

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. These archives may be used to give a more comprehensive picture of the Web as it was but they may also contain sensitive or private information. In this work we introduce a framework to mitigate these issues of aggregation in private, personal, and public Web archives without compromising potential sensitive information contained in private captures. We amend Memento syntax and semantics to allow TimeMap enrichment to account for additional attributes to be expressed inclusive of the requirements for dereferencing private Web archive captures. We provide a means of involving the user further in the negotiation of archival captures in dimensions beyond time and introduce a model for archival querying precedence and short-circuiting, as appropriate when aggregating private and personal Web archive captures with those from public Web archives through Memento. Negotiation of this sort is novel to Web archiving and allows for the more seamless aggregation of various types of Web archives to convey a more accurate picture of the past Web.

KEYWORDS

web archiving, memento, personalization, privacy

ACM Reference Format:

Mat Kelly, Michael L. Nelson, and Michele C. Weigle. 2018. A Framework for Aggregating Private and Public Web Archives. In *Proceedings of Joint Conference on Digital Libraries (JCDL'18)*. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

1 INTRODUCTION

Conventional Web archives preserve publicly available content on the live Web. Some Web archives allow users to submit URIs to be individually preserved or used as seeds for an archival crawl. However, some content on the live Web may be inaccessible (e.g., beyond the crawler's capability compared to a live Web browser) or inappropriate (e.g., requires a specific user's credentials) for these crawlers and systems to preserve. For this reason and enabled by the recent influx of personal Web archiving tools, such as WARCcreate, WAIL, and Webrecorder.io, individuals are preserving live Web content and personal Web archives are proliferating [19].

Personal and private captures, or mementos, of the Web, particularly those preserving content that requires authentication on the live Web, have potential privacy ramifications if shared or made publicly replayable after being preserved [20]. To facilitate a more

accurate picture of the Web as a whole, it may be desirable to discover and temporally aggregate these private mementos with those in public web archives. Given the privacy issues, strategically regulating access to these personal and private mementos would allow individuals to preserve, replay, and collaborate in personal Web archiving endeavors. Adding personal Web archives with privacy considerations to the aggregate view of the "Web as it was" will provide a more comprehensive picture of the Web while mitigating privacy violations.

1.1 Solutions Beyond Institutions

Personal Web archives may contain captures with personally identifiable information, such as a time sensitive statement verification Web page (Figure 1c) or a user's facebook.com feed (Figure 1a). A user may want to selectively share their facebook.com mementos [22] but wish to also regulate access to them [21]. Without the ability of authenticating as a user on the live Web, many public Web archives simply preserve the facebook.com login page (Figure 1b). Both captures are representative of facebook.com, and they may have even been captured at the same time. Users may be hesitant to share their mementos of facebook.com (or other personal or private Web pages) without a mechanism to ensure that the Web page as the user experienced it is faithfully captured and that the access of those captures can be regulated.

As a counterpoint, an individual's personal Web archive is more susceptible to disappearing without an institution's backing. Maintaining backups of archived content is unwieldy, requires diligence or automation, and is still at the mercy of hardware failures. While distributed propagation of the captures to other places may ameliorate this issue, another privacy issue remains in that distributed content may be sensitive and must be handled differently at the level of access.

To observe the more representative picture of "what I saw on the Web of the past" inclusive of private Web archive captures, we could give precedence to private Web archives over public Web archives when aggregating. For example, temporally aggregating my friends' captures (potentially residing in multiple private Web archives) with those consisting of preserved Facebook login pages (Figure 1a) from public Web archives (who are rightly not responsible for preserving my Facebook feed) may not be desirable. Instead, a user may want to instruct the aggregator to only aggregate captures from archives with certain characteristics, e.g., a set of private Web archives, and only if no personal captures are found, look to the public Web archives for captures (Figure 2). This sort of runtime specification of archival query precedence does not currently exist. Today's Memento aggregators can only query a static set of public Web archives specified at configuration time (Figure 3).

Because more personal and private, non-institutional backed Web archives are being created, and these archives may contain sensitive data that cannot be shared without special handling, more



Figure 1: Personalized and Private web pages.

work must be done to address the impermanence of personal Web archives with consideration for their contents.

1.2 Enrichment of Archival Aggregates

This work seeks to lay the groundwork for resolving multiple issues in Web archiving through the facilitation of personal, private, and public Web archives. To accomplish this, we provide amendments to the semantics of Memento TimeMaps to encourage aggregation of captures from more archives while still allowing for the distinction between conventional and enriched captures with additional metadata. We introduce additional *mementities* (a portmanteau of “Memento” and “entity”)¹ for the process of accessing Web archives of a variety of sorts to encourage concepts in Web archives like negotiation in additional dimensions beyond time, systematic aggregation of private captures, regulated access control to Web archives that may contain personal or private captures, etc.

In this work (Section 4) we introduce three new types of attributes for richer TimeMaps: **content-based attributes** based on data when a URI is dereferenced, **derived attributes** requiring further analysis beyond dereferencing but useful for evaluating capture quality, and **access attributes** that guide users and software as to requirements needed to dereference mementos in private, personal, and archives with access restrictions.

Through this TimeMap enrichment, a user will be able to specify the semantics to be selective in the set of archived URIs (URI-Ms) returned in a TimeMap through a set of attributes beyond time and original URI (URI-R). This will allow the user to interact with the

¹Used for distinction from the term “entity” as defined and liberally used in the now-deprecated RFC2616 describing HTTP/1.1.

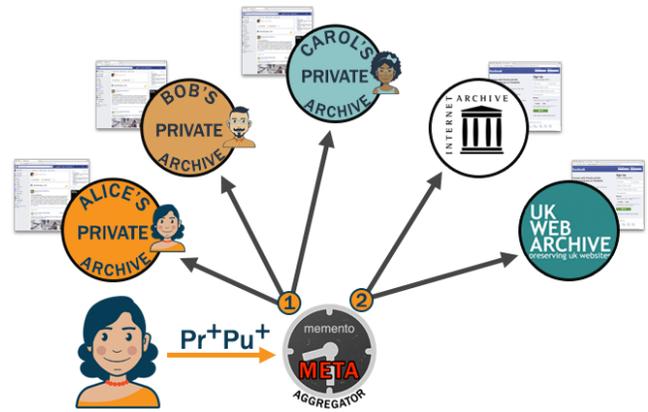


Figure 2: Archival precedence using private first then public Web archiving querying model (Pr+Pu+).



Figure 3: Conventional Memento aggregators query a set of public Web archives and do so in an equally-weighted querying model circuiting.

Memento aggregator to specify a custom subset and/or supplement the existing supporting archives in the aggregated result returned.

In this work we also describe a cascading hierarchical relationship between the aggregators and the mementities involved in aggregation and negotiation. We introduce a “Meta-Aggregation” concept (Section 5) to allow for a recursive relationship of one aggregator onto another, “building up” an aggregate result, potentially including supplemental information in the aggregation. Section 5.1 describes multiple scenarios where supplementing the memento aggregation using a meta-aggregator mementity would be useful. This hierarchy may also include other mementities like one to regulate access to private Web archives (identified with a URI-P) and another, a StarGate, to allow for selective negotiation (Section 6), e.g., allowing the client to request that only results from private Web archives are returned from an aggregator.

Aggregation in the context of this work is that of Memento-style aggregation of identifiers and not the “aggregation” of representations in the Web Architecture process used to assemble a web page. Conventional Memento aggregation (Figure 3) is accomplished by a user requesting captures for a URI-R from a remote endpoint. The software receiving the request then relays this request to a set of Web archives with which it is configured. Once the Web archives return a response containing their captures, the aggregator software temporally sorts the URI-Ms, adds additional Memento metadata, and returns the aggregated TimeMap to the user.

This work has four primary contributions to Web archiving:

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

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

Multi-dimensional user-driven content negotiation of archives: Increase user involvement in request for URI-Ms in both temporal and other dimensions (Sections 5 and 6.1).

Public/Private Web Archive Aggregation: Introduce additional special handling of access to private Web archives for Memento aggregation using OAuth (Section 6.2).

2 BACKGROUND AND RELATED WORK

2.1 Archiving and Linked Data

The Memento Framework [29] provides the constructs to interact with Web archives in the temporal dimension. Memento aggregation allows identifiers for captures (URI-Ms) from multiple Web archives to be temporally sorted using the parameters of the original live Web URI (URI-R) and a timestamp of the capture (Memento-Datetime). An archival capture (memento) is identified by a URI-M. Memento TimeMaps contain a listing of URI-Ms for mementos of an original resource on the live Web. TimeMaps also include contextual information like the URI-R that the TimeMap represents, URIs for identifiers for mementos to handle temporal content negotiation (TimeGate, i.e., URI-Gs), and identifiers for other TimeMaps (URI-Ts).

Memento TimeMaps are conventionally formatted and extend upon the Web Linking specification [24]. The syntax of the Link format applies to both information expressed in HTTP headers as well as information supplied in a TimeMap listing. Because of this, a limited set of attributes about URI-Ms is allowed within a TimeMap inclusive of `rel` and `datetime`. Additional information about a URI-M would be useful if present in a TimeMap. For example, knowing the HTTP status code of the dereferenced URI-M would reduce the amount of time needed to determine unique captures in the archive [14]. Extending TimeMaps may also provide the facility for the integration of private and public Web archives.

Alam et al. [2] defined the CDXJ format, an extension of the conventional CDX [11] archival indexing format, as an extensible means of associating additional attributes to URI-Ms. CDX files serve as indexes for Web archive files and contain many fields, like MIME-type, status code, and content-digest of the memento, which are not present in TimeMaps. MemGator [4] is an open source Memento aggregator that supports CDXJ TimeMaps (example in Figure 4) along with conventional Link-formatted and additionally JSON-formatted TimeMaps. In this work, we adapt the code for MemGator to handle additional HTTP request parameters supplied by a client as well as producing TimeMaps with the additional proposed attributes.

While these two dimensions are sufficient for the aggregation of public Web archives, additional parameters are required to express the need for privacy considerations or further steps to be executed to dereference the URI-M. Beyond the ability to express distinction in private and public mementos, it may not make sense to request public Web archive captures from an aggregator based on a variety of conditions. Some examples where expressions to distinguish captures are in isolation by URI (explicit exclusion of public captures from results) and Archival precedence (e.g., only check for captures of facebook.com in public Web archives when none are in my own).

The ‘profile’ link relation type [31] (usage discussed in Section 3) provides a standard set of semantics for processing a resource representation. We leverage and extend these semantics by allowing a user to request mementos with certain properties from Web archives.

In earlier work, we developed WARCreate [17], a Google Chrome browser extension, to allow a user to capture content from their browser, even pages behind authentication, into the standard web archiving (WARC) format. We also re-packaged institutional grade Web archiving tools in WAIL [5], a native desktop application, to allow individuals to preserve, replay, and retain complete control of their captures. More recently, the Webrecorder.io service allows similar capability, inclusive of allowing the user to capture content behind authentication, but relies on the user’s credentials being proxied through the service, a potentially undesirable feature with privacy ramifications. Our tools for private Web archiving use the WARC format for interoperability with current archival replay systems. This leaves the responsibility for maintaining privacy and further interoperability of the mementos contained within the WARCs with the archival replay system.

2.2 Privacy and Security

The Snowden Archive-in-a-Box project [18] is an autonomous version of the the Snowden Digital Surveillance Archive. The project uses a Raspberry Pi single-board computer along with other hardware and a data set containing files leaked onto the Internet by Edward Snowden to allow browsing of the files without a user being surveilled. This use case highlights access as being the problematic factor beyond the base case of the content being sensitive. When aggregating captures from a Snowden archive with captures from other archives, requesters may wish to prevent requests from propagating to other archives via the aggregator (by specifying a `privateOnly` profile) or to only consult other archives when no results are returned from their instance (request precedence, both discussed in Section 3).

The OAuth 2.0 Authorization Framework [9] (usage discussed in Section 4.3) provides a model for tokenization that we apply for persistent access to private Web archives. Using this model requires a secondary authorization server, implemented in this research as an additional mementos to decouple the authentication burden on the archive. This model, however, requires the archive to be aware of this additional mementos to act as a gateway to the archive.

2.3 Memento and HTTP Mechanics

Rosenthal [26] emphasized that temporal order may not be optimal for TimeMaps returned from Memento aggregators. He stated that aggregators need to develop ways of estimating usefulness of preserved content and conveying these estimates to readers. In a different work, Rosenthal [25] described the behavior of aggregators returning “Soft 403s” consisting of captures of login pages when the user likely expected content shown that was originally behind authentication.

Rosenthal [25] also described a “hints list” that an aggregator might provide based on its own experience of requesting content from archives. In this work Rosenthal also alluded to a hypothetical mechanism of the aggregator filtering content like login pages from

```

@context ["http://tools.ietf.org/html/rfc7089"]
@id {"uri": "http://localhost:1208/timemap/cdxj/http://facebook.com"}
@keys ["memento_datetime_YYYYMMDDhhmmss"]
@meta {"original_uri": "http://facebook.com"}
@meta {"timegate_uri": "http://localhost:1208/timegate/http://facebook.com"}
@meta {"timemap_uri": {"link_format": "http://localhost:1208/timemap/link/http://facebook.com", "json_format":
  ↳ "http://localhost:1208/timemap/json/http://facebook.com", "cdxj_format":
  ↳ "http://localhost:1208/timemap/cdxj/http://facebook.com"}}
19981212013921 {"uri": "http://archive.is/19981212013921/http://facebook.com/", "rel": "first memento", "datetime":
  ↳ "Sat, 12 Dec 1998 01:39:21 GMT"}
19981212013921 {"uri": "http://web.archive.org/web/19981212013921/http://facebook.com/", "rel": "memento", "datetime":
  ↳ "Sat, 12 Dec 1998 01:39:21 GMT"}
19981212024839 {"uri": "http://web.archive.org/web/19981212024839/http://www.facebook.com/", "rel": "memento",
  ↳ "datetime": "Sat, 12 Dec 1998 02:48:39 GMT"}
...
20170330231113 {"uri": "http://web.archive.org/web/20170330231113/http://www.facebook.com/", "rel": "memento",
  ↳ "datetime": "Thu, 30 Mar 2017 23:11:13 GMT"}
20170331013527 {"uri": "http://web.archive.org/web/20170331013527/https://www.facebook.com/", "rel": "last memento",
  ↳ "datetime": "Fri, 31 Mar 2017 01:35:27 GMT"}

```

Figure 4: An abbreviated CDXJ TimeMap from MemGator for facebook.com.

the results and redirecting a user to a version of the TimeMap containing only captures that are not a login page.

Jones et al. [12, 30] discussed obtaining the “raw mementos” consisting of un-rewritten links in captures in a systematic way using the HTTP Link response header. By utilizing the HTTP Prefer request header [28], a user would be able to obtain a version of the memento as it appeared at the time of capture instead of a version with relative links rewritten by the archive to point back within the archive and not the live Web. An archive, in response and to confirm compliance with the request, would return the memento with the HTTP Preference-Applied response header along with the requested original version of the memento.

The HTTP Prefer header [28] allows an explicit means for a client to express their preferences of optional aspects in an HTTP Request. Van de Sompel et al. [30] highlighted that the Prefer header could be used by Web archives to allow clients to specify a request for the unaltered or un-rewritten content. Rosenthal [27] echoed Van de Sompel et al. by suggesting a list of transformations (screenshot, altered-dom, etc.) for a memento via a new HTTP header.

This work focuses on the transformation of TimeMaps, not the mementos themselves. The rewriting problem in previous work is pertinent to replay of URI-Ms whereas what we accomplish is more expressive metadata of the mementos prior to and to mitigate issues with dereferencing URI-Ms. A goal of this work is to further involve the client in the aggregation process. Interaction with the aggregators through these sort of mechanisms will be a first step in accomplishing this, as described further in this work.

Fielding and Reschke [7] defined proactive and reactive content negotiation as that which is determined by the server as a best-guess based on metadata and a model involving communication and selection of representations, respectively. The latter may be accomplished a variety of ways inclusive of the utilization of the HTTP 300 Multiple Choices and 406 Not Acceptable status codes as well as the less commonly implemented HTTP Alternates header and Transparent Content Negotiation [10]. As we anticipate generating derivative TimeMaps consisting of any number of permutations of additional attributes applied on the mementos, it would be useful to associate and allow users to choose the variant they prefer using these status codes and HTTP transaction patterns.

In previous work [13], we highlighted an issue of URI-collision in the realm of personal Web archives wherein (for example) both a

login page and the authenticated content of a live Web application may reside at the same URI-R (Figure 1). We [15] extended this work by identifying personalized representations of mementos and providing a mechanism to navigate between additional dimensions beyond time. As personal Web archives proliferate and are at some point aggregated into multi-archive TimeMaps (cf. a TimeMap from and containing only listings from the archive itself), it would be useful to distinguish URI-Ms that represent personalized mementos, mementos that were originally behind authentication, and mementos in personal web archives that require additional considerations and mechanisms to access.

3 ARCHIVE QUERY PRECEDENCE AND SHORT-CIRCUITING

Private Web archives contain an inherent characteristic where exposing the metadata about an archive’s contents could be sufficient to identify the archive’s contents. For example, a private archive responding with a TimeMap containing URI-Ms for captures of my online bank statement would reveal that I am preserving personal banking information (or with fewer ramifications but still a need for privacy, a site with embarrassing photos).

A second aspect exists independent of exposing the metadata that may reveal a private Web archive’s contents. Were a client to setup a Memento aggregator inclusive of their private Web archive, they may prefer a mechanism that returns the results only from their private archive if it contains contents for a given URI-R and only default to sending the request to public Web archives if no results were returned. The set of archives queried may have a tiered request configuration with requests being performed in a more synchronous procedure with the aforementioned short-circuiting procedure applied.

Figure 2 illustrates requests being first sent to the private archives then to public Web archives. It may also be desirable to allow this behavior to functionally coexist with conventional pipelined asynchronous archive querying. As with the Snowden Archive-in-a-box [18] example in Section 2, access to this content if checking for the existence for captures in other archives may imply interest or association with the subject matter, in some cases itself being revealing or even incriminating.

For a Memento aggregator to include contents or simply metadata from the Snowden archive along with other personal, private,

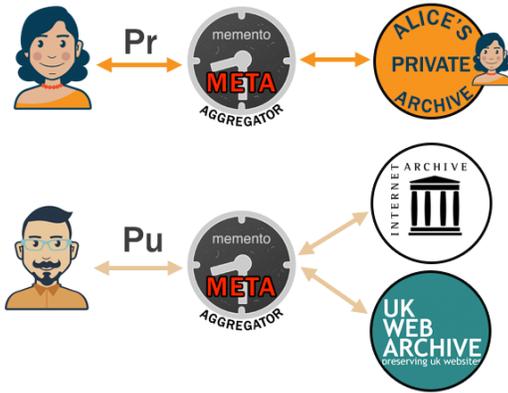


Figure 5: PrivateOnly and PublicOnly aggregation in an MMA.

and public captures would require special handling to be considered when accessing resource from the Snowden archive. For example, a user may want requests for a certain set of URI-Rs to not also be requested from other private Web archives beyond the Snowden archive or their own personal Web archive for the sake of privacy of the request.

We propose two initial approaches to accomplish this: explicit specification by a client at the time of request and analysis of mementos with a potentially personalized representation. For the latter, we [15] identified three methods for identifying personalized representations. Of the methods proposed, one not investigated further (we opted for one of the other three) was to be able to specify additional environment variables when selecting a representation of a resource. The downside, we mentioned, was the requirement of a specialized client. The specialized “client” in this case may be the mementity responsible for determining the degree of personalization of the representation, i.e., the StarGate.

When aggregating and replaying a URI-R over time from a set of archives consisting of captures from both public and private Web archives, it may be desirable to first check for private captures prior to requesting URI-Ms from public Web archives (Figure 2). For example, in aggregating URI-Ms for `facebook.com` that include mementos of my news feed from my private archive and unauthenticated login pages from institutional public Web archives (Figure 1), the latter is less useful in observing how the page has changed over time. To maintain relevancy of the desired sort of representation, we may want to check for the existence of captures from private Web archives *first* and then, only if none are present, resort to requesting the captures consisting of a login page. This model of precedence (request priority) and short-circuiting (stop requesting captures if a condition is met) via Memento aggregators does not currently exist but could be critical in a user expressing what they expect from an aggregator beyond simply mementos for a URI-R.

In the basic model below, we express various access precedence models (henceforth *profile*) for containing boolean categorization of private and public Web archives. In each profile, order is significant and thus a simple regular expression can be used where P_u symbolizes a public Web archive endpoint, P_r a private Web archive endpoint, and the “+” superscript indicating at least one or more consecutive instances.

$$noArchives \rightarrow \emptyset \rightarrow \{\} \tag{1}$$

$$publicOnly \rightarrow P_u^+ \tag{2}$$

$$privateOnly \rightarrow P_r^+ \tag{3}$$

$$privateFirst \rightarrow P_r^+ P_u^+ \tag{4}$$

$$publicFirst \rightarrow P_u^+ P_r^+ \tag{5}$$

The basic profiles pair with the syntax of the profile relation type [31], allowing clients to request resulting TimeMaps containing URI-Ms from a subset of archives from which the Memento mementity requests (Figure 5). The preliminary scheme for short-circuiting of subsequent requests is also boolean, e.g., requests should only be made to public Web archives when the *privateFirst* profile (Equation 4) is specified by the client when no identifiers for captures are returned from private archives. This model also assumes that the sets P_u and P_r are disjoint ($P_u \cap P_r = \emptyset$) for simplicity but may not be the case in reality. For Web archives that contain both private and public captures, an approach toward achieving mutually exclusivity could be to separate each set of the private and public URI-Rs into an abstraction of separate collections. For example, as discussed earlier, the UK Web Archive contains captures from its legal deposit with restricted off-site access; that is, a user cannot access the mementos unless physically on location at the library (Figure 6).

4 ADDITIONAL TIMEMAP ATTRIBUTES

Aggregating private and public mementos requires the ability to distinguish captures that require special handling. To accomplish this and to provide the ability for TimeMaps to be more descriptive of the URI-Ms they contain, we extend on the TimeMap syntax and semantics to allow additional attributes, contained in the TimeMap, to be associated with the URI-Ms. In this work we introduce three new types of attributes for richer TimeMaps: **content-based attributes** based on data when dereferenced, **derived attributes** requiring further analysis beyond dereferencing but useful for evaluating capture quality, and **access attributes** that guide users and software as to requirements needed to dereference mementos in private, personal, and archives with access restrictions. In this work we focus primarily on the access attributes but define other classes of attributes for future extensibility.

4.1 Content-based Attributes

In previous work [14], we highlighted that the URI-Ms in a TimeMap for `google.com` produce nearly 85% HTTP redirects when dereferenced. Determining how many mementos exist from an archive for a URI-R is thereby impossible from a TimeMap alone. Enriching a TimeMap with information about the dereferenced capture would improve methods for determining how well (in quantity) a URI-R has been captured. HTTP data obtained when dereferencing a URI-M like status code [7], content-type [7], and Last-Modified [1] would be useful. For some of these attributes (like the aforementioned), the data may exist in the archival indexes, typically formatted as CDX. However, while many Web archives expose a Memento endpoint, few make these additional content-based attributes about the captures available through a CDX server². Thus,

²Internet Archive does currently expose a CDX endpoint with limited fields at <http://web.archive.org/cdx/search/cdx?url=example.com>

```

$ curl -I https://www.webarchive.org.uk/wayback/archive/*/http://www.example.org
HTTP/1.1 451 Unavailable For Legal Reasons
Date: Wed, 25 Oct 2017 04:39:35 GMT
Server: Apache-Coyote/1.1
Content-Type: text/html;charset=utf-8
Transfer-Encoding: chunked
Set-Cookie: JSESSIONID=823BD09DF8DD489087763640A8150023; Path=/; HttpOnly
Content-Language: en

```

Figure 6: Accessing a URI-M at UKWA using curl returns an HTTP 451 status code.

once these attributes are discovered by dereferencing, they may be retained and expressed with the assumption that they are an accurate account of the archival record.

4.2 Derived Attributes

Other attributes about a memento may require calculation to obtain, which can be computationally and temporally expensive when performed at archival or even URI-R scale. This section briefly describes one such example of a derived attribute: Memento Damage calculation. Adding the ability for this derived attribute to be present in a TimeMap would allow for more efficient evaluation of memento quality. Brunelle et al. [6] developed a metric for determining the quality of a capture (cf. quantity in Section 4.1) when dereferencing a URI-M with a particular focus on the quantitative significance of missing embedded resources. Determining “Memento Damage” requires calculation beyond simple counting, as all resources are not equally weighted, particularly when absent. Having this information calculated and present in a TimeMap would allow a user to select the best or most complete URI-M without needing to iterate through all URI-Ms.

4.3 Access Attributes

An impetus for this research is integrating private and personal Web archives through aggregation via Memento TimeMaps. CDXJ allows additional attributes to be specified and considered when a URI-M is dereferenced. Figure 8 shows how a token may be stored in an enriched CDXJ TimeMap where the authentication procedure is discoverable at runtime. We utilize OAuth2 [9] for authorization when dereferencing URI-Ms with this field for tokenization for persistent access to private mementos.

Access control may be needed in cases where private and personal Web archives are aggregated with public Web archives via TimeMaps. An authentication procedure and subsequent tokenization will allow persistent access using a token derived from authenticating. Figure 8 shows a token being attributed on a per-URI-M basis, though a single token may be applied to all URI-Ms returned from an archive. The responsibility for attributing the token to an individual or set of mementos may lie in either the archive itself or the aggregator in this preliminary model.

4.4 Sources of Derivatives

CDXJ allows metadata fields (lines beginning with @meta) about the TimeMap to precede the listing of captures. Figure 4 contains metadata fields within a CDXJ TimeMap that are typically also found in a Link-formatted TimeMap, e.g., URI-R for the original resource, TimeGates, and other related TimeMaps. With the introduction of derived attributes (Section 4.2), it is critical to not just give context as to the semantics of new attributes like “damage” but also to provide guidance in generating this value.

Figure 7 provides an example where a derived attribute requiring calculation (memento damage [6]) and an access attribute are defined for guidance within the TimeMap. Definitions in the extended_attributes metadata field serve as templates as applied to URI-Ms in the TimeMap when present and applicable. The “service” rel value (inspired by Atom [8]), for instance, instructs parsers to look to the the URI specified in the template and the URI-M itself to obtain a value for this attribute. The “access” attribute is given a contextual definition using a rel value of “self” via [23] wherein the expectation is for a parser to look to the URI-M where the access attribute exists for resolution. The “via” rel value [23] for each of these attributes instructs parsers to look to the respective identifier for the source of the information for the links context; “self” for the access attribute and the service defined by the URI in the “service” attribute for the JSON block.

5 MEMENTO META-AGGREGATOR

In this work we extend on the role of a Memento aggregator to possess additional capabilities when interacting with Web archive users, Web archives, other Memento aggregators, and other mementities. MemGator [4] allows a user to host a Memento aggregator at a location of their choosing (inclusive of the user’s local machine) and configure a set of Web archives to query when starting the software. While conventional, remotely located aggregators assume that all Web archives queried are publicly accessible, a locally hosted, customizable aggregator may interface with archives that have restricted access. For example, a MemGator instance may request mementos from a Web archive that is only accessible on the user’s local area network or co-hosted on the user’s machine on which the aggregator resides. Non-public archives are treated and aggregated agnostically without further consideration of their holdings. In this section we describe a mementity to account for the shortcomings of conventional Memento aggregators while also extending their standard functionality and Memento interfaces.

A Memento Meta-Aggregator (MMA) serves as a functional superset of a conventional Memento Aggregator (MA), along with adding functionality outside of the scope of a conventional MA. A conventional MA provides access through identifiers to mementos (URI-Ms), TimeGates (URI-Gs), and TimeMaps (URI-Ts) from a set of Web archives. An MMA provides the ability to both supplement (Figure 10) and selectively filter the results returned from an MA with URI-Ms from additional Web archives at the request of the user or as configured with the MMA (Figure 9). In a proof-of-concept of this work, we build upon the open source MemGator to introduce these additional roles outside of the scope of the Memento aggregator mementity type to define the MMA mementity type.

Figure 10 describes a sample hierarchical relationship of mementities consisting of MMAs, MAs, and Web archive (WAs). When MA₁ receives a request for URI-Ms for a URI-R, for instance, the

```
@meta {"extended_attributes": {
  "damage": {"rel="service via", "service": "http://memento-damage.cs.odu.edu/?uri={uri}", type="float"},
  "access": {"rel="self via", "token": "self", type="string"}
}}
```

Figure 7: Additional metadata atop a TimeMap provides guidance to both the user and generation tools to produce derived attributes for URI-Ms in a TimeMap.

```
19981212013921 {
  "uri": "http://localhost:8080/20101116060516/http://facebook.com/",
  "rel": "memento",
  "datetime": "Tue, 16 Nov 2010 06:05:16 GMT",
  "status_code": 200,
  "damage": 0.24,
  "access": {
    "type": "Blake2b"
    "token": "c6ed419e74907d220c69858614d8669ff3732df0cc5647ef0a3a88a41..."
  }
}
```

Figure 8: An amended CDXJ record for a private capture of facebook.com. Line breaks added for readability.

<p>A = Alice's archive B = Bob's archive C = Carol's archive I = Internet Archive R = URI-R MMA_X = Set of archives sourced for X's MMA for R MA = Memento aggregator at mementoweb.org</p>

$$\begin{aligned}
 \text{MMA}_{\text{Alice}} &= \begin{cases} \{A, B, C\}, & \text{"facebook.com"} \in R \\ \{A, C\}, & \text{"alicesembarassingphotos.net/vacation.html"} \in R \\ \{A, B, C, I\}, & \text{otherwise} \end{cases} \\
 \text{MMA}_{\text{Bob}} &= \begin{cases} \{B, A\} \\ \{C\}, & \text{"carolsembarassingphotos.net"} \in R \\ \{\text{MMA}_{\text{Alice}}, \text{MA}\}, & \text{otherwise} \end{cases} \\
 \text{MMA}_{\text{Carol}} &= \begin{cases} \{C\}, & \text{"carolsembarassingphotos.net"} \in R \\ \{\text{MMA}_{\text{Alice}}, \text{MA}\}, & \text{otherwise} \end{cases}
 \end{aligned}$$

Figure 9: A Memento Meta-Aggregator is configured to perform selective aggregation.

request is relayed to WA₁, WA₂, and WA₃ for the sets of mementos {a₁m₁, a₁m₂}, {a₂m₁, a₂m₂, a₂m₃}, and {a₃m₁, a₃m₂}, respectively. MA₁ is then responsible for combining and temporally sort the URI-Ms then returning the aggregated TimeMap to the requesting user (or mementity). The temporal ordering within an archive corresponds to the second index (*m*) for convenience in the figure, however, this ordering may not hold between archives. For example, a₂m₂ is older than a₃m₁ per the temporal ordering diagram on the right side of the figure. The ordering for the mementos contained within the configured archives as requested from various mementities is displayed in the bottom portion of the figure. This figure also shows examples of an MMA obtaining results from multiple MAs (e.g., MMA_α from MA₁ and MA₂) and even MMAs referring to other MMAs for their results when queried (e.g., MMA_γ referring to MA₁, WA₅, and MMA_β with the latter referring to WA₇ and WA₈). The configuration of MMA_β is similar to the relationship of MMA_{Carol} to MMA_{Alice} in Figure 9 where a user may configure an MMA to both refer to a custom set of sources for results as well as reuse the in-place selective filtering of the sources. In this case, MMA_{Carol} would inherit the restriction of MMA_{Alice} of not sending requests for mementos of http://alicesembarassingphotos.net/vacation.html to Bob's archive.

Results from other Web archives that are aggregated with the results from an MA may be public non-aggregated Memento-compliant Web archives or private Web archives. A conventional MA is not required for the functionality of an MMA. An MMA may serve as a functional replacement for an MA at a fundamental level, that is, the aggregation of a static set of public Web archives may be

performed by an MMA in a black box manner as if the MMA were identically configured with the same archives as the MA.

An MMA can be configured to return an aggregated TimeMap based on a set of Web archives for which it has been configured or it may be provided a set of archives to query upon request from a client. This abstraction provides a level of extensibility to current Memento aggregators for which the additional functionality may not be appropriate, scalable, or interoperable; however, providing an on-demand set of archives to query is useful in the context of personal Web archiving.

User-driven specification of aggregation parameters is particularly important for accessing personal Web archives using a Memento aggregator. If a user requests a TimeMap from a conventional Memento aggregator (Figure 3), the aggregator will request the URI-Ms from each archive with which the aggregator is configured to communicate. A user may wish to customize, prioritize, or give precedence to the archives queried (as described in Section 3). If a user hosts an aggregator themselves, the aggregator would need to be reconfigured to prevent requests for certain URI-Rs from propagating to certain archives on the basis of URI-R-archive pairs. Though this may become unwieldy, what follows is a useful example to illustrate where configuring an MMA with a core ruleset prior to considering further user-driven specification would be useful when aggregating personal and public Web archives.

5.1 MMA Archive Selection

Figure 9 abstracts the following scenario to show how an MMA can perform selective aggregation. Alice archives Web pages she views in her browser using WARCreate [17] and replays them using her local Wayback instance within WAIL [5]. Bob, Alice's acquaintance, and Carol, Alice's sister, each do the same for their own captures. Alice sets up an MMA (MMA_{Alice}) that is configured to request captures from her archive (A), Bob's archive (B), Carol's archive (C), and the Internet Archive (I). For some URI-Rs, like facebook.com, it may not make sense to aggregate Alice, Bob, and Carol's captures with those from Internet Archive, so she can specify a rule of only aggregating mementos from {A, B, C} when those URI-Rs are requested³. For other URI-Rs, like alicesembarassingphotos.net,

³Note that MMAs do not protect the contents of an archive from being viewed, which is handled by the mementity described in Section 6.2.

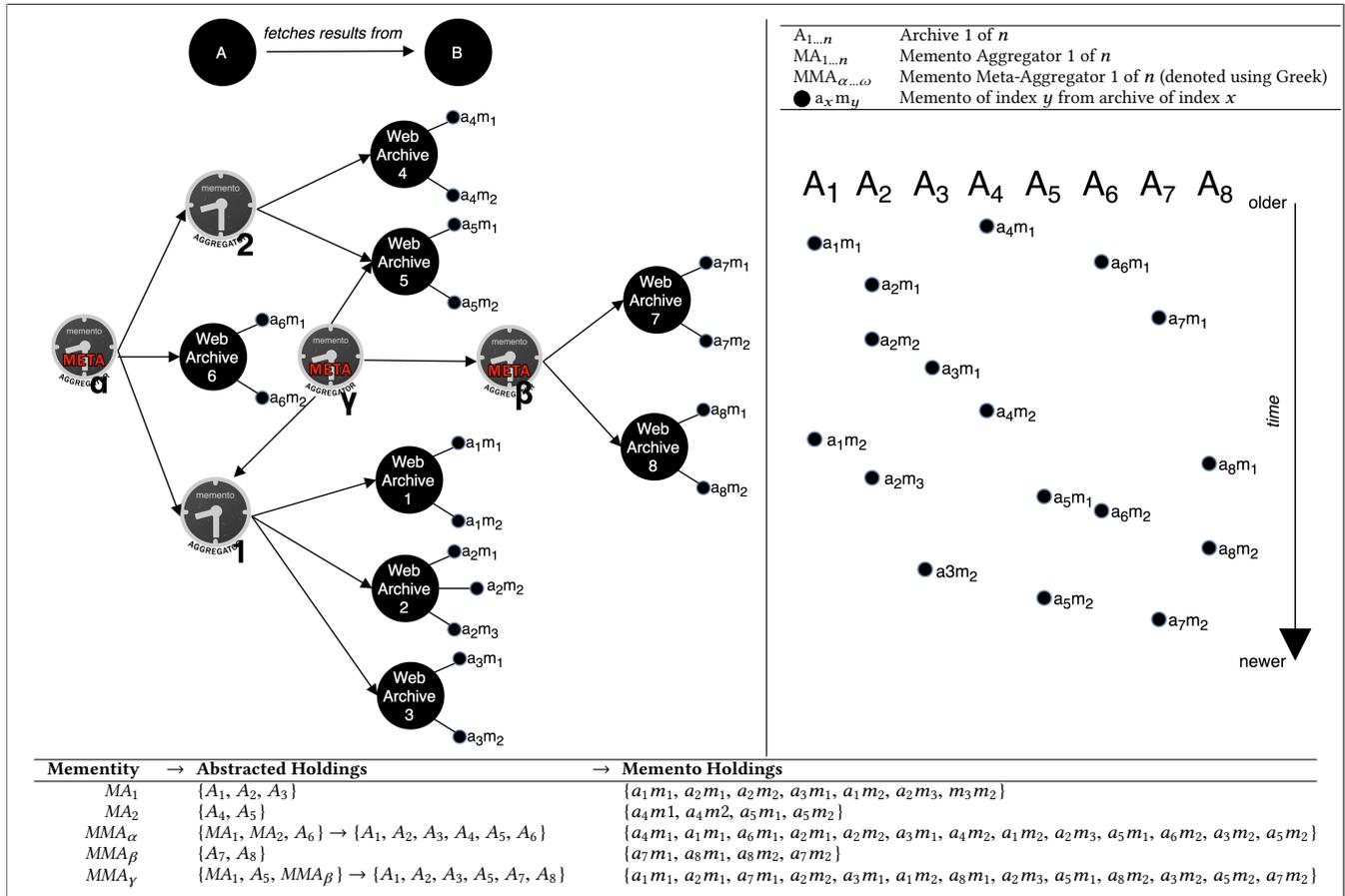


Figure 10: Memento Meta-Aggregators may aggregate URI-Ms from archives, Memento aggregators, and other MMAs equivalently. Shown is an example of temporally sorted captures as served from an MMA in a variety of permutations in a potentially ad hoc hierarchy.

Alice may want to prevent exposing the fact that she is looking for certain old captures to Bob and the Internet Archive, but wants to also aggregate captures from Carol’s archive, with whom she does not mind exposing the URI-Rs requested. She does this by creating another rule to only aggregate from archives $\{A,C\}$ in those cases. By Alice controlling the MMA, she can both pre-configure the set of potential archives queried as well as provide the ability for her, Bob, or Carol to selectively aggregate from the set of archives when requesting captures for a URI-R. Were Bob uncomfortable with his aggregation requests going to Carol’s archive when he used Alice’s MMA, he may set up his own MMA (MMA_{Bob}) to request captures from only his and Alice’s archives without a URI-R filtering scheme like Alice’s MMA. Carol also sets up an MMA (MMA_{Carol}) that defaults to using Alice’s MMA and the mementoweb.org MA except when requesting URI-Rs from carolsembarrassingphotos.net.

As an endpoint, MMAs may aggregate and request access to captures to private Web archives using a token-based authorization model (e.g., using OAuth [9] as described further in Section 6.2). The query may be subsequently routed to an applicable and corresponding Web archive (private or public) after authentication has been established. MMAs may query other MMAs with the expectation that the results returned will be consistent with those from an MA

with additional indicators for content beyond the scope of an MA (e.g., a flag for content from a non-aggregated or public archive). In the scenario above, Carol may want additional archives aggregated beyond the default case in Figure 9 so she can utilize the ruleset of Alice’s MMA, as well as add filtering rules of her own. The filtering that an MMA performs may not be (and more likely is not) exposed to clients or other MMAs that look to it as a source for URI-Ms. Doing so would be a detriment to the function of an MMA preventing selective aggregation, though does not prevent clients from accessing the aggregated archives directly. Note that in the case of Carol’s MMA, there exists a redundancy in that both Alice’s MMA and the mementoweb.org MA will request URI-Ms from IA. While Carol’s MMA may perform an operation to consolidate duplicates (i.e., a “UNIQUE” operation), time may still be wasted waiting for all archived sources to respond to requests to Carol’s MMA. Carol may also only want to look to some archives if none, too few, or some other quantifier or qualifier exists in an initial set or series of archives. For advanced querying of this sort, a separate mementity exists and is described in Section 6.

5.2 User-driven Archival Specification

As in the scenario described above, a user may wish to include additional archives in the aggregation process or specify the exclusion of URI-Ms from specific archives at the time of the request. The MMA mementity type allows a user to be more descriptive in the results they would like returned compared to a conventional Memento aggregator where only a URI-R and a datetime are specified. Introducing a separate mementity instead of assigning additional roles to the existing Memento aggregator concept provides extensibility while retaining the semantic responsibilities of conventional aggregators and reusing existing infrastructure.

For a user to be able to express additional archives to be aggregated at run time requires both cooperation of the client and recipient to communicate through the same “protocol”. We accomplish this using a fabricated X-More-Archives HTTP request header, that is consumed by a modified MemGator (serving as an MMA) to supplement the list of archives to be queried (see curl command below). Additional attributes may be specified to the MMA, for instance, if the newly supplied archive requires special handling.

```
curl -H "X-More-Archives:
  → http://myLocalWebArchive/myCollection/timemap/*/"
  "http://mmaHost/timemap/json/http://www.themaneater.com"
```

6 STARGATE

Memento TimeGates generally accept a URI-R and a datetime (through the Accept-Datetime HTTP header [29]) and redirect to a URI-M in return. This work introduces negotiation with a mementity that serves as an extension to a TimeGate, a StarGate⁴. A StarGate extends the functionality of a TimeGate with additional content negotiation on other dimensions, such as those described in Section 4. This broadens archival negotiation beyond the temporal dimension into a range of others. A StarGate also acts as an endpoint to enrich a TimeMap with additional attributes about URI-Ms.

6.1 Negotiation in Other Dimensions

Content negotiation in other dimensions may be accomplished in a variety of ways in HTTP, each with their own dynamics. The Prefer HTTP header [28] provides a basis for content negotiation in other dimensions. Inclusion of the Prefer header requires defining preference in the Vary header of an HTTP response [28]. Though the specification consists of a registry of preference (of which `return=minimal` and `return=representation` are a part), Van de Sompel et al. [30] utilized the extensibility of the definition with Prefer values of `original-content`, `original-links`, and `original-headers` despite them not being registered. These are used (in theory) to obtain the raw, unmodified content from an archive instead of content that is rewritten.

6.2 Authentication and Authorization through Negotiation

Figure 8 shows an example CDXJ containing the access attributes of type and token. These attributes for a memento specify a previously established authentication and authorization procedure with a retained token for access persistence. In this initial work,

⁴“Star” here refers to common syntax for a wildcard (*)

- (1) User requests captures for URI-R from MMA
- (2) MMA requests URI-R from Public Web Archives $P_{u_1...n}$ and Private Web Archive P_{r_1}
 - $P_{u_1...n}$ each return a respective set of URI-Ms $\{\{M_1\}, \{M_2\}, \dots, \{M_n\}\}$ to MMA
 - P_{r_1} returns an HTTP 401 and an identifier for an authentication mementity (URI-P)
- (3) MMA returns HTTP 401, URI-P, and P_{r_1} identifier to User
- (4) User sends credentials and URI-R to URI-P
- (5) mementity at URI-P returns a token to User
- (6) User requests URI-R again from MMA with token and P_{r_1} identifier
- (7) MMA requests URI-R from P_{r_1} along with token
 - P_{r_1} returns the set of URI-Ms $\{M_{Pr}\}$ to MMA after potentially consulting mementity at URI-P for validity
- (8) MMA sorts and transforms $\{\{M_1\}, \{M_2\}, \dots, \{M_n\}, \{M_{Pr}\}\}$ into a TimeMap for URI-R
- (9) MMA returns TimeMap to User

Figure 11: Abstraction of the authentication to private Web archives follows a flow similar to OAuth 2.

we use an OAuth 2.0 procedure to established these attributes but the representation is extensible and not coupled to the procedure dynamics.

Figure 11 describes the interaction flow of authentication and authorization to a private Web archive. This model uses the model described by OAuth 2.0 wherein the archive from which a capture is being requested takes on the roles of the resource owner and resource server (a fundamental pattern described in the specification), an MMA or user taking on the role of the client, and a mementity at URI-P (an identifier for an authentication mementity) taking on the role of the authorization server.

6.3 Endpoint for Archival Aggregate Enrichment

StarGates are responsible for receiving data about URI-Ms and enriching any subsequent TimeMaps. In typical usage of Memento, the TimeGate mementity is not aware of the state of the resources it identifies. Two different approaches can be used to specify additional attributes for URI-Ms in a TimeMap: server-driven and client-driven enrichment.

In server-driven enrichment, the StarGate (or some other server-based mementity) accesses the URI-M for a URI-R in the archive and attempts to acquire the content-based attributes described in Section 4.1. These attributes are then retained by the server-based mementity and added inline to TimeMap when the respective original URI-R is requested by a client.

A client-based enrichment approach involves further user interaction of the client accessing a URI-M and a StarGate. A StarGate may provide a URI-M that acts as a proxy to the URI-M that would be requested in conventional Memento usage. By acting as a proxy, a StarGate may setup further communication between the client and StarGate when a URI-M is accessed. This may be accomplished using conventional JavaScript callbacks, a runtime executed service worker (similar to the approach for rerouting at replay-time by Alam et al. [3]).

The client-side approach also allows for distribution of the computation procedure for derived attributes (Section 4.2) with the StarGate acting as an endpoint for the result. As a safeguard to prevent malicious or miscalculated data from being served in a TimeMap, a

StarGate may use a consensus model to ensure the accuracy of the result prior to associating it with a URI-M. As an example, a StarGate may change the URI-M in Figure 4 from <http://web.archive.org/web/19981212013921/http://facebook.com/> to <http://stargatehost/calculate/http://web.archive.org/web/19981212013921/http://facebook.com/>. Upon a client accessing the latter URI, the StarGate returns a page to the client with callback information, a key to associate the calculation procedure, and a redirect to the former URI-M with additional embedded JavaScript. For clients that do not support service worker or JavaScript (e.g., curl), accessing the stargatehost URI will provide the same experience the client would receive if accessing the non-proxied URI-M.

7 DISCUSSION AND FUTURE WORK

We developed initial prototypes of the MMA⁵ (extending MemGator [4]) and StarGate⁶ as well as integrated a prototype of Mink [16] with the capabilities to interact with other mementities through the dynamics described in this work. Through the implementation of the concepts, mementities, and dynamics described here, users with private Web archives may aggregate their captures both with other private as well as public Web archives to get a better picture of the Web as it was without compromising the information contained in their captures. In the future, we anticipate exploring additional attributes to associate with URI-Ms as classified and described in Section 4 along with the temporal and spatial ramifications of further supplementing TimeMaps and Link response headers as well as the ramifications on caching and efficient querying and aggregation of the TimeMaps by users and mementities alike. We plan on further exploring archival query precedence models (Section 3) to consider attributes of archives and their contained mementos in dimensions beyond public-private as well as more complex asynchronous models and short-circuiting techniques.

8 CONCLUSIONS

In this work we laid the foundation for aggregating private and public Web archives. We introduced conceptual Web archiving Memento entities (mementities) to facilitate a hierarchical approach toward aggregation and provided extensible means and methods to consider to further aggregate these captures for a better picture of the Web of the past. We introduced and explored archival precedence and short-circuiting of requests to archival aggregators to allow querying of individual or subsets of archives and to halt if and when a condition is met. We provided the syntax and semantics for enriching Memento TimeMaps with additional attributes to encourage them to be more expressive, particularly as required when aggregating private mementos. We introduced a model to integrate conventional live Web authentication methods with an additional mementity (StarGate) for systematic access control to private captures both from an individual user as well as from other mementities.

Acknowledgements. This work is supported by the NEH grant HK-50181-14 and IMLS grant RE-33-16-0107-16. We would also like to thank Scott Ainsworth, Sawood Alam, and Shawn Jones for preliminary reviews. Some icons adapted from the work of Agatha Krych and licensed CC-BY-SA 4.0.

⁵<https://github.com/machawk1/gogator>

⁶<https://github.com/machawk1/stargate>

REFERENCES

- [1] Scott G. Ainsworth, Michael L. Nelson, and Herbert Van de Sompel. 2015. Only One Out of Five Archived Web Pages Existed as Presented. In *Proceedings of the 26th ACM Conference on Hypertext & Social Media*. 257–266.
- [2] 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).
- [3] 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.
- [4] 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.
- [5] John 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, Mat Kelly, Michele C. Weigle, and Michael L. Nelson. 2015. Not All Mementos Are Created Equal: Measuring the Impact of Missing Resources. *International Journal on Digital Libraries* 16 (May 2015), 283–301. Issue 3.
- [7] R. Fielding and J. Reschke. 2014. Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content. IETF RFC 7231. (June 2014).
- [8] J. Gregorio and B. de hOra. 2007. The Atom Publishing Protocol. IETF RFC 5023. (October 2007).
- [9] Dick Hardt. 2012. The OAuth 2.0 Authorization Framework. IETF RFC 6749. (October 2012).
- [10] K. Holtman and A. Mutz. 1998. Transparent Content Negotiation in HTTP. IETF RFC 2295. (March 1998).
- [11] Andy Jackson. 2015. The CDX File Format. <https://iipc.github.io/warc-specifications/specifications/cdx-format/cdx-2015/>. (2015).
- [12] Shawn M. Jones, Herbert Van de Sompel, and Michael L. Nelson. 2016. Mementos in the Raw. <http://ws-dl.blogspot.com/2016/04/2016-04-27-mementos-in-raw.html>. (April 2016).
- [13] Mat Kelly. 2012. *An Extensible Framework for Creating Personal Archives of Web Resources Requiring Authentication*. Master's thesis. Old Dominion University, Norfolk, Virginia, USA.
- [14] Mat Kelly, Lulwah M. Alkwa, Sawood Alam, Michael L. Nelson, Michele C. Weigle, and Herbert Van de Sompel. 2017. Impact of URI Canonicalization on Memento Count. In *Proceedings of JCDL*. 303–304.
- [15] Mat Kelly, Justin F. Brunelle, Michele C. Weigle, and Michael L. Nelson. 2013. A Method for Identifying Personalized Representations in the Archives. *D-Lib Magazine* 19, 11/12 (Nov/Dec 2013). <http://www.dlib.org/dlib/november13/kelly/11kelly.html>
- [16] 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.
- [17] Mat Kelly and Michele C. Weigle. 2012. WARCcreate - Create Wayback-Consumable WARC Files from Any Webpage. In *Proceedings of JCDL*. 437–438.
- [18] Evan Light. 2016. The Snowden Archive-in-a-Box: A year of travelling experiments in outreach and education. *Big Data & Society* 3, 2 (2016), 1–7.
- [19] Siân E. Lindley, Catherine C. Marshall, Richard Banks, Abigail Sellen, and Tim Regan. 2013. Rethinking the Web as a Personal Archive. In *Proceedings of the 22nd International Conference on World Wide Web*. 749–760.
- [20] Catherine C. Marshall and Frank M. Shipman. 2012. On the Institutional Archiving of Social Media. In *Proceedings of JCDL*. 1–10.
- [21] Catherine C. Marshall and Frank M. Shipman. 2014. An Argument for Archiving Facebook As a Heterogeneous Personal Store. In *Proceedings of JCDL*. 11–20.
- [22] Frank McCown and Michael L. Nelson. 2009. What Happens When Facebook is Gone?. In *Proceedings of JCDL*. 251–254.
- [23] M. Nottingham. 2005. The Atom Syndication Format. IETF RFC 4287. (December 2005).
- [24] Mark Nottingham. 2017. Web Linking. IETF RFC 8288. (October 2017).
- [25] David S. H. Rosenthal. 2010. The Importance of Discovery in Memento. <http://blog.dshr.org/2010/12/importance-of-discovery-in-memento.html>. (December 2010).
- [26] David S. H. Rosenthal. 2011. Memento & the Marketplace for Archiving. <http://blog.dshr.org/2011/01/memento-marketplace-for-archiving.html>. (January 2011).
- [27] David S. H. Rosenthal. 2016. Content negotiation and Memento. <http://blog.dshr.org/2016/08/content-negotiation-and-memento.html>. (August 2016).
- [28] J. Snell. 2014. Prefer Header for HTTP. IETF RFC 7240. (June 2014).
- [29] 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).
- [30] 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).
- [31] E. Wilde. 2013. The 'profile' Link Relation Type. IETF RFC 6906. (March 2013).