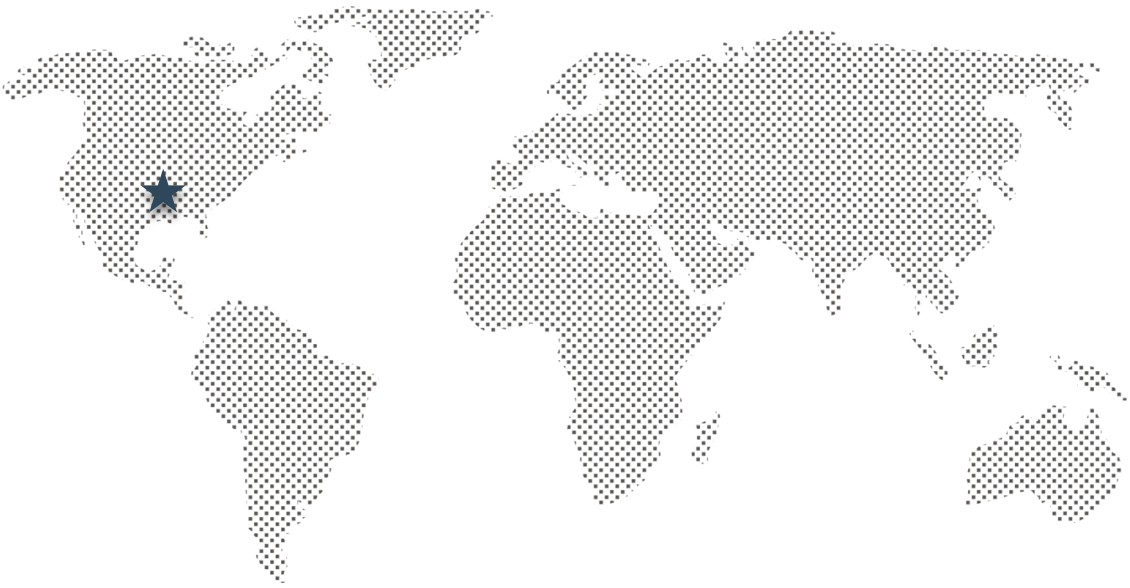
Proceedings *of the*



2018

Web Archiving & Digital Libraries

Workshop

June 6, 2018

Fort Worth, Texas

Martin Klein

Edward A. Fox

Zhiwu Xie

Editors

WADL 2018 homepage

Web Archiving and Digital Libraries
Workshop at JCDL 2018 (http://2018.jcdl.org)
Fort Worth, TX, USA

Please see the approved WADL 2018 workshop proposal.

Please also see last year's WADL 2017 homepage and the homepage of the 2016 edition of WADL. That workshop led in part to a special issue of International Journal on Digital Libraries.
We fully intend to publish a very similar IJDL issue based on WADL 2018 contributions.

SCHEDULE:

Featured Talk by Weigle, Michele C. (mweigle@cs.odu.edu): Enabling Personal Use of Web Archives

Wednesday, June 6, 10:30am-5pm

Time	Activity, Presenters/Authors	Title of Presentation
10:30	Organizers and everyone speaking	Opening, Introductions
10:45	Zhiwu Xie et al.	IMLS project panel
11:45	John Berlin, Michael Nelson and Michele Weigle	Swimming In A Sea Of JavaScript, Or: How I Learned To Stop Worrying And Love High-Fidelity Replay
12:15	Get boxes and return	Lunch
1:00	Liuqing Li and Edward Fox	A Study of Historical Short URLs in Event Collections of Tweets
1:30	Keynote by Michele Weigle	Enabling Personal Use of Web Archives
2:30	Libby Hemphill, Susan Leonard and Margaret Hedstrom	Developing a Social Media Archive at ICPSR
3:00	Posters: Littman Justin; Sawood Alam, Mat Kelly, Michele Weigle and Michael Nelson	Supporting social media research at scale; A Survey of Archival Replay Banners
3:15	Discussions around posters	Break
3:30	Mohamed Aturban, Michael Nelson and Michele Weigle	It is Hard to Compute Fixity on Archived Web Pages
4:00	Mat Kelly, Sawood Alam, Michael Nelson and Michele Weigle	Client-Assisted Memento Aggregation Using the Prefer Header
4:30	Closing discussion	Plans for future activities and collaborations

---

Description:

The 2018 edition of the Workshop on Web Archiving and Digital Libraries (WADL) will explore the integration of Web archiving and digital libraries. The workshop aims at addressing aspects covering the entire life cycle of digital resources and will also explore areas such as community building and ethical questions around Web archiving.

In addition, the chairs will initiate the workshop proceedings being published in a special issue of IEEE TCDL Bulletin.

WADL 2018 will cover all topics of interest, including but not limited to:

Archival Metadata, Description, Classification	Archival Standards, Protocols, Systems, and Tools	Collection Building
Community Building	Crawling of Dynamic and Mobile Content	Discovery of Archived Resources
Diversity in Web Archives	Ethics in Web Archiving	Extraction and Analysis of Archival Records
Focused Crawling	Social Media Archiving	Special Event Archiving

Objectives:

- to continue to build the community of people integrating Web archiving & digital libraries
- to help attendees learn about useful methods, systems, and software in this area
- to help chart future research and improved practice in this area
- to promote synergistic efforts including collaborative projects and proposals
- to produce an archival publication that will help advance technology and practice

Workshop Co-chairs:

- Chair, Martin Klein, Los Alamos National Laboratory Research Library, mklein@lanl.gov,
- Co-chair, Edward A. Fox, Professor and Director Digital Library Research Laboratory, Virginia Tech, fox@vt.edu http://fox.cs.vt.edu,
- Co-chair, Zhiwu Xie, Professor, Director of Digital Library Development, Virginia Tech Libraries, zhiwuxie@vt.edu,

Program Committee:

- Jefferson Bailey jefferson@archive.org Internet Archive
- Justin Brunelle jbrunelle008@gmail.com Old Dominion University
- Sumitra Duncan duncan@frick.org Frick Art Reference Library
- Joshua Finnell joshfinnell@gmail.com Colgate University
- Abbie Grotke abgr@loc.gov Library of Congress
- Olga Holownia olga.holownia@bl.uk British Library
- Gina Jones gjon@loc.gov Library of Congress
- Lauren Ko lauren.ko@unt.edu UNT Libraries
- Frank McCown fmccown@harding.edu Harding University
- Michael Nelson mln@cs.odu.edu Old Dominion University
- Nicholas Taylor ntay@stanford.edu Stanford Libraries

Other closely related events and results:

- Web Archiving and Digital Libraries (WADL'16), 22-23 June, at JCDL 2016, see website and proceedings in a special issue of the IEEE TCDL Bulletin, V. 13, Issue 1, April 2017

- Web Archiving and Digital Libraries (WADL'15), 24 June, at JCDL 2015, see website and proceedings in a special issue of the IEEE TCDL Bulletin, V. 11, Issue 2, Oct. 2015
- Working with Internet Archives for Research (WIRE 2014) NSF workshop, 17-18 June 2014, Cambridge, MA – see http://wp.comminfo.rutgers.edu/nsfia/
- Web Archiving and Digital Libraries (WADL'13), 25-26 July, at JCDL 2013, see http://www.ctrnet.net/sites/default/files/JCDL2013WorkshopWebArchiving20130603.pdf and report in SIGIR Forum http://sigir.org/files/forum/2013D/p128.pdf
- Web Archive Globalization Workshop, WAG 2011 – see http://cs.harding.edu/wag2011/ , with 4 organizers plus 5 presenters and about 20 participants, held in Ottawa after JCDL 2011 (June 16-17)
- Ongoing work by attendees in this area, growth in collaborative activity involving the Internet Archive, and specific community building successes like the Web Archive Cooperative – see http://infolab.stanford.edu/wac/
- Annual meetings of the International Internet Preservation Consortium (IIPC), partner meetings of the Internet Archive (Archive-It), and ten workshops held with ECDL/TPDL: International Web Archiving Workshop (IWAW), 2001-2010

Submissions (please provide contact and supporting info in <= 2 pages):

- EasyChair submission page: https://easychair.org/conferences/?conf=wadl2018
- Due: April 2, 2018
- Notifications: April 17, 2018
- Please use the ACM Proceedings template.
- **Categories:** (pick one of the three and identify it in the submission)
- 20 min. presentation + Q&A
- Poster/Demonstration + lightning talk
- 30 min. panel with interactive plenary discussion

Client-Assisted Memento Aggregation Using the Prefer Header

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

1 INTRODUCTION

Preservation of the Web ensures that future generations have a picture of how the Web was. Web archives like Internet Archive's Wayback Machine[1], WebCite[2], and archive.is[3] allow individuals to submit URIs to be archived, but the captures they preserve then reside at the archives. Traversing these captures in time as preserved by multiple archive sources (using Memento [8]) provides a more comprehensive picture of the past Web than relying on a single archive. Some content on the Web, such as content behind authentication, may be unsuitable or inaccessible for preservation by these organizations. Furthermore, this content may be inappropriate for the organizations to preserve due to reasons of privacy or exposure of personally identifiable information [4]. However, preserving this content would ensure an even-more comprehensive picture of the Web and may be useful for future historians who wish to analyze content beyond the capability or suitability of archives created to preserve the public Web.

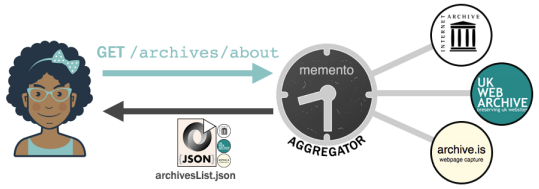
State-of-the-art Memento aggregators relay requests to a "static" set of archives. Thus, a client requesting an aggregated TimeMap has no say in which Web archives are used as the sources ($\{A_0\}$). By leveraging our previous work [4] of supplementing the capability of Memento aggregators (e.g., adding query precedence, aggregation short-circuiting, and multi-dimensional content negotiation of TimeMaps), we reuse this functionality for a more standards-based approach. This approach provides the novel contribution of involving the client's request in the Memento aggregation process beyond the specification of a URI-R and datetime.

More sophisticated aggregation may require filtering on a memento-level (e.g., only source mementos from archives with a certain quality of capture) or on a TimeMap-level. For instance, a user may wish to provide a previous unaggregated public archive (e.g., the "Freedonia Web Archive" in Figure 1b) or a private/personal Web archive as an additional source for aggregation. A conventional Memento aggregator may be required to provide additional parameters or communication flows to obtain mementos for a URI-R from private Web archives (as we discuss more in-depth in our preceding work [4]). In the current operation, a Memento aggregator assumes that all archives in a set are willing to provide a TimeMap in all instances. This may not be the case for a client's personal archive or a public Web archive that is not currently included in the aggregated set.

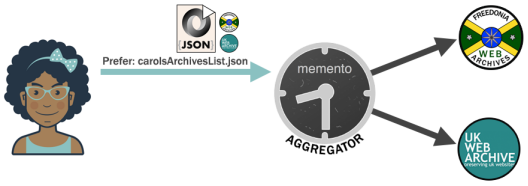
This submission represents a preliminary investigation in allowing the clients of Memento aggregators to be involved in determining the set of archives aggregated. In this work, we leverage the HTTP Prefer header [7]. Previous discussions have revolved around using Prefer for memento-level negotiation [5, 9]. This work considers using Prefer for TimeMap-level aggregation, particularly for the set of archives via archive specification instead of the representation of an individual memento.

2 BACKGROUND AND RELATED WORK

MemGator [2], the open-source Memento aggregator, provides conventional Memento aggregation with extended features including additional support for TimeMap formats beyond Link [6] and customization of the set of archives on startup of the aggregator software. CDXJ [1] is one such TimeMap format that is leveraged by the TimeMap endpoints in MemGator. Originally created as a replacement for CDX[4] files that act as an index to WARC [3] files, the CDXJ format allows for additional attributes about mementos to be specified within a JSON block. This capability allows for CDXJ-formatted TimeMaps to be much richer than Link-formatted TimeMaps due to the extensible semantics.



(a) Client requests archives list from aggregator



(b) Client supplies own list, potentially with custom attributes

Figure 1: A client first requests a list of aggregated archives from a Memento aggregator then modifies the response, encodes it, and supplies the encoded archive specification using the Prefer header for the aggregator to process.

Previously [4], we introduced the "Memento Meta-Aggregator" (MMA) concept to supplement functionality to conventional Memento aggregators using a hierarchical approach. There, we also introduced a rudimentary approach for a client to specify additional

[1] https://web.archive.org/
[2] http://www.webcitation.org/
[3] http://archive.is/

[4] https://iipc.github.io/warc-specifications/specifications/cdx-format/cdx-2015/

WADL'18, June 2018, Fort Worth, Texas USA
2018. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

```
> GET /timemap/link/http://fox.cs.vt.edu/wadl2017.html HTTP/1.1
> Host: mma.cs.odu.edu
> Prefer: archives="data:application/json;charset=utf-8;base64,Ww0KICB7...NCn0="

< HTTP/1.1 200
< content-type: application/link-format
< vary: prefer
< preference-applied: return=representation; archives="data:application/json;charset=utf-8;base64,Ww0KICB7...NCn0="
< content-location: /timemap/link/5bd...8e9/http://fox.cs.vt.edu/wadl2017.html
```

Figure 2: Client-side specification of a set of archives via encoded JSON using HTTP Prefer. The Memento aggregator responds with a location of a TimeMap for the URI-R at a URI-T representative of the set.

archives to an MMA using an ad hoc `X-Archives` HTTP request header. We also explored utilizing the Prefer [7] HTTP header to accomplish negotiation of mementos in dimensions beyond time, as may be facilitated with the usage of CDXJ TimeMaps.

Van de Sompel et al. [9] described using the Prefer header to distinguish mementos that have been rewritten when replaying Web archives to those with an untouched response body. By using Prefer header values like `original-content` and `original-headers`, a client may request that the representation return not be transformed by the Web archive.

Various presentations exist for an aggregator to use as the defining a set of archives to be aggregated, inclusive of definitions by MementoWeb.org[5], Webrecorder.io, and MemGator[6].

3 ARCHIVE SET SPECIFICATION WITH PREFER

An objective of this work is to allow a client of a Memento aggregator to be able to specify a custom set of archives ($\{A_f\}$) to be aggregated using standard syntax and semantics. We anticipate a 3-step process for a client to specify the archive set: **(1)** Client requests the set of archives to be aggregated by default from a Prefer-aware Memento aggregator (Figure 1a). **(2)** The aggregator returns the set of archives, e.g., as a JSON (per MemGator) or an XML (per mementoweb.org) file (Figure 1a), represented as $\{A_0\}$. **(3)** Once a response is received from the aggregator (e.g., https://git.io/archives), a client may manipulate the contents to be either an identical set ($\{A_f\} = \{A_0\}$), subset ($\{A_f\} \subset \{A_0\}$), supplementary set ($\{A_f\} \supset \{A_0\}$), or disjoint set ($\{A_f\} \dot{\bigcup} \{A_0\}$) (Figure 1b) and submit back to the aggregator for subsequent queries (Figure 2).

A client may also manipulate an existing archive's specification in the response received. For instance, a profiling probability (a value already defined in the MemGator specification) may be manipulated or a value of query precedence or short-circuiting may be modified, both of which we discussed in previous work [4].

Given that no Memento aggregator yet supports the client-side archive specification, we extend this idea with the assumption that a JSON response is received (like MemGator and Webrecorder's aggregator). A client may perform step 3 using the HTTP Prefer request header. After potentially manipulating the JSON response, a client would encode the JSON as a base64-encoded data URI (or supply some other URI for specification-by-reference) and submit a request with the Prefer header and a URI-R (Figure 2).

Archive supplementation may be accomplished using a hierarchical MMA approach (Figure 3), as we described in previous work [4]. This approach is necessary to adapt the capability of

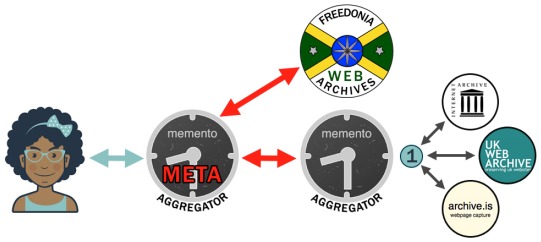


Figure 3: Using a hierarchical MMA approach, a previously unaggregated public Web archive may be aggregated with the results for a URI-R from a conventional Memento aggregator.

conventional Memento aggregators while still allowing them to be functionally cohesive. However, this hierarchical approach is insufficient if the a client would rather that subsequent queries to the aggregator after step 3 not be sent to certain archives supported by the base Memento aggregator. With the disclosure of the aggregated archives from a conventional aggregator (which is not conventionally exposed), an MMA could configure the default set from the conventional aggregator as the default to be queried and subsume the functions of the conventional aggregator.

4 FUTURE WORK

In future work we will explore this approach's interoperability with using Prefer on mementos, which is ongoing research. We will also look to additional approaches like Cookies, and usage of CoRE's *well-known* syntax for archive specification.

REFERENCES

[1] Sawood Alam. 2015. CDXJ: An Object Resource Stream Serialization Format. http://ws-dl.blogspot.com/2015/09/2015-09-10-cdxj-object-resource-stream.html. (September 2015).
[2] Sawood Alam and Michael L. Nelson. 2016. MemGator - A Portable Concurrent Memento Aggregator: Cross-Platform CLI and Server Binaries in Go. In *Proceedings of JCDL*. 243–244.
[3] ISO 28500. 2009. WARC (Web ARChive) file format. http://www.digitalpreservation.gov/formats/fdd/fdd000236.shtml. (August 2009).
[4] Mat Kelly, Michael L. Nelson, and Michele C. Weigle. 2018. A Framework for Aggregating Private and Public Web Archives. In *Proceedings of JCDL*. Accepted for Publication.
[5] Ilya Kreymer. 2018. Feedback on new implementation of Prefer header in pywb. https://github.com/mementoweb/rfc-extensions/issues/7. (March 2018).
[6] M. Nottingham. 2017. Web Linking. IETF RFC 8288. (October 2017).
[7] J. Snell. 2014. Prefer Header for HTTP. IETF RFC 7240. (June 2014).
[8] Herbert Van de Sompel, Michael Nelson, and Robert Sanderson. 2013. HTTP Framework for Time-Based Access to Resource States – Memento. IETF RFC 7089. (December 2013).
[9] Herbert Van de Sompel, Michael L. Nelson, Lyudmila Balakireva, Martin Klein, Shawn M. Jones, and Harihar Shankar. 2016. Mementos In the Raw, Take Two. http://ws-dl.blogspot.com/2016/08/2016-08-15-mementos-in-raw-take-two.html. (August 2016).

[5] http://labs.mementoweb.org/aggregator_config/archivelist.xml
[6] https://git.io/archives

A Study of Historical Short URLs in Event Collections of Tweets

Liuqing Li and Edward A. Fox
Department of Computer Science, Virginia Tech
Blacksburg, VA, USA
[liuqing,fox]@vt.edu

ABSTRACT

Since 2012 we have integrated our Web archiving efforts by collecting both tweets and webpages, using URLs in tweets to find webpages and as seeds for focused crawling. Key to this is extraction and utilization of short URLs found in tweets. Fortunately, with roughly 1,500 different tweet collections (about important events and topics), we can study the characteristics and utility of short URLs. We designed and implemented a short URL analysis system, studied the historical short URLs from a sampling of collections, and uncovered interesting results.

KEYWORDS

Data curation, events, Twitter, webpages, URL analysis

ACM Reference Format:
Liuqing Li and Edward A. Fox. 2018. A Study of Historical Short URLs in Event Collections of Tweets. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Since 2006, Twitter has grown in scale of services and impact. Users posted 50 million tweets per day in 2010 [1]; by 2018 the number has grown to 660 million [2]. While many study tweets [4, 7, 9], we have focused on the broader topic of Web archiving, integrating the collection and processing of both tweets and webpages. A key connection between these two types of data is the short URLs found in tweets. These allow a broader understanding of a tweet based on analyzing the content of the corresponding webpage(s), as well as more precise building of webpage collections based on use of the short URLs to fetch webpages directly or through focused crawling [8]. Accordingly we discuss short URLs, their characteristics, and the related webpages.

We have been collecting tweets using a variety of tools, initially yourTwapperKeeper [13]. To date we have over 2 billion tweets, about important events, trends, and topics. Many of those tweets include short URLs. Our short URL analysis system takes an event collection as input and uses Hadoop [1] and Spark [2] to extract short URLs. We expanded them, fetched the webpage with the corresponding long URL, and applied the WayBack CDX Server API [10] to attempt to restore the most likely snapshot. Then, we conducted a systematic URL analysis, for different types of events.

2 RELATED WORK

Some researchers made use of short URLs to analyze Twitter users' activity and influence [4, 9]. Other researchers detected suspicious/spam URLs through different approaches [6, 7, 11, 12].

Few researchers worked on analyzing short URLs. Antoniades et al. [3] conducted a short URL analysis, which is most relevant to our study. By focusing on two shorten URL websites and tracking the URLs, they analyzed the targeted webpages and their popularity and activity over time. However, they were limited to two services and not concerned about broken links and archives.

3 METHODOLOGY

Figure 1 gives the architecture of our short URL analysis system.

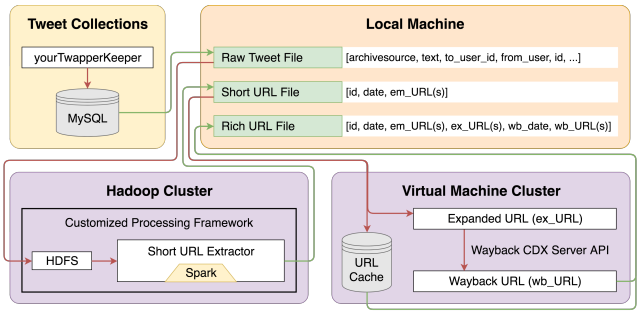


Figure 1: Historical Short URL Analysis System

We deployed yourTwapperKeeper [13] for tweet collection. For this study, we exported 12 tweet collections from MySQL. The resulting raw file has all fields stored in the database, including *archivesource*, *text*, and *id*. We uploaded the file into our Hadoop cluster, using our framework for tweet archives [5] to extract short URLs. Each record contains 3 fields: *tweet id*, *tweet posted date*, and *short URL(s)*. We expanded short URLs into expanded URLs (*ex_URLs*). Using the WayBack CDX Server API [10], we retrieved snapshots (*wb_URLs*) for some URLs from the Internet Archive [3]. We applied a URL cache to avoid duplicate processing.

4 PRELIMINARY ANALYSIS

4.1 Data Description

We choose 12 collections from 2013-2017 from our tweet archives, representing 4 categories: *Nature*, *Health*, *Man-made*, and *Particular Event*; see Table 1. The first three are general, while the fourth covers specific events. To reduce computation time, we randomly selected about 20% of the tweets. In future work we will use the full collections and run them though a pipeline of cleaning and classifying to eliminate noise.

4.2 Results

For each year, we calculate the percentage of tweets with short URLs; see Figure 2. We notice that there is no significant difference

[1] http://hadoop.apache.org/
[2] https://spark.apache.org/
[3] https://archive.org

Table 1: Different Categories of Collections

General Type	Keyword	Number of Tweets
Nature	flood	2,201,160
	hurricane	2,103,014
	typhoon	1,158,824
Health	diabetes	2,135,363
	heart attack	3,659,421
	obesity	1,249,644
Man-made	gun control	1,206,863
	gun violence	783,040
	terrorism	1,566,884
Particular Event	hurricane isaac	19,149
	hurricane sandy	385,337
	connecticut school shooting	14,141

among different categories of events. Percentages are lowest for "heart attack" and highest for "connecticut school shooting"; for that there is a great reduction from 2016 to 2017. For most collections, the peak value appears in 2015 or 2016 instead of 2017. The reason might lie in Twitter's mid-2016 decision to exclude URLs from the tweet length limit.

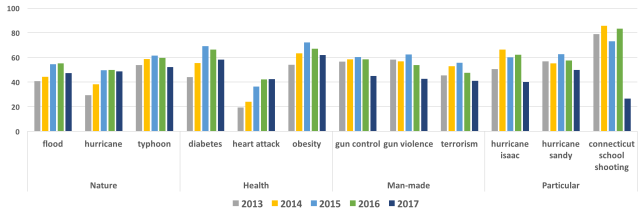


Figure 2: Percentage of tweets with short URLs over years

Figure 3 shows that recent URLs are less likely to reflect broken links, but there is less difference for "hurricane isaac". Table 2 shows the average percentage values over the years. In general, the percentage of broken URLs dropped 3%-6% year by year. The average percentage of broken links over the past 5 years is 33%.

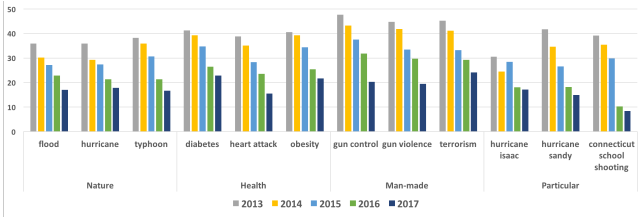


Figure 3: Percentage of broken URLs over years

The Wayback Machine allows retrieval for more URLs in former years. For the "man-made" events, more snapshots can be retrieved. Table 3, shows that the WayBack Machine provides webpages for 17.4% of the short URLs. We further split all URLs into two classes: broken and unbroken, and find older unbroken URLs are more likely to be saved. We will explore these findings with the Internet Archive as we collaborate to study global trends.

We calculate collection coverage ratio with Equation 1.

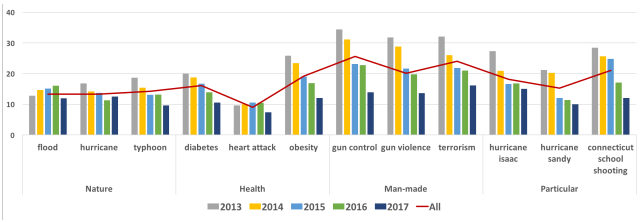


Figure 4: Percentage of retrievable URLs over years

Table 2: Average percentage of broken URLs over year

	All Years	2013	2014	2015	2016	2017
Avg %	32.9	40.9	37.2	31.9	25.7	19.5

Table 3: Average percentage of retrievable URLs over years

	All Years	2013	2014	2015	2016	2017
Avg % - all	17.4	23.2	20.7	17.3	15.9	12.0
Avg % - broken	14.8	16.4	15.9	14.0	15.7	12.4
Avg % - unbroken	18.5	27.7	23.4	18.7	15.9	11.9

$$coverage@10 = \frac{|ex\_URLs_{10} \cap wb\_URL_{10}|}{10} \quad (1)$$

Based on our 12 collections, the minimum coverage@10 is 40% while the maximum is 80%. On average, 76% of the Top-10 URLs have snapshots on Wayback Machine.

5 SUMMARY AND FUTURE WORK

We built a short URL analysis system, conducted a systematic URL analysis on different categories of collections, and observed:

- Twitter policies have made use of short URLs less important;
- The percentage of broken URLs is higher for URLs about older webpages;
- There are more broken URLs and more retrievable URLs for man-made events than for other collections;
- Older URLs are retrieved more by the Wayback Machine.

In the future, we will analyze the contents of short URLs, the correspondence between tweets and webpages, and identify high quality URLs to help further crawling and archiving.

6 ACKNOWLEDGMENTS

Thanks go to the US NSF for its support of the Coordinated, Behaviorally-Aware Recovery for Transportation and Power Disruptions (CBAR-tpd) and Global Event and Trend Archive Research (GETAR) projects, through grants CMMI-1638207 and IIS-1619028 as well as IIS-1619371 to partner Internet Archive. We would also thank the student team in CS4624 for their early work.

REFERENCES

[1] Measuring tweets. https://blog.twitter.com/official/en_us/a/2010/measuring-tweets.html, 2010. Accessed: 2018-03-24.
[2] Twitter usage statistics. http://www.internetlivestats.com/twitter-statistics/#trend, 2018. Accessed: 2018-03-24.
[3] Antoniades, D., Polakis, I., Kontaxis, G., Athanasopoulos, E., Ioannidis, S., Markatos, E. P., and Karagiannis, T. we. b: The web of short URLs. In *Proc. 20th int'l conf. on World Wide Web* (2011), ACM, pp. 715–724.

[4] Bakshy, E., Hofman, J. M., Mason, W. A., and Watts, D. J. Everyone's an influencer: quantifying influence on Twitter. In *Proceedings of the fourth ACM international conference on Web search and data mining* (2011), ACM, pp. 65–74.

[5] Bock, M. A framework for Hadoop based digital libraries of tweets. Master's thesis, Virginia Tech, 2017. http://hdl.handle.net/10919/78351.

[6] Cao, C., and Caverlee, J. Detecting spam URLs in social media via behavioral analysis. In *European Conf. on Information Retrieval* (2015), Springer, pp. 703–714.

[7] Chu, Z., Widjaja, I., and Wang, H. Detecting social spam campaigns on Twitter. In *International Conference on Applied Cryptography and Network Security* (2012), Springer, pp. 455–472.

[8] Farag, M. M., Lee, S., and Fox, E. A. Focused crawler for events. *International Journal on Digital Libraries* (2017), 1–17.

[9] Ghosh, R., Surachawala, T., and Lerman, K. Entropy-based classification of 'retweeting' activity on Twitter. *arXiv preprint arXiv:1106.0346* (2011).

[10] InternetArchive. Wayback CDX server API. https://github.com/internetarchive/wayback/tree-/master/wayback-cdx-server, 2017.

[11] Lee, S., and Kim, J. Warningbird: Detecting suspicious URLs in Twitter stream. In *NDSS* (2012), vol. 12, pp. 1–13.

[12] Maggi, F., Frossi, A., Zanero, S., Stringhini, G., Stone-Gross, B., Kruegel, C., and Vigna, G. Two years of short URLs internet measurement: security threats and countermeasures. In *proceedings of the 22nd international conference on World Wide Web* (2013), ACM, pp. 861–872.

[13] O'Brien III, J. YourTwapperKeeper. https://github.com/540co/yourTwapperKeeper, 2013.

Developing a Social Media Archive at ICPSR

Libby Hemphill
University of Michigan
Ann Arbor, MI
libbyh@umich.edu

Susan H. Leonard
University of Michigan
Ann Arbor, MI
hautanie@umich.edu

Margaret Hedstrom
University of Michigan
Ann Arbor, MI
hedstrom@umich.edu

ABSTRACT

Social media are implicated in many of contemporary society's most pressing issues, from influencing public opinion, to organizing social movements, and identifying economic trends. Increasing the capacity of researchers to understand the dynamics of such social, behavioral and economic phenomena will depend on reliable, curated, discoverable and accessible social media data. To that end, ICPSR will develop a new archive of curated datasets, workflows, and code for use by social science researchers for the empirical analysis of social media platforms, content, and user behavior. The goal is to provide a user-friendly, large-scale, next-generation data resource for researchers conducting data-intensive research using data from social media platforms such as Facebook, Twitter, Reddit, and Instagram. In our presentation, we will explain SOMAR's goals and structure and discuss opportunities for collaboration.

CCS CONCEPTS

• **Information systems → Data management systems**;

KEYWORDS

social media archiving, collection building, community building

ACM Reference Format:
Libby Hemphill, Susan H. Leonard, and Margaret Hedstrom. 2018. Developing a Social Media Archive at ICPSR. In *Proceedings of Web Archiving and Digital Libraries (WADL'18).* ACM, New York, NY, USA, Article 4, 2 pages. https://doi.org/10.475/123_4

1 INTRODUCTION

An archive for social media data will enable researchers to discover reusable social media datasets, provide a means for evaluating and/or replicating research based on social media data, and enable new insights, longitudinal studies, or comparative analyses that are nearly impossible today. Common, transparent, and reproducible approaches to privacy protection, linkage methodology, and analytical tools for these data will help ensure that research using social media data meets the highest scientific and ethical standards, and therefore gains the legitimacy necessary to advance the underlying science to its full potential.

The Social Media Archive (SOMAR) will bring together social media datasets as a corpus with associated services and resources to aid researchers in further interacting with and mining the data. This will enable extension of original findings and the creation of new knowledge, leading to a greater return on the original investment in the data. Research has shown strong and consistent evidence that data sharing, both formal and informal, increases research productivity across a wide range of publication metrics and that formal data archiving, in particular, yields the greatest returns on investment with an increased number of publications resulting when data are archived [3, 4].

We currently focus our efforts on addressing four communities of researchers: those who (1) study social media use specifically, (2) leverage social media data to understand people and society more generally, (3) study social science methods, and (4) investigate new methods for curation, publication, confidentiality and quality assessment, and long-term management of research data. One of the primary benefits of the archive is that it enables historical and longitudinal analysis. In the absence of an archive, these questions have been explored in specific, isolated, historical, social, political, and technical moments, and SOMAR's federation and long-term availability of data enables research across and between those moments.

2 TECHNICAL OVERVIEW

SOMAR will be designed to house a variety of data products related to social media research, supported by a newly developed and sustainable infrastructure. We anticipate that there will be some social media data analysis projects for which the data themselves cannot be deposited – for example, cases in which platform terms of service prohibit data sharing. As these are often exactly the instances where transparency or replicability are lacking, ICPSR will in these cases archive data workflows and code that enable users to replicate the data collection, transformation, and analysis procedures researchers' followed.

Federating data through a shared archive will result in more opportunities for comparative and historical analyses, higher quality user experience, less duplication of effort, and lower overall costs. By capturing metadata such as included hashtags and dates, SOMAR also makes it possible to generate new datasets by searching across deposits. For instance, if the same hashtag appears in multiple Twitter datasets, users could generate a dataset of those tweets even if the hashtag wasn't an original search term in any of the datasets.

Figure 2 provides an overview of the SOMAR system. There are many possible paths through the system, but the most common is likely that a user deposits data (and associated files) such as "dehydrated" data or unique identifiers of social media content (e.g., tweet ids). Often, platform terms of service dictate what data users are allowed to deposit. Twitter, for instance, allows users to share the IDs of tweets but not the tweets themselves. Common practice among researchers is then to share the list of tweet IDs included in

WADL'18, June 2018, Fort Worth, Texas USA

ACM ISBN 123-4567-24-567/08/06…$15.00
https://doi.org/10.475/123_4

a study and some link to code that would re-collect those tweets through the Twitter API [1, 5]. For instance, the Beyond the Hashtags dataset [2] contains roughly 40 million tweet ids; the website where it resides provides Python code for rehydrating the data through the Twitter API. If the data a user deposits is dehydrated, then the SOMAR system rehydrates that data by querying the platform's API (see the blue loop in Figure 1) and stores the complete data on its servers.

The curation team then uses both the data deposited and rehydrated data to create metadata enhancements (e.g., provenance, description of the platform at the time of collection, dates). Some enhancements such as expanding shortened URLs and using consistent case for hashtags and mentions [1] are straightforward and can be accomplished programmatically while others, such as disclosure risk review, require human labor.

SOMAR end users can then access data through pre-defined studies where the data they download is the same (plus metadata enhancements) as the data deposited. For instance, in the Beyond the Hashtags example, users would be able to download the list of tweet IDs or to interact with them in JupyterHub. By federating rehydrated datasets, SOMAR also enables end users to create dynamic studies by querying the entire SOMAR database and retrieving results that include data from multiple studies. For instance, they may query for all data with a certain date stamp or containing a particular set of terms and receive subsets of Researcher A's and Researcher B's studies. These dynamic studies may also include data from multiple platforms (e.g, Twitter and Reddit). End users may interact with the data through download or through JupyterHub. In this overview "data" refers to all data, documentation, code, etc.

ICPSR provides user support across the system, but most user contact occurs around deposit, download, and JupyterHub.

3 GOVERNANCE

A Steering Committee co-led by Libby Hemphill and Margaret Levenstein will set the direction of SOMAR and have final say on features and design. This steering committee will provide a mechanism for communication and governance to ensure that the needs and perspectives of the different disciplines involved in SOMAR are fully considered. We have recruited researchers from each of the scientific communities described above who regularly grapple with the challenges associated with social media data, such as managing the scale and velocity of social media data, understanding platform terms of service, and handling personally identifiable information.

4 CONCLUSION

As we begin developing each of SOMAR's components, ICPSR is interested in feedback from and collaboration with other researchers collecting, managing, and using social media data. Current opportunities include participating in our study of social media data management practices, depositing social media datasets to seed SOMAR and inform its development, and researching ways to link social media data with other data types (e.g., census, surveys) without compromising individual users.

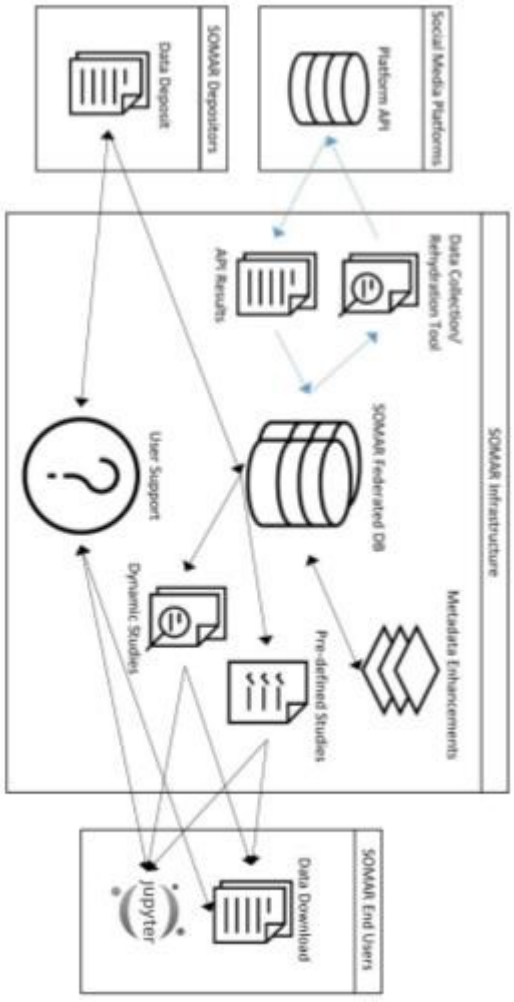


Figure 1: SOMAR System Overview.

REFERENCES

[1] Kevin Driscoll and Shawn Walker. 2014. Big Data, Big Questions| Working Within a Black Box: Transparency in the Collection and Production of Big Twitter Data. *Int. J. Commun. Syst.* 8 (2014), 20.
[2] D Freelon. 2017. Beyond the Hashtags Twitter data. (Jan. 2017).
[3] Amy M Pienta, George C Alter, and Jared A Lyle. 2010. The Enduring Value of Social Science Research: The Use and Reuse of Primary Research Data. (Nov. 2010).
[4] Heather A Piwowar, Roger S Day, and Douglas B Fridsma. 2007. Sharing detailed research data is associated with increased citation rate. *PLoS One* 2, 3 (March 2007), e308.
[5] Katrin Weller and Katharina E Kinder-Kurlanda. 2015. Uncovering the challenges in collection, sharing and documentation: The hidden data of social media research. In *Standards and practices in large-scale social media research. Oxford: International Conference on Web and Social Media.*

Swimming In A Sea Of JavaScript Or: How I Learned To Stop Worrying And Love High-Fidelity Replay

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

1 INTRODUCTION

Preserving and replaying modern web pages in high-fidelity has become an increasingly difficult task due to the increased usage of JavaScript. Reliance on server-side rewriting alone results in live-leakage and or the inability to replay a page due to the preserved JavaScript performing an action not permissible from the archive. The current state-of-the-art high-fidelity archival preservation and replay solutions rely on handcrafted client-side URL rewriting libraries specifically tailored for the archive, namely Webrecoder's and Pywb's wombat.js [12]. Web archives not utilizing client-side rewriting rely on server-side rewriting that misses URLs used in a manner not accounted for by the archive or involve client-side execution of JavaScript by the browser.

We have developed a general framework for the automatic generation of client-side rewriting libraries using the Web Interface Design Language (Web IDL) [10] that is archive and replay system independent. We provide a high-level overview of the auto-generation framework and evaluation performed that tested the auto-generated client-side rewriter's ability to augment the existing server-side rewriting system of the Internet Archive's Wayback Machine [3]. We show that client-side rewriting would both increase the replay fidelity of mementos and enable mementos that were previously unreplayable from the Internet Archive's Wayback Machine to be repayable again.

2 BACKGROUND AND RELATED WORK

Brunelle and Kelly [6] conducted a study of 1,861 URIs which had mementos in the Internet Archive between 2005 to 2012 in order identify the impact of JavaScript on the archivability of web pages. They found that JavaScript was responsible for 52.7% of all missing resources and that by 2012 JavaScript was responsible for 33.2% more missing resources than in 2005. Brunelle and Kelly [4, 5] also conducted a study that looked at the proportion of missing resources for mementos [15] in order to assess their damage, finding that the users' perception of damage to be a more accurate metric for judging archival quality than the proportion of missing resources.

Alam et al. [1] describe an additional solution for mitigating JavaScript replay issues through the usage of a ServiceWorker, which can intercept HTTP requests made by the currently replayed page and rewrite any URI-Rs to URI-Ms, client-side that were missed server-side. Lerner et al. [14] describes attacks, also launched from the live web, targeting web archives that are perpetrated by users of the web archive. The solutions posed by Lerner, namely archival modification of JavaScript at replay time and the separation of replayed content from the archive's presentation components of replay, parallel the existing replay strategies employed by Webrecorder and Perma.cc [7].

3 AUTO-GENERATION

Web IDL was created by the W3C to "describe interfaces intended to be implemented in web browser", "allow the behavior of common script objects in the web platform to be specified more readily", and "provide how interfaces described with Web IDL correspond to constructs within ECMAScript execution environments" [10]. Our framework uses the Web IDL definitions for the JavaScript APIs of the browser included in or link to by the HTML and CSS specification [8, 9] in combination with the description of how Web IDL maps to the JavaScript environment, provided by the Web IDL specification, in order to auto-generate a client-side rewriting library. This allows the generated rewriter to perform the same URL rewriting done server-side in addition to applying targeted overrides to the JavaScript APIs of the browser in order to intercept and rewrite un-rewritten URLs client-side.

We have released the generated client-side rewriter as FireFox[1] and Chrome[2] browser extensions so that others may use it to improve the replay of mementos from the Internet Archive. Note that although the generated client-side rewriter is similar to the *de-facto* implementation for client-side rewriting libraries, wombat.js, it is replay system agnostic.

4 EVALUATION

We retrieved the TimeMaps for the web pages listed in the June 2017 Alexa top 1,000,000 most visited websites and selected the first 700 pages, excluding Google and Facebook pages, that had a mememnto in the Internet Archive between June 1 and June 30. We then pre-crawled the URI-Ms using the Google Chrome browser controlled via the DevTools Protocol[3] removing URI-Ms from the frontier that redirected more than 10 times or took longer than 20 seconds for the browser to navigate to the page, resulting in 577 resolved URI-Ms. We then crawled each composite memento using the controlled browser four times, twice without client-side rewriting and twice with client-side rewriting, recording the number of requests made by the composite memento and the number of requests blocked by the Wayback Machine's content-security policy (CSP).

The crawler visited each composite memento for a maximum of 90 seconds or until network idle was determined. The determination for network-idle was calculated by keeping track of the request and response pairs for a page, and when there was only one in-flight request (no response) for 3 seconds the crawler moved to the next URI-M. Once all crawls had completed, we selected the data generated from one of the two crawls, with or without client-side rewriting, that recorded the most number of requests. We found

[1] https://addons.mozilla.org/en-US/firefox/addon/waybackplusplus/
[2] https://chrome.google.com/webstore/detail/wayback%20%20/kcpoejoblnjdkdfdnjkgcmmmkccjjhka
[3] https://chromedevtools.github.io/devtools-protocol/

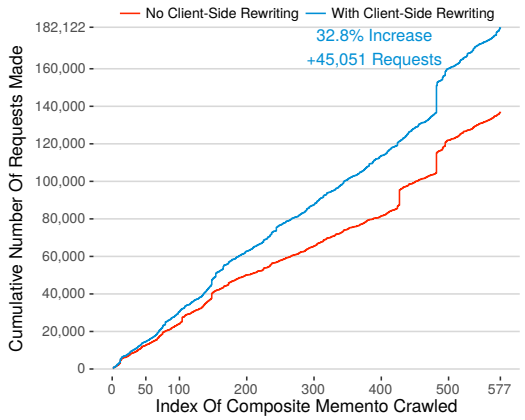


Figure 1: Cumulative number of requests made by 577 composite mementos replayed from the Internet Archive's Wayback Machine with and without client-side rewriting

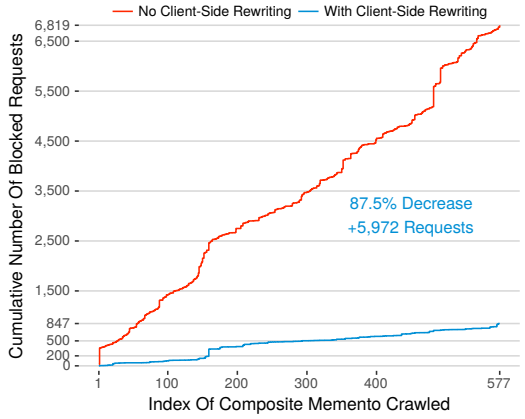


Figure 2: Cumulative number of blocked requests for 577 composite mementos replayed from the Internet Archive's Wayback Machine with and without client-side rewriting

that the composite mementos replayed with client-side rewriting made a total of 45,051 additional requests, a 32.8% increase (Figure 1) via 134,923 rewrites which occurred client-side. By crawling the mementos with client-side rewriting we were able to decrease the number of requests blocked by the CSP of the Wayback Machine by 87.5%, an increase of an additional 5,972 requests (Figure 2).

As a direct result of including the generated the client-side rewriter in the replay of the composite mementos, we were able to make composite mementos which were previously un-replayable, replayable again. The home page of cnn.com became replayable again because the generated client-side rewriter applies an override targeting the document domain issue [2]. Another notable page that became replayable again was the e-commerce site soufeel.com, which used three different ways of lazy loading its images (Figure 3).

5 CONCLUSIONS

One might believe that the usage of client-side rewriting is only limited to the most dynamic of web pages or web applications, but ensuring both high fidelity replay and the secure replay of archived JavaScript necessarily requires an archive to employ client-side rewriting. Client-side rewriting is a general solution to the increasingly difficult problems of mitigating the impact of JavaScript

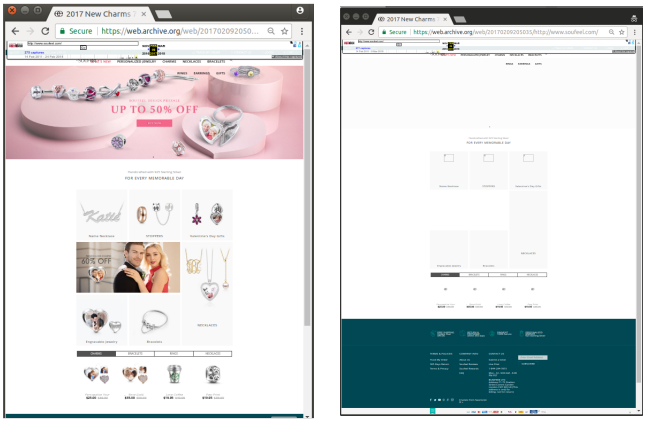


Figure 3: https://web.archive.org/web/20170209205035/http://www.soufeel.com/ increased replay fidelity from the Internet Archive's Wayback Machine with client-side rewriting

on archivability, increasing users' perception of archival quality and ensuring the secure replay of JavaScript [5, 6, 11, 13, 14].

6 ACKNOWLEDGMENTS

This work sponsored in part by NEH HK-50181 and NSF IIS 1526700.

REFERENCES

[1] Sawood Alam, Mat Kelly, Michele C. Weigle, and Michael L. Nelson. 2017. Client-side Reconstruction of Composite Mementos Using ServiceWorker. In *Proceedings of the 17th ACM/IEEE-CS Joint Conference on Digital Libraries*. 1–4.

[2] John Berlin. 2017. CNN.com has been unarchivable since November 1st, 2016. http://ws-dl.blogspot.com/2017/01/2017-01-20-cnncom-has-been-unarchivable.html. (2017).

[3] John A. Berlin. 2018. *To Relive The Web: A Framework For The Transformation And Archival Replay Of Web Pages.* Master's thesis. Old Dominion University, Department of Computer Science.

[4] Justin Brunelle, Mat Kelly, Hany SalahEldeen, Michele C. Weigle, and Michael L. Nelson. 2015. Not All Mementos Are Created Equal: Measuring the Impact of Missing Resources. *International Journal of Digital Libraries (IJDL)* (2015). https://doi.org/10.1007/s00799-015-0150-6

[5] Justin F. Brunelle, Mat Kelly, Hany SalahEldeen, Michele C. Weigle, and Michael L. Nelson. 2014. Not All Mementos Are Created Equal: Measuring the Impact of Missing Resources. In *Proceedings of ACM/IEEE Digital Libraries (DL)*. 321–330.

[6] Justin F. Brunelle, Mat Kelly, Michele C. Weigle, and Michael L. Nelson. 2015. The Impact of JavaScript on Archivability. *International Journal of Digital Libraries (IJDL)* 17 (2015). https://doi.org/10.1007/s00799-015-0140-8

[7] Jack Cushman and Ilya Kreymer. 2017. Thinking like a hacker: Security Considerations for High-Fidelity Web Archives. Presented at International Internet Preservation Consortium (IIPC) Web Archiving Conference (WAC). (June 2017).

[8] W3C Working Group. 2017. *CSS Snapshot 2017.* W3C Editor's Draft. https://www.w3.org/TR/CSS/

[9] WHATWG Working Group. 2017. *HTML Living Standard.* WHATWG Living Standard. The Web Hypertext Application Technology Working Group. https://html.spec.whatwg.org/

[10] WHATWG Working Group. 2017. *WebIDL Level 1.* W3C Recommendation. The Web Hypertext Application Technology Working Group. https://www.w3.org/TR/WebIDL-1/

[11] Mat Kelly, Justin F Brunelle, Michele C Weigle, and Michael L Nelson. 2013. On the change in archivability of websites over time. In *International Conference on Theory and Practice of Digital Libraries (TPDL)*. 35–47.

[12] Ilya Kreymer. 2018. wombat.js: Wombat JS-Rewriting Library. As apart of Pywb, https://github.com/webrecorder/pywb/blob/develop/pywb/static/wombat.js. (2018).

[13] Kalev Leetaru. 2017. Are Web Archives Failing The Modern Web: Video, Social Media, Dynamic Pages and The Mobile Web. https://www.forbes.com/sites/kalevleetaru/2017/02/24/are-web-archives-failing-the-modern-web-video-social-media-dynamic-pages-and-the-mobile-web. (2017).

[14] Ada Lerner, Tadayoshi Kohno, and Franziska Roesner. 2017. Rewriting history: Changing the archived web from the present. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. 1741–1755.

[15] H. Van de Sompel, M. Nelson, and R. Sanderson. 2013. HTTP Framework for Time-Based Access to Resource States – Memento. (12 2013). http://tools.ietf.org/rfc/rfc7089.txt RFC7089.

A Survey of Archival Replay Banners

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 surveyed various archival systems to compare and contrast different techniques used to implement an archival replay banner. We found that inline plain HTML injection is the most common approach, but prone to style conflicts. *Iframe*-based banners are also very common and while they do not have style conflicts, they suffer from screen real estate wastage and limited design choices. *Custom Elements*-based banners are promising, but due to being a new web standard, these are not yet widely deployed.

1 INTRODUCTION

Web archival replay systems express that a user is interacting with a *memento* (an archived representation of a resource at a *URI-M*) by adding an archival banner. Archival banners provide metadata about both the memento and the original resource as well as serve to distinguish a memento from its corresponding original resource. These banners may also contain various controls and toolbars to interact with the archive and the memento.

There are many ways to include an archival banner, both from the code and the user interface (UI) perspectives. An archival banner can be part of a standalone native archival application (e.g., WAIL [5]), a browser toolbar (e.g., now defunct MementoFox), or directly included in the markup of the served memento. It is the latter that is the focus of this survey, i.e., the banners that share the viewport and rendering environment with the memento (injected by the server or a client-side script/extension). Banner injection in a memento is generally obtrusive (it makes the page look different from the original) and may consume additional screen real estate. We illustrate this in Figure 1(a) by archiving `example.com` in three different archives successively, resulting in cascading banners. Not including a banner, on the contrary, loses metadata and provenance information.

2 BANNER COMPARISON

Table 1 compares three primary techniques to serve an archival banner markup with a memento used in archival replay banners of various archival systems. Below are their brief descriptions and how are they used in various archival systems.

Table 1: Comparison of different archival banner types

Features	Plain HTML	IFrame	Custom Elements
Implementation	Simple	Difficult	Intermediate
Markup rewriting	Messy	None	Clean
Compatible browsers	All	All	Modern
Isolation level	None	Document	Style
Positioning	Anywhere	Edges	Anywhere
Element overlap	Likely	Unlikely	Likely
Draggable & floatable	Possible	No	Possible
URI-M visibility	Clear	Hidden	Clear
Origin isolation	Limited	Possible	Limited

2.1 Inline Plain HTML Banners

Inline plain HTML is the simplest and most commonly used method of adding an archival banner in which necessary markup and style are injected directly in the archived HTML. While simple, it poses some issues such as vulnerability to attacks [4], conflicts with the style of the memento (as illustrated in Figure 1(b)), or hiding important elements of the page (e.g., the header of the site).

Many services such as the Internet Archive, Archive-It, and UK Web Archive (Figure 1(b)) use this method. OpenWayback, a commonly used archival replay system, supports it. The *Archive.is* banner, highlighted as number 3 in the Figure 1(a), is different from the above mentioned archives. It flattens the rendered memento markup, removes all the JavaScript, and injects it into a page that is surrounded by the banner markup. *Oldweb.today* uses a similar technique, but it utilizes server-side rendering that is emulated on a canvas element using Virtual Network Computing (VNC). Mink [10] is a Chrome extension that injects banner markup in a page on-demand (as illustrated in Figure 1(c)). It uses Shadow DOM [8] to isolate the style of the banner from the page.

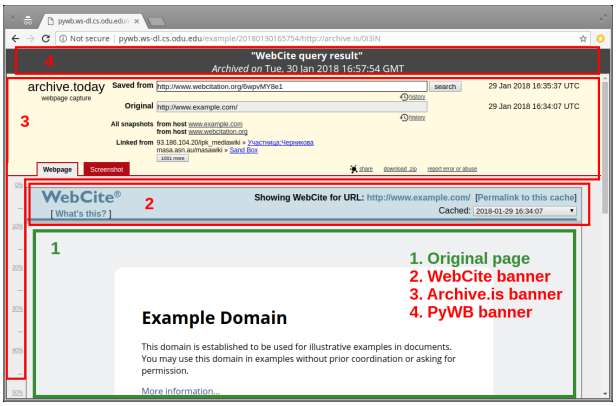
2.2 Frame/Iframe Banners

Using *frame* or *iframe* HTML elements is another common technique to provide archival banners. Iframe banners provide full document isolation, both origin and style, hence do not conflict with the position or style of the other elements of the memento. However, their positioning is not flexible enough to place them at any arbitrary location in the viewport. Since these banners must be placed clear of the memento without any overlay possibilities, often less screen real estate is available to render mementos.

This can be implemented by 1) serving both the banner and memento documents in separate frames/iframes of a parent page, or 2) making the banner document as the outer page and serving the memento inside an iframe. For example, WebCite (highlighted as number 2 in the Figure 1(a)) uses the first approach while many archives, such as the Portuguese Web Archive and National Records of Scotland, use the latter. PyWB, a popular web archival replay system, uses the latter approach by default (highlighted as number 4 in the Figure 1(a)), but allows using plain inline HTML banners.

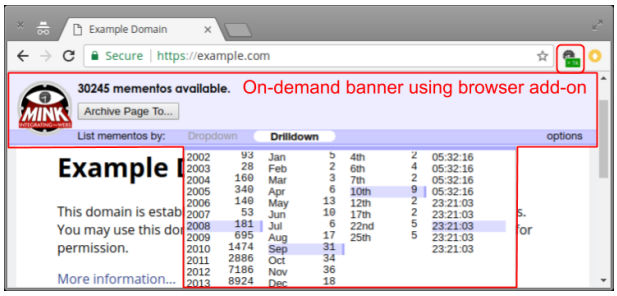
2.3 Custom Elements Banners

Custom Elements [7] is a recent web standard. It allows developers to define their own custom HTML element in JavaScript using the same capabilities that native elements have. Implementation details of the banner can be hidden, allowing a minimal and clean markup injection. By using the Shadow DOM the style is scoped to the banner, hence, there are no conflicts with other elements of the memento. This method allows both flexible design and placement choices like inline markup banners and style isolation like iframes.



(a) Three Cascading Archival Banners in a Memento

(b) Page Style Leaks into the Banner



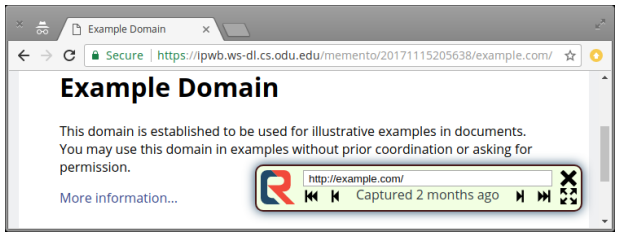
(c) Mink Banner: Injected On-demand by the Chrome Extension

Figure 1: Various Inline-and Iframe-based Archival Banners

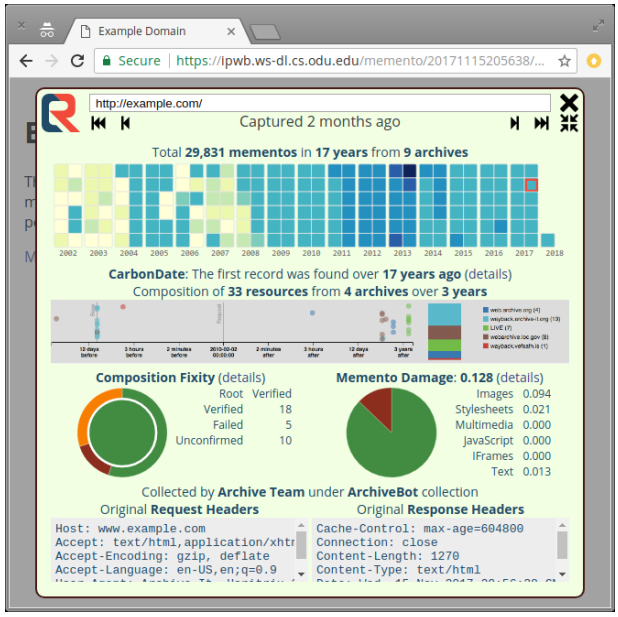
This approach is used by a banner introduced in Reconstructive [1], which is used by the IPWB [9] replay system. Reconstructive Banner [3] is an unobtrusive, interactive, responsive, and extensible multi-state archival banner. It requires minimal real estate in the *Floating Action Bar (FAB)* state (as illustrated in Figure 2(a)) and allows drag-and-drop repositioning in the viewport while hiding itself when not needed. In the on-demand *Expanded* state (as illustrated in Figure 2(b)) it provides an extensible set of interactive visualizations and provenance information that are customizable by the archive. To prevent from any live-leaks (or zombies [6]) it utilizes ServiceWorker for client-side reconstruction [2] of composite mementos.

3 CONCLUSIONS AND FUTURE WORK

We surveyed various archival systems and described different techniques used to implement an archival replay banner. Inline plain HTML injection is the most common approach used by many systems, but prone to style conflicts. *Iframe*-based banners are also used by many archival systems and while having style isolation, they suffer from screen real estate wastage and limited design choices. A more promising approach is *Custom Elements*-based banners, as used in the Reconstructive and IPWB.



(a) Draggable Floating Action Bar (FAB): Brief Information and Quick Actions



(b) Expanded: Metadata, Provenance, and Interactive Visualizations

Figure 2: Reconstructive Banner Modes

4 ACKNOWLEDGEMENTS

This work is supported in part by NSF grant III 1526700.

REFERENCES

[1] Sawood Alam. 2018. Introducing Reconstructive - An Archival Replay ServiceWorker Module. http://ws-dl.blogspot.com/2018/01/2018-01-08-introducing-reconstructive.html. (2018).
[2] Sawood Alam, Mat Kelly, Michele C. Weigle, and Michael L. Nelson. 2017. Client-side Reconstruction of Composite Mementos Using ServiceWorker. In *Proceedings of JCDL*. 237–240.
[3] Sawood Alam, Mat Kelly, Michele C. Weigle, and Michael L. Nelson. 2018. Unobtrusive and Extensible Archival Replay Banners Using Custom Elements. In *Proceedings of JCDL*.
[4] John A. Berlin. 2018. *To Relive The Web: A Framework For The Transformation And Archival Replay Of Web Pages.* Master's thesis. Old Dominion University, Department of Computer Science.
[5] John A. Berlin, Mat Kelly, Michael L. Nelson, and Michele C. Weigle. 2017. WAIL: Collection-Based Personal Web Archiving. In *Proceedings of JCDL*. 340–341.
[6] Justin F. Brunelle. 2012. Zombies in the Archives. (2012). http://ws-dl.blogspot.com/2012/10/2012-10-10-zombies-in-archives.html
[7] Domenic Denicola. 2016. Custom Elements. https://www.w3.org/TR/custom-elements/. (2016).
[8] Hayato Ito. 2017. Shadow DOM. https://www.w3.org/TR/shadow-dom/. (2017).
[9] Mat Kelly, Sawood Alam, Michael L. Nelson, and Michele C. Weigle. 2016. Inter-Planetary Wayback: Peer-To-Peer Permanence of Web Archives. In *Proceedings of TPDL*. 411–416.
[10] Mat Kelly, Michael L. Nelson, and Michele C. Weigle. 2014. Mink: Integrating the Live and Archived Web Viewing Experience Using Web Browsers and Memento. In *Proceedings of JCDL*. 469–470.

It is Hard to Compute Fixity on Archived Web Pages

Mohamed Aturban
Old Dominion University
Norfolk, Virginia 23529, USA
maturban@cs.odu.edu

Michael L. Nelson
Old Dominion University
Norfolk, Virginia 23529, USA
mln@cs.odu.edu

Michele C. Weigle
Old Dominion University
Norfolk, Virginia 23529, USA
mweigle@cs.odu.edu

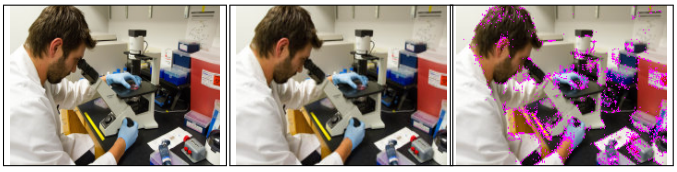
1 INTRODUCTION

Checking fixity in web archives is performed to ensure archived resources, or mementos (denoted by URI-M), have remained unaltered since when they were captured. The final report of the PREMIS Working Group [2] defines information used for fixity as "information used to verify whether an object has been altered in an undocumented or unauthorized way." The common technique for checking fixity is to generate a current hash value (i.e., a message digest or a checksum) for a file using a cryptographic hash function (e.g., SHA-256) and compare it to the hash value generated originally. If they have different hash values, then the file has been changed, either maliciously or not. We implicitly trust content delivered by web archives, but with the current trend of extended use of other public and private web archives, we should consider the question of validity of archived web pages. Most web archives do not allow users to retrieve fixity information. More importantly, even if fixity information is accessible, it is provided by the same archive delivering the content. A part of our research is dedicated to establishing and checking the fixity of archived resources with the following requirements:

- Any user can generate fixity information, not only the archive
- Fixity information can be generated on the mementos playback

2 EXAMPLES OF HOW MEMENTOS CHANGE

We have found that the HTTP entity stored in an archive change for several reasons. One example is the embedded image `http://perma-archives.org/warc/20170101182814id_/http://umich.edu/includes/image/type/gallery/id/113/name/ResearchDIL-19Aug14_DM%28136%29.jpg/width/152/height/152/mode/minfit/` in the archived page `perma-archives.org/warc/20170101182813/http://umich.edu/`. Calculating hashes on the same image downloaded at two different times produced different results as Figure 1 depicts. We used Resemble.js[1] to compare the two images pixel by pixel. The mismatched pixels are shown in Figure 1c in pink.



(A) On November 16, 2017, the hash ends in "...88c7". (B) On December 25, 2017, the hash ends in "224b". (C) Compare images (a) and (b). Mismatched pixels in pink.

FIGURE 1: The same image from perma-archives.org downloaded at two different times, produced two different hashes.

Figure 2 shows that we receive different entities for the same URI-M at different times. The memento is a stylesheet (CSS) file, and the URI-M is `http://webarchive.proni.gov.uk/raw/20150303184134/http://fonts.googleapis.com/css?family=Droid+Serif`.

Those two examples should never occur in a web archive. Add to those examples the known difficulties of client-side execution of JavaScript and network related transient error, and connection, fixity approaches for detecting tampering will produce many false positives.

```
@font-face {
  font-family: 'Droid Serif';
  font-style: normal;
  font-weight: normal;
  src: local('Droid Serif'),
       local('DroidSerif'),
       url('http://themes.googleusercontent.com/static/fonts/droidse
            rif/v2/0AKsP294HTD-nvJgucYTaJ0EAVxt0G0biEntp43Qt6E.ttf')
       format('truetype');
}
```

(A) Requesting the CSS file on November 11, 2017.

```
@font-face {
  font-family: 'Droid Serif';
  font-style: normal;
  font-weight: 400;
  src: local('Droid Serif Regular'),
       local('DroidSerif-Regular'),
       url(http://fonts.gstatic.com/s/droidserif/v7/0AKsP294HTD-nvJg
           ucYTaJ0EAVxt0G0biEntp43Qt6E.ttf)
       format('truetype');
}
```

(B) Requesting the CSS file on December 07, 2017.

FIGURE 2: Getting different content when requesting the same CSS file

3 QUANTIFYING CHANGES IN THE PLAYBACK OF MEMENTOS

We studied 18,472 mementos from 17 different web archives. We downloaded these mementos 10 times using Headless Chrome during 45 days between November 16, 2017 and December 31, 2017. The main aim of this study is to learn how the playback of these archived web pages changes during this period of time. Identifying and quantifying the types of changes present in today's archives will help us to differentiate between malicious and non-malicious changes in mementos in the future. Understanding these changes is important because conventional archival approaches regarding fixity are not applicable for web archives [1]. Table 1 shows the final number of selected mementos (URI-Ms) per archive. After downloading each memento 10 times over the 45 days, we quantified the following types of changes in the memento:

TABLE 1: The number of URI-Ms per archive. Total URI-Ms of 18,472

Archive	URI-Ms	Archive	URI-Ms
web.archive.org	1,600	archive.is	1,600
archive.bibalex.org	1,600	webarchive.loc.gov	1,600
arquivo.pt	1,600	webcitation.org	1,600
wayback.vefsafn.is	1,600	wayback.archive-it.org	1,407
swap.stanford.edu	1,233	nationalarchives.gov.uk	1,011
europarchive.org	990	webharvest.gov	733
veebiarhiiv.digar.ee	518	webarchive.proni.gov.uk	477
webarchive.org.uk	362	collectionscanada.gc.ca	359
perma-archives.org	182		

TimeMaps: Changes in TimeMaps can affect how a composite memento is constructed. The same memento might redirect differently

[1] `https://github.com/Huddle/Resemble.js`

each time it is requested (i.e., a change in the "Location" HTTP header).
HTTP entity body. Changes in the HTTP entity may occur because of dynamic content or random content generated by JavaScript.
Transient error: There are many types of transient errors. For example, web servers send back a "500" status when unable to handle the request, or an HTTP request gets a connection timeout error.
HTTP response headers: For instance, the MIME type (i.e., Content-Type Response header) of a resource might be converted (e.g., from GIF to PNG), or the server could return a "Memento-Datetime" header with a different datetime value each time.
HTTP status code: A web archive could respond with different HTTP status code when requesting the same URI-M. For example, the archive returns "404 Not Found" for a previously "200 Ok" resource because it was deleted from the server.
Other. This would include any other type of change than those mentioned above. For example, similar to HTTP entity, URI-Ms of an embedded resource of a memento may have random values generated by JavaScript code, such as values associated with the current datetime, geolocation, weather, etc.

We found that 19.48% of mementos (3, 599 out of 18, 472 URI-Ms) have changed at least one time within the 10 downloads as Table 2 shows. All archives except `archive.is` have at least one memento with a change type of "other". Similarly, all archives had some mementos experience an "entity" change, except `archive.is`, `europarchive.org`, and `stanford.edu`. The percentage of mementos with "Response headers" change does not exceed 8%. The "Transient error" change occurs in the fewest archives, but as mentioned earlier, 54% of `perma-archives`'s mementos experienced this type of change. As Figure 3 shows, all

Table 2: Number of mementos with at least one change.

Archive	URI-Ms	URI-Ms with changes (%)
web.archive.org	1,600	673 (42.06)
archive.is	1,600	6 (0.38)
archive.bibalex.org	1,600	300 (18.75)
webarchive.loc.gov	1,600	88 (0.55)
arquivo.pt	1,600	807 (50.44)
webcitation.org	1,600	365 (22.81)
wayback.vefsafn.is	1,600	378 (23.62)
wayback.archive-it.org	1,407	220 (15.64)
swap.stanford.edu	1,233	96 (7.79)
nationalarchives.gov.uk	1,011	37 (3.66)
europarchive.org	990	24 (2.42)
webharvest.gov	733	150 (20.46)
veebiarhiiv.digar.ee	518	16 (3.09)
webarchive.proni.gov.uk	477	16 (3.35)
webarchive.org.uk	362	256 (70.72)
collectionscanada.gc.ca	359	45 (12.53)
perma-archives.org	182	122 (67.03)
(total)		**3,599 (19.48)**

types of changes are noticed in mementos from `archive.org`. Only five of these mementos experience an "entity" change. About 54% (98 out of 182) mementos from `perma-archives.org` produced different hash values because of the "Transient error" (i.e., returning "5xx" HTTP status code). Approximately half of `webarchive.org.uk`'s mementos produced different hashes because of the "other" type of change. In general, transient errors and some HTTP status code changes are not unexpected, but these types of changes will make consistently computing fixity of archived resources challenging.

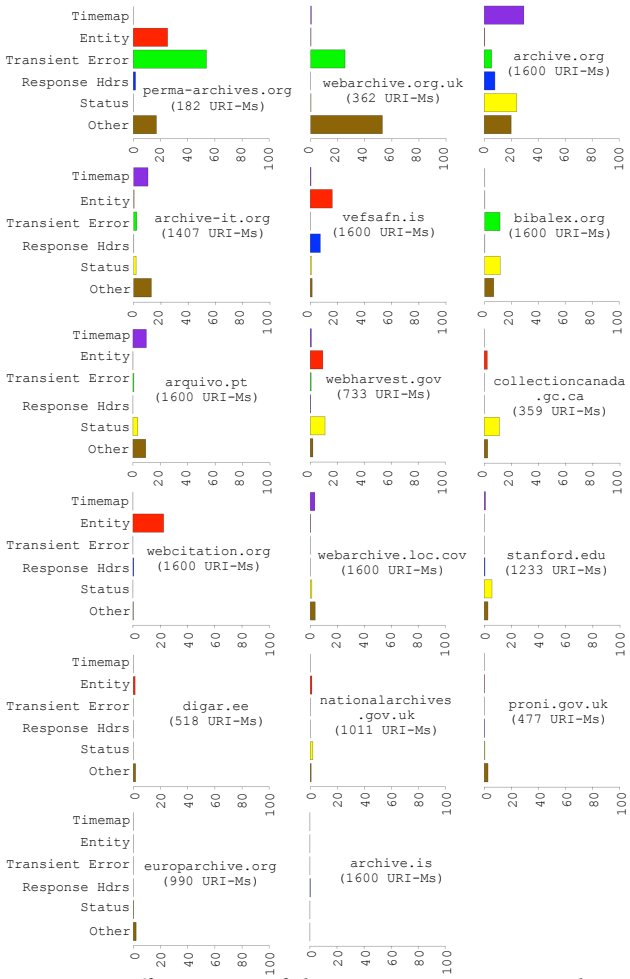


Figure 3: Different types of changes in mementos per archive.

4 CONCLUSIONS

A change in a memento may indicate malicious modification, but as we show, changes are caused by different playback-related issues. In general, we can categorize the cause of changes on the playback of mementos as: (1) expected changes, (2) unexpected non-malicious changes, and (3) unexpected malicious changes. In this article, we identify and quantify changes in the playback of mementos in general. We are currently working toward defining and quantifying each category. Being able to differentiate between malicious and non-malicious changes in mementos is important and will help us to introduce new approaches for verifying fixity of memento as conventional approaches regarding fixity are not applicable in web archives.

5 ACKNOWLEDGEMENTS

This work is supported in part by The Andrew W. Mellon Foundation (AMF) grant 11600663.

REFERENCES

[1] Jefferson Bailey. 2012. File Fixity and Digital Preservation Storage: More Results from the NDSA Storage Survey. https://blogs.loc.gov/thesignal/2012/03/file-fixity-and-digital-preservation-storage-more-results-from-the-ndsa-storage-survey/.
[2] PREMIS Working Group and others. 2005. Data dictionary for preservation metadata: final report of the PREMIS Working Group. *OCLC Online Computer Library Center & Research Libraries Group, Dublin, Ohio, USA, Final report* (2005).