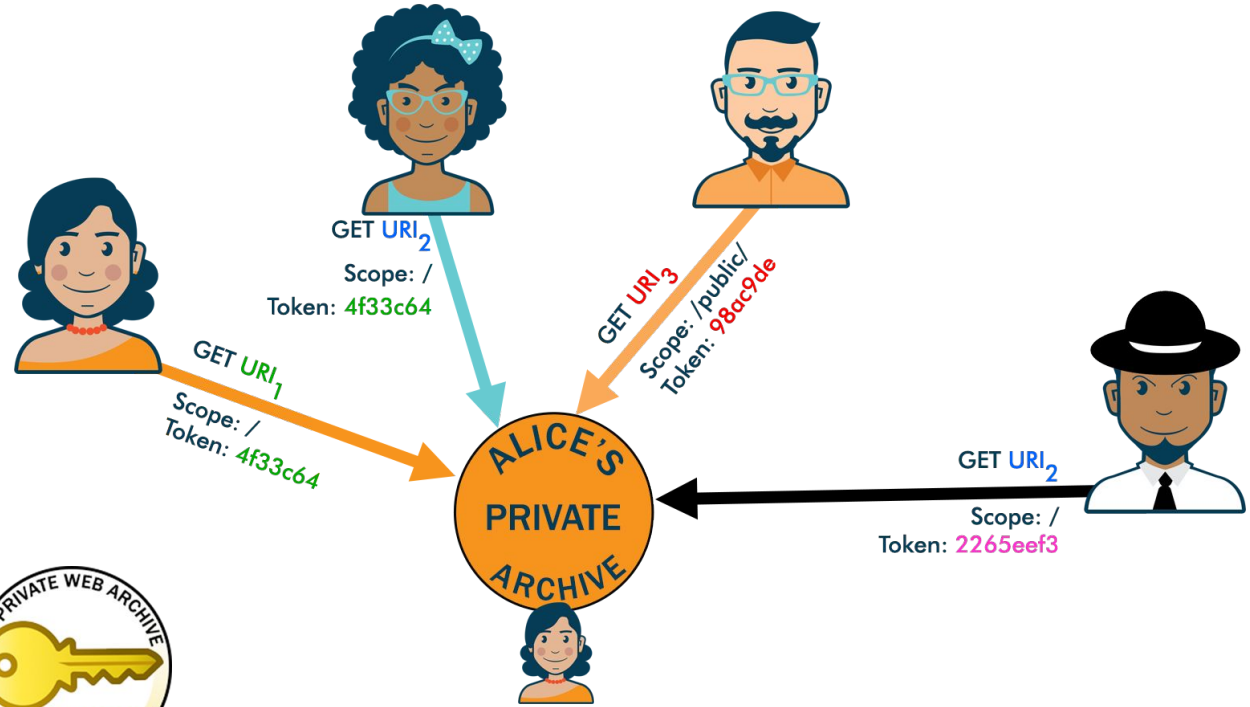
RQ6: What kinds of access control do users who create private Web archives need to regulate access to their archives?



PWAA - Previously Authorized

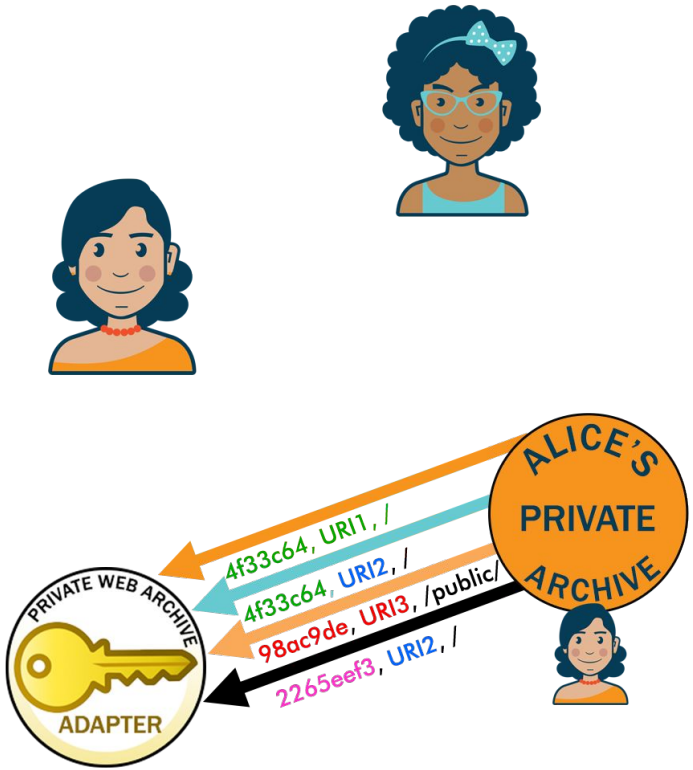


PWAA - Unauthorized Request

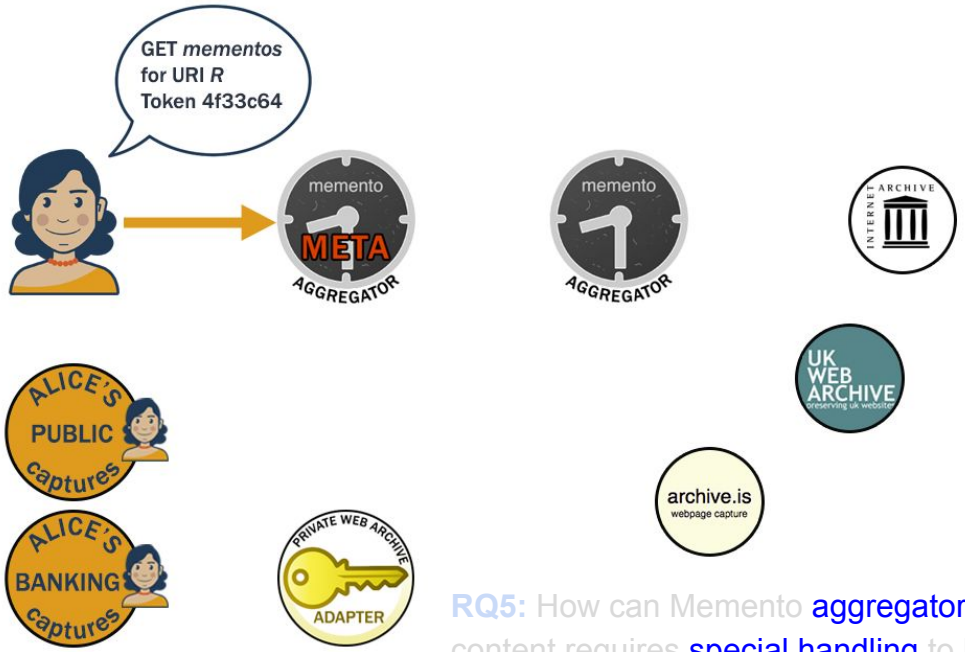


PWAA - Sharing Tokens

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



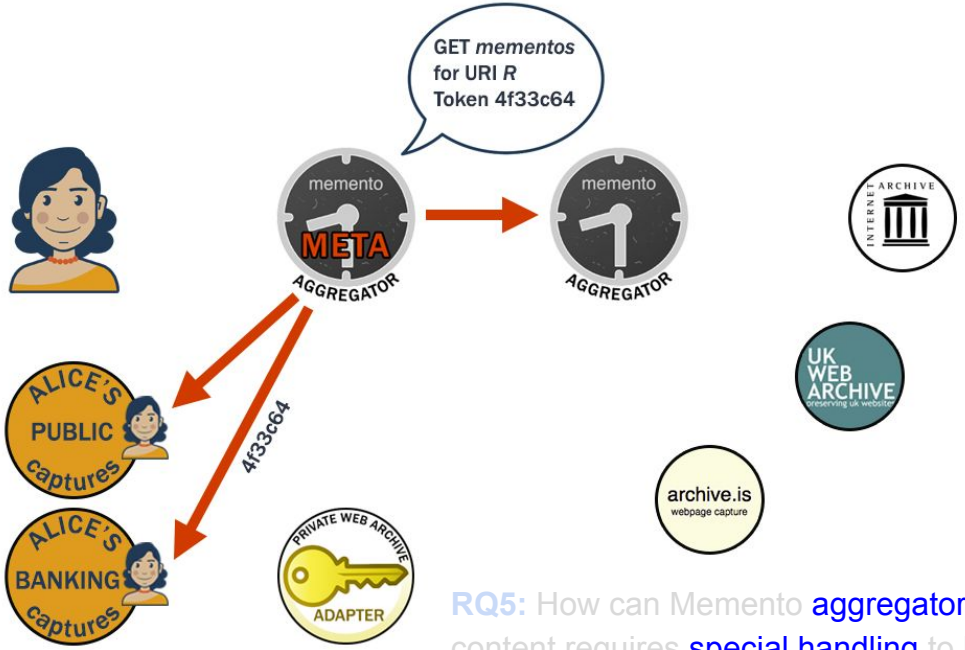
Alice Passes Associative Token to MMA



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?

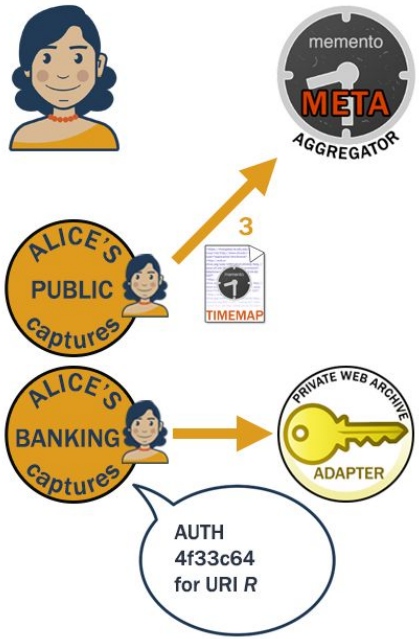
MMA requests URI-R...

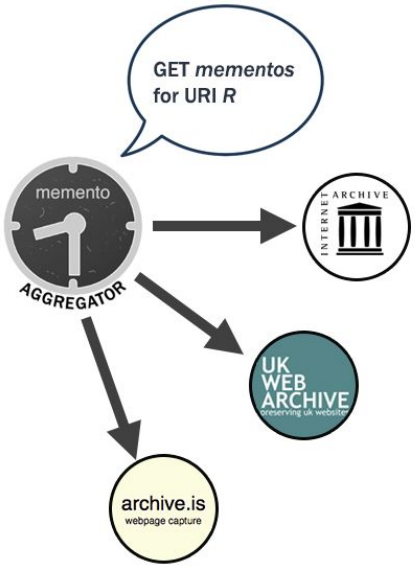
...relays token where applicable



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?

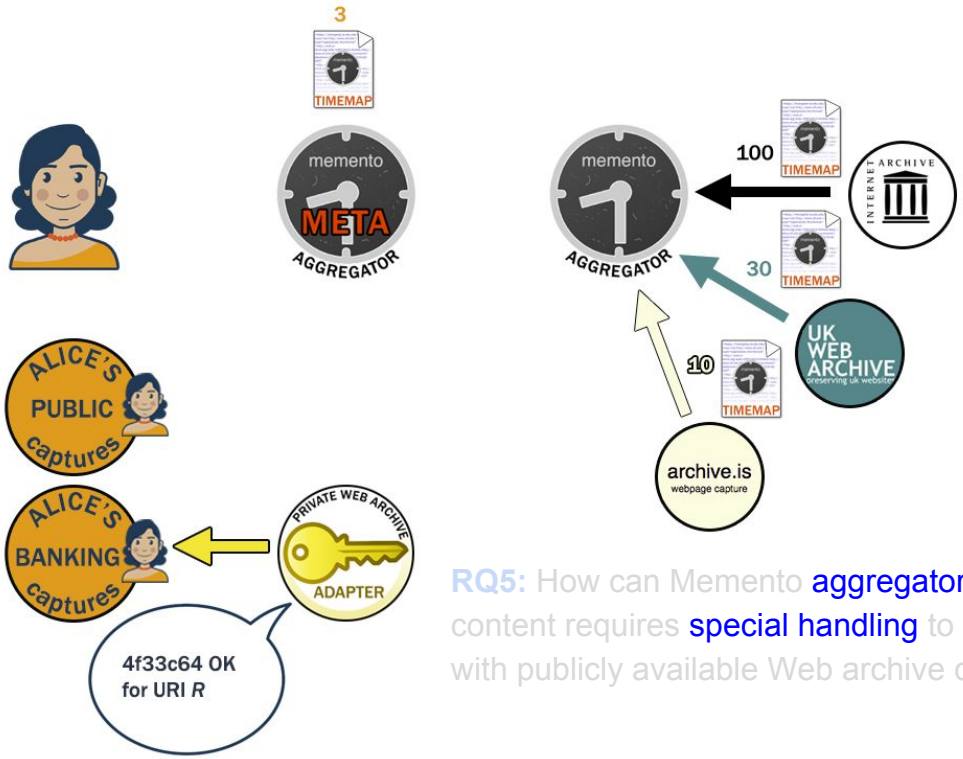
Private Archive Validates with PWAA





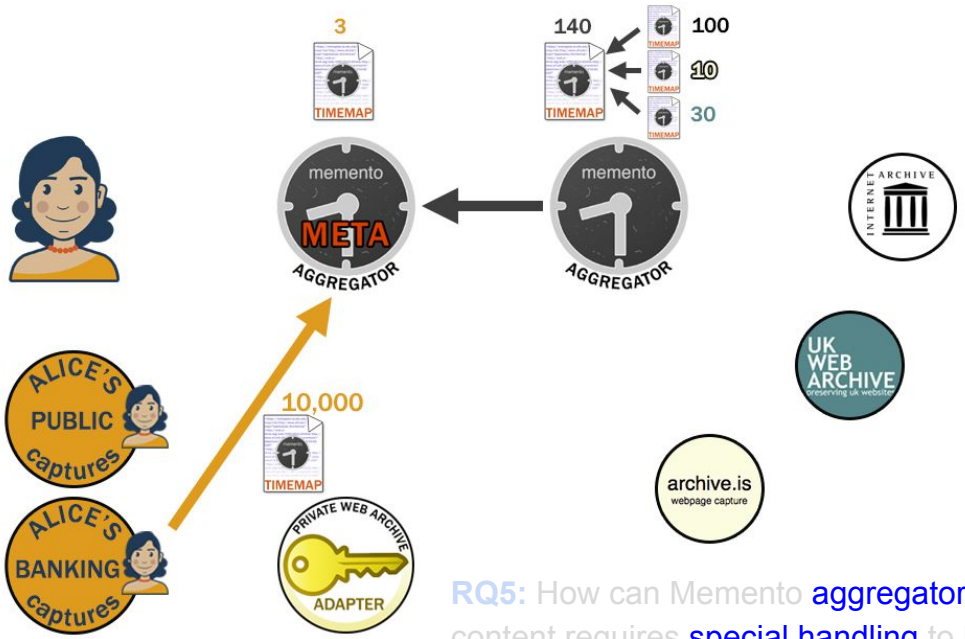
RQ5: How can Memento aggregators indicate that private Web archive content requires special handling to be replayed, despite being aggregated with publicly available Web archive content?

PWAA Confirms Token



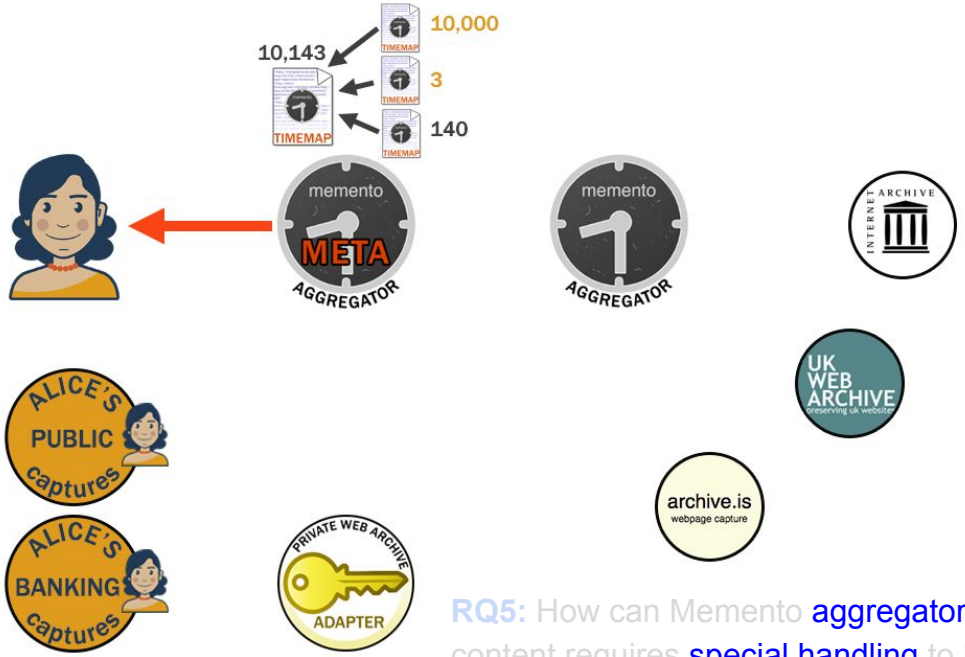
RQ5: How can Memento aggregators indicate that private Web archive content requires special handling to be replayed, despite being aggregated with publicly available Web archive content?

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

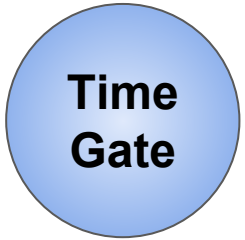


RQ5: How can Memento aggregators indicate that private Web archive content requires special handling to be replayed, despite being aggregated with publicly available Web archive content?

PROPOSED FRAMEWORK

Mementities

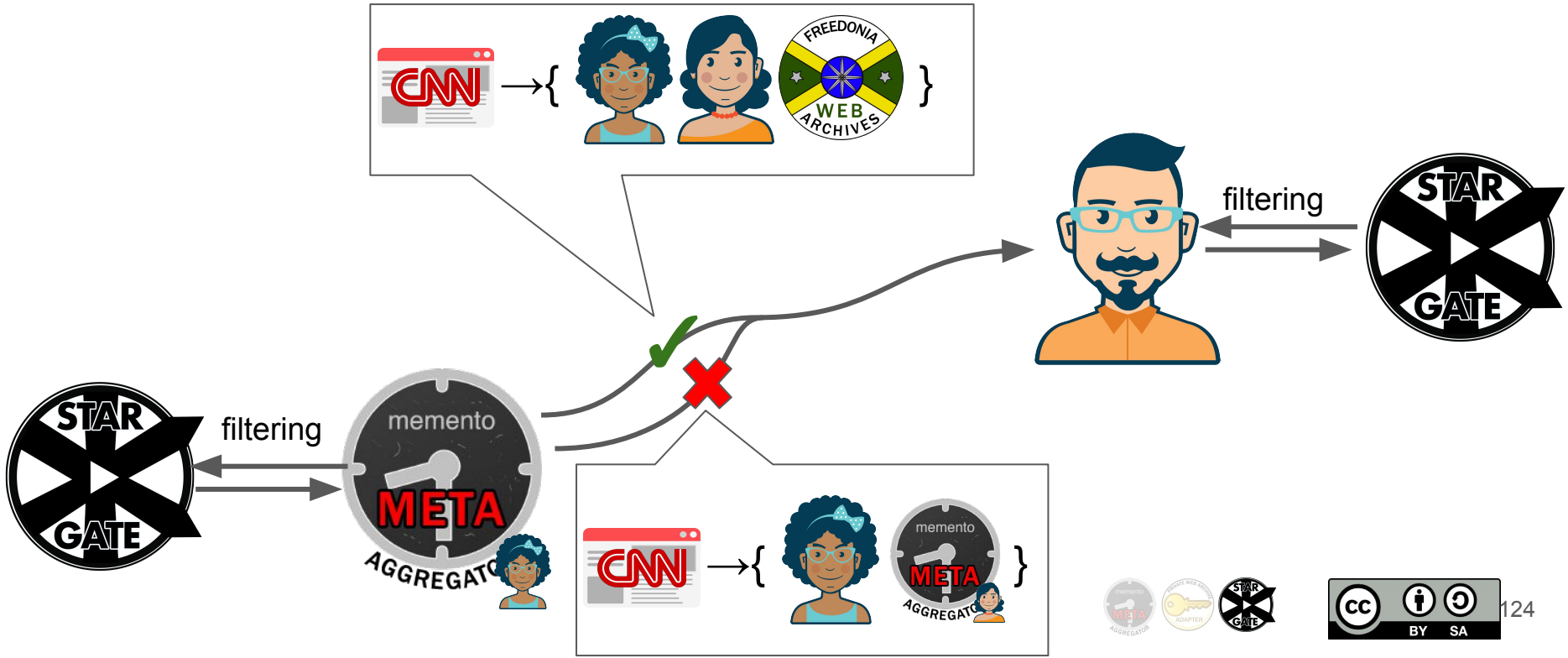
StarGate



functional $\subseteq$

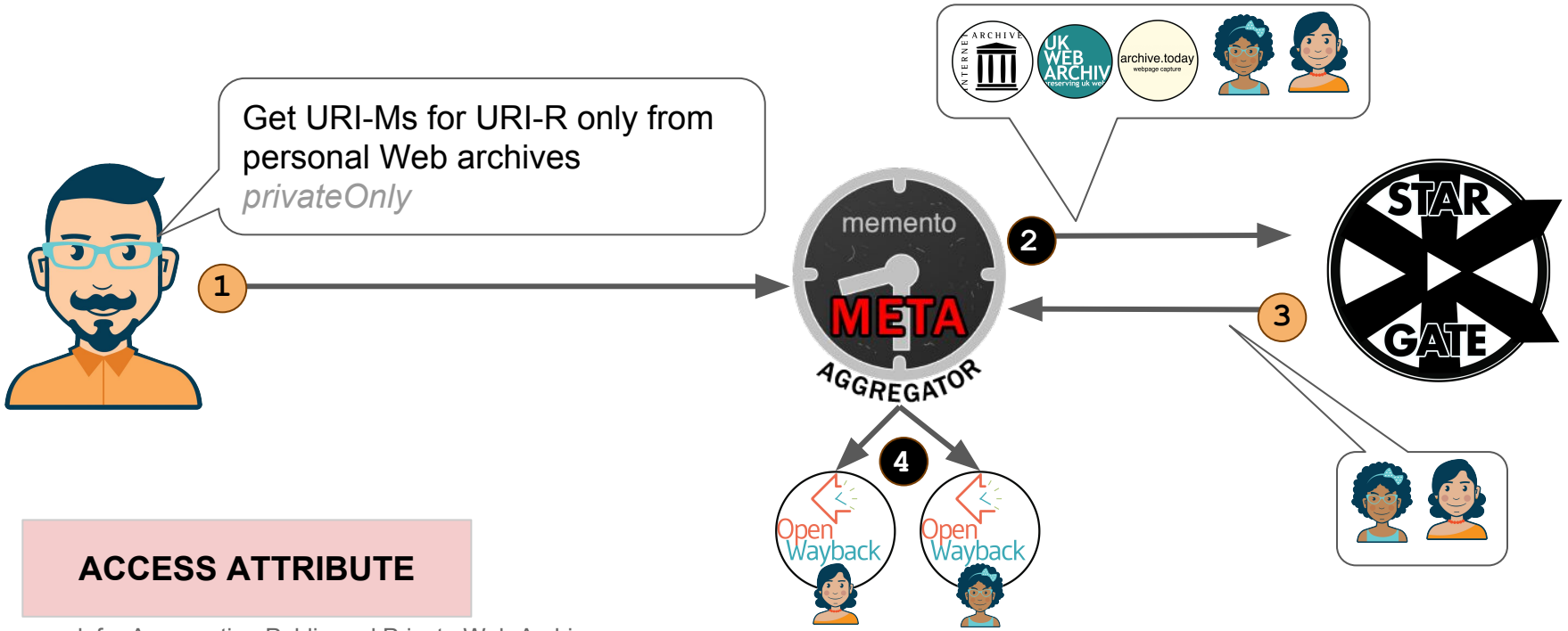
- Content negotiation in Web archives **beyond time**
- "Star" ~ wildcard (*) → **any** dimension of negotiation
- Allow for queries like: *Only show me mementos…*
  - That are not redirects (*content-based attribute* HTTP Status ≠ 3XX)
  - Of a sufficient quality (*derived attribute* Memento Damage < 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)



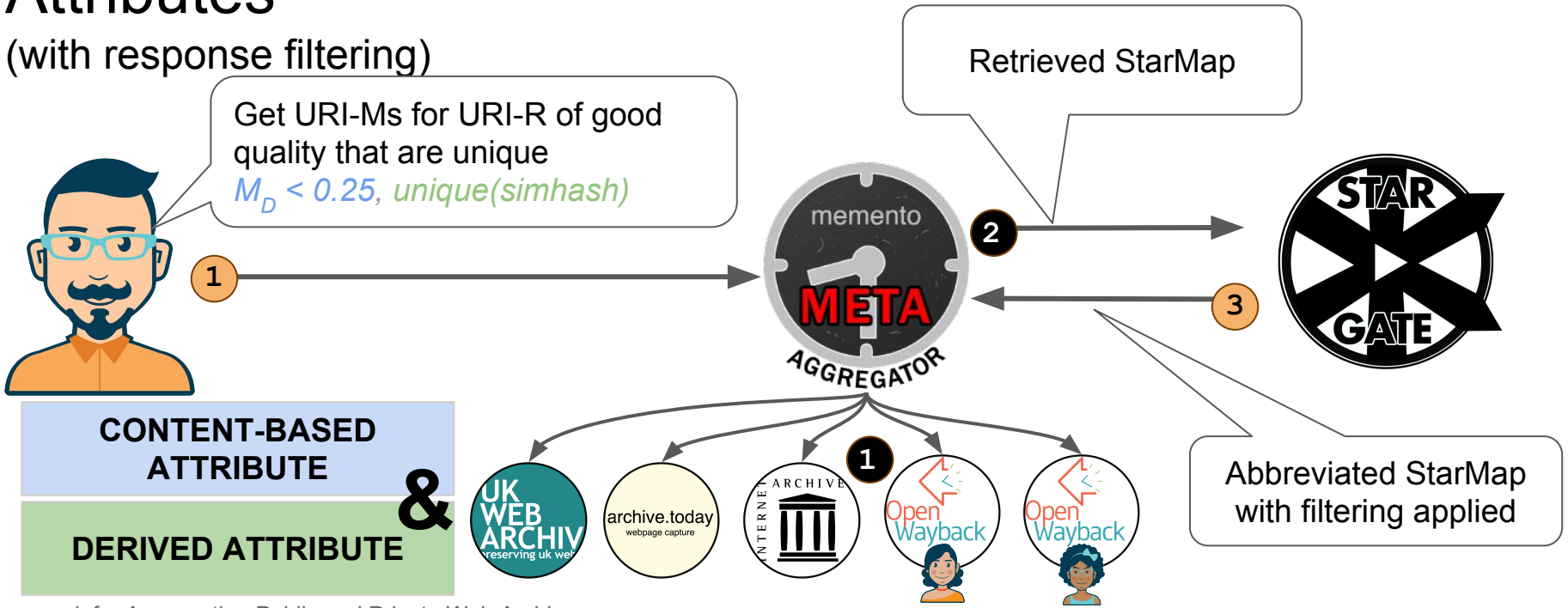
Negotiation in the Privacy Dimension

(via short circuiting)



Negotiation on Content-Based or Derived Attributes

(with response filtering)



Outline

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

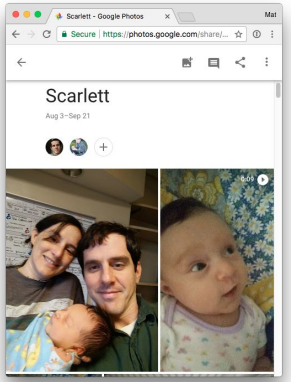
Framework Evaluation

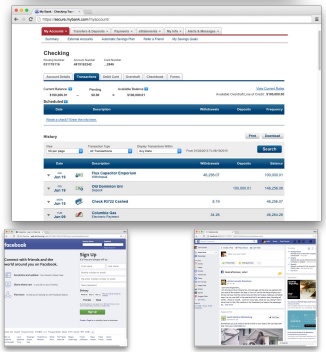
- Evaluation of mementity design decisions 

- 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.”*





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

Extend for client-side archival specification

extend

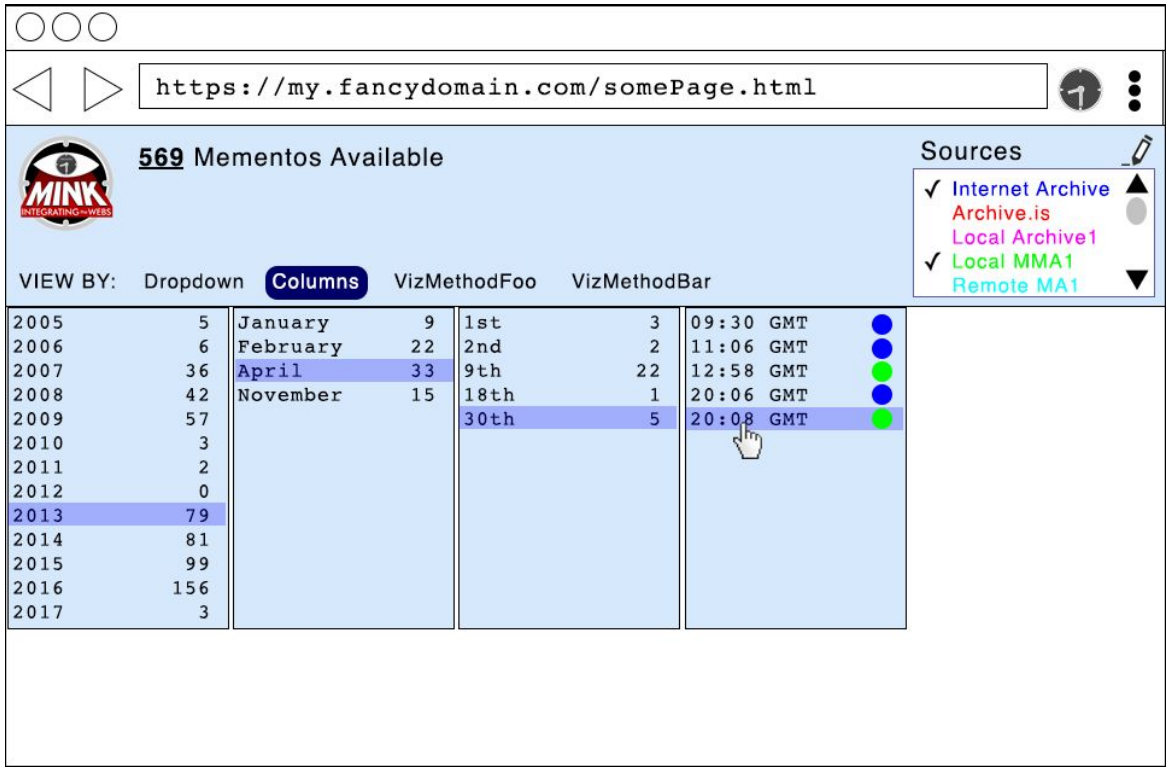
Exhibit features of an MMA

create

Regulate access to Private Web archives

create

Facilitate archival negotiation in more dimensions



EVALUATION: Design/Practicality • Enrichment costs • Implementation

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

OLD DOMINION UNIVERSITY

Seminar, Penn State University
February 14, 2019

WSDL Old Dominion University

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

Contribution beyond proposal

- 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 

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"
```

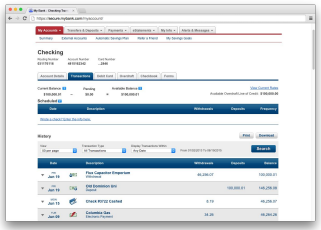
```
!context ["http://tools.ietf.org/html/rfc7089"]
!id {"uri": "http://localhost:1208/timemap/cdxj/http://matkelly.com"}
!keys ["memento_datetime_YYYYMMDDhhmmss"]
!meta {"original_uri": "http://matkelly.com"}
!meta {"timegate_uri":
"http://localhost:1208/timegate/http://matkelly.com"}
!meta {"timemap_uri": {"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"}
```

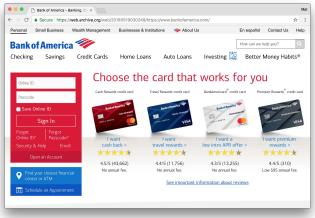
Link (RFC 7089) TimeMap | **CDXJ TimeMap**

Original URI (URI-R) | Other TimeMaps (URI-Ts) | TimeGate (URI-G) | Relative Relations

See Alam, "CDXJ: An Object Resource Stream Serialization Format", 2015

Private & Public Archives May Differ for the Same URI





Should Public Archives *Really* Capture the Private Web?