

# A Chromium-based Memento-aware Web Browser

Abby Mabe<sup>1</sup>, Michael L. Nelson<sup>1</sup><sup>[0000–0003–3749–8116]</sup>, and Michele C. Weigle<sup>1</sup><sup>[0000–0002–2787–7166]</sup>

Old Dominion University  
Norfolk, VA 23529 USA  
`amabe002@odu.edu`, `{mln,mweigle}@cs.odu.edu`

**Abstract.** Current web browsers do not differentiate between the live Web and the past Web. If a user loads an archived webpage, or *memento*, they have to rely on elements in the page itself to be aware that they are not viewing a live webpage. Another problem is that browser bookmarks only save the URL of a webpage. But if the webpage is deleted or changed, the user may have lost access to the very information that they had wanted to save. We present a proof-of-concept Memento-aware Browser, built on Google’s Chromium open-source web browser. We add native memento detection that displays an icon when the user is visiting an archived webpage and a “bookmark as archive” feature that allows users to request archiving of individual webpages when they are bookmarked.

**Keywords:** Web browser · Web archiving · Memento · Bookmarks.

## 1 Introduction

Archiving the Web has become increasingly important. Archived webpages, or *mementos* [37], allow users to go back in time on the Web and view webpages as they used to appear. One way to view mementos is by using a web browser to visit a web archive, such as the Internet Archive [12] or Archive.today [5]. We rely on web browsers to allow us to safely and securely navigate the Web. Web browsers process information about a webpage as it loads and present us with appropriate user interface (UI) cues and messages, keeping us aware of important information about the webpages we are viewing. For example, browsers can detect if a connection is secure and warn us when a website is potentially unsafe. However, unlike displaying HTTP status or blocking web trackers, web browsers do not have native support for indicating that a displayed webpage is a memento. Because of this, the user has to look for visual cues on the page itself or inspect the browser’s address bar to see if the webpage they are viewing is from the past or live Web. As web archiving becomes more known by the general public, web browsers should natively provide mechanisms for users to be aware of and interact with archived webpages.

We have built a *proof-of-concept* Memento-aware Web Browser, based on Google’s Chromium open-source browser [29]. Prior work has focused on building

browser extensions [14,27], but extensions depend on the user choosing to install and activate the extension. Only users who already know about web archives would be likely to install a browser extension related to web archiving. With this work we wanted to demonstrate the feasibility of native browser support, in the same idiom that browsers currently convey significant metadata about what they display. With such a capability in browsers, web archives could gain even more visibility with the general public. We are not advocating that users switch to a new browser, but exploring what could be enabled if current popular browsers included such capabilities. We have added memento detection with UI additions to notify the user when they are viewing an archived webpage. We have also added a “bookmark as archive” feature that allows users to request that the page they are bookmarking be archived by a public web archive. This proof-of-concept browser is available for download in a GitHub repo [20] and more details on the implementation are available in a tech report [19].

## 2 Background

Memento [37] is an HTTP protocol extension that allows for content-negotiation in the time dimension, essentially providing HTTP-level mechanisms for accessing archived webpages. A web archive that is Memento-compliant will return Memento headers in the HTTP response. For our purposes, the key header is Memento-Datetime (e.g., `Memento-Datetime: Mon, 12 Apr 2010 12:50:57 GMT`), which indicates the datetime when the resource was captured. Web archives only return this header for archived resources.

In the Memento framework, the identifier of an original resource from the live Web is a *URI-R* and the identifier of an archived version of that resource at a particular point in time is a *URI-M*. Many web archives construct URI-Ms that include the Memento-Datetime and the URI-R, such as <https://web.archive.org/web/20100412125057/http://www.mitre.org/>. In this example, the Memento-Datetime is represented by the 14-digit date string 20100412125057 and the URI-R is <http://www.mitre.org/>. Not all web archives use this type of URI-M construction, so we will use the presence of the Memento-Datetime HTTP header as an indication that the resource is a memento.

Within the Chromium browser, HTTP response header information for all resources on a page makes up the Security State for that page. Included in the Security State is information about whether the page contains any insecure (HTTP) content and if the user should be presented with the HTTPS secure lock icon or the “Not Secure” warning. We used the method for displaying the security icon as a template for adding a Memento icon to Chromium.

## 3 Related Work

There have been several browser extensions developed for interacting with web archives. Sanderson et al. [27] describe the MementoFox Firefox plugin that allowed users to navigate in the past Web, an Internet Explorer plugin providing

the same functionality, and Memento Browser app for Android mobile devices that allowed users to view only mementos, but not the live Web. Nelson et al. [22,23] describe a Google Chrome extension for navigating to mementos. Mink [14] is another Chrome extension that allowed users to survey all of the available mementos for a web page and provided a feature to add live web pages to public web archives. Tweedy et al. [33] developed a Memento Web Browser for iOS mobile devices. The Internet Archive has developed extensions [9,13] for Chrome, Firefox, Edge, and Safari, but these only provide mementos from the Internet Archive. All of these tools provide valuable features and demonstrate the usefulness of being able to interact with the past Web, but they all require the user to install an extension to a browser or install a separate app from the user’s standard mobile web browser. Our goal is to demonstrate how memento-aware UI elements could be integrated into a standard web browser, without having to use an extension. The Internet Archive has collaborated with Brave to integrate automatic links to the Wayback Machine on 404 pages [10], but this is not a fully Memento-aware browser implementation.

In addition to providing navigation in the past Web, some of the extensions also allow users to submit webpages for archiving from the browser [13,14]. This feature is also something that we support, but again, our goal is to show how this could be integrated into a standard browser without an extension.

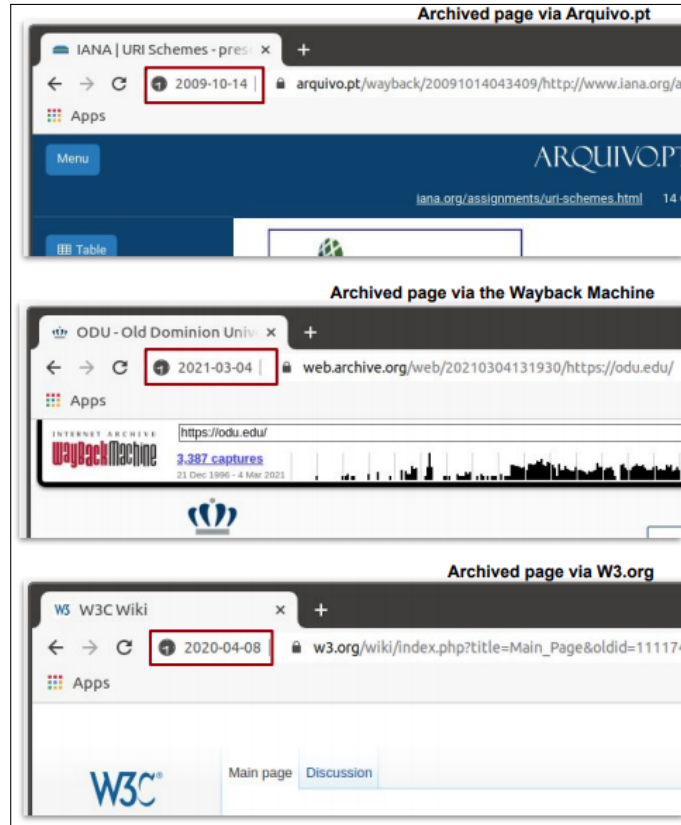
## 4 Memento Detection and Icon Display

The first step to creating a Memento-aware Browser is to be able to reliably detect archived resources. As mentioned in Section 2, we use the presence of the Memento-Datetime HTTP header to indicate that a resource is a memento. For most web archives, this is trivial as the root HTML page is a memento and includes the Memento-Datetime HTTP header. But some archives include the root memento as an iframe inside a live webpage. In addition, we want to also detect when root mementos load resources from the live Web. Here we discuss these three scenarios.

### 4.1 Detection of Root Page Memento

The first memento detection possibility to consider is when the entire root webpage is archived and considered to be a memento. In this case, we want the browser to classify the webpage as a memento and alert the user that the page is archived and not live. We cannot accomplish this by looking at the URI of the webpage alone since, as mentioned in Section 2, there is no single defined format for a URI-M. Instead, we parse the HTTP response headers and look for the Memento-Datetime HTTP header to determine if the webpage is a memento.

When the browser detects that a root webpage is a memento, the Memento icon plus the datetime in YYYY-MM-DD format will appear to the left of the HTTPS secure lock icon, as shown in Figure 1.



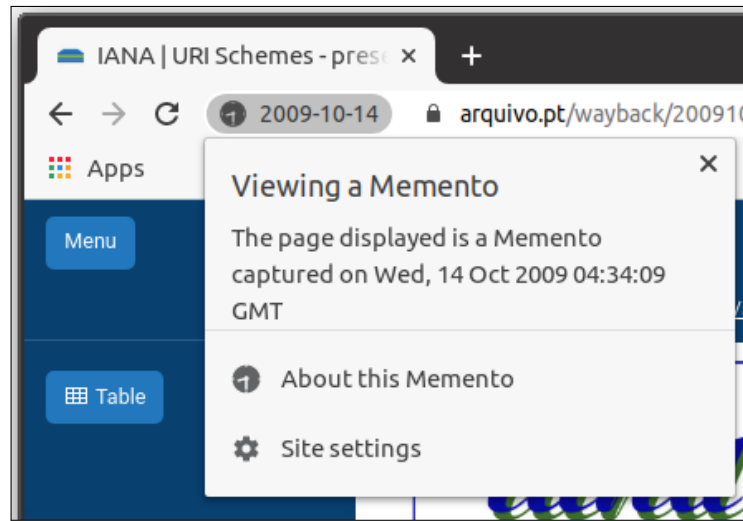
**Fig. 1.** If the root page is considered to be an archived page, the Memento icon will display along with the Memento-Datetime in YYYY-MM-DD format.

The user can click on the Memento icon for more information about the archived page as shown in Figure 2, similar to the operation of the HTTPS secure lock icon to bring up the connection popup for more connection security information.

## 4.2 Detection of Mementos Embedded in a Live Page

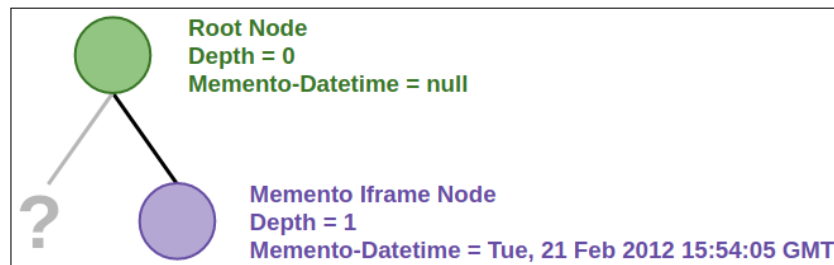
It is possible that a live root webpage could contain one or more embedded frames with archived content. Most commonly this occurs when the root webpage contains an iframe that loads an archived webpage. There are two different scenarios where this may occur.

First, the iframe memento may be intended to make up the whole webpage. In this case, the root page should be considered a memento even though the HTTP response for the root page does not contain the Memento-Datetime header. Web archives such as Trove [32] and Perma.cc [25] display archived web-



**Fig. 2.** The user can click the memento icon to view the exact memento datetime.

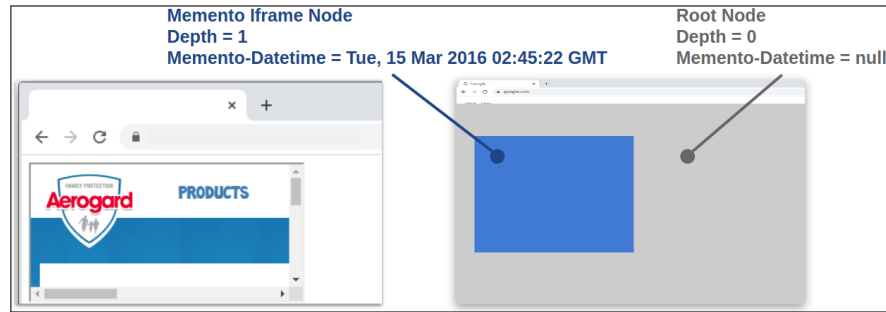
pages in this manner. Figure 3 shows this case translated into a tree structure where the root webpage is the root node and the memento iframe is the child node. The Memento-aware Browser will detect the datetime from the headers of the archived content in the iframe, detect that the iframe is formatted to make up the entire page, and then promote the frame's Memento-Datetime to the root. In this case, the Memento icon and Memento-Datetime will display in the browser just as in the root page memento case.



**Fig. 3.** Tree structure of a live root webpage from Trove containing an iframe Memento

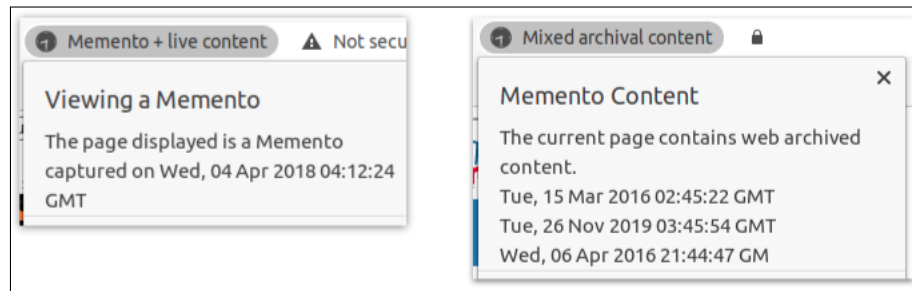
The second case involves a live webpage that is simply displaying a memento without intending for the memento to make up the whole page. This possibility is shown in Figure 4, where a live page has included a memento for <https://www.aerogard.com.au/products/> in an iframe. This can be translated into a tree structure similar to the structure of the memento from Trove shown in Figure

3. In this case, the browser will detect the archived content within the page, but also detect that the embedded iframe is an additional element on the live page. The browser similarly handles the case where there are multiple iframes of archived content on the live webpage.



**Fig. 4.** Memento displayed within an iframe as an additional page element on a live webpage

When the user visits such pages, the Memento-aware Browser will display the Memento icon and show the message “Mixed archival content” instead of a YYYY-MM-DD datetime. All Memento-Datetimes will be listed in the popup that appears when the Memento icon is clicked (Figure 5, right).



**Fig. 5.** The browser shows a message if the archived page contains live content (left). If the page is live and contains archived content, the browser will indicate this (right).

#### 4.3 Detection of Live Content Embedded in a Memento

An archived iframe could contain one or more embedded elements that are displaying content from the live Web. This is a rare yet undesirable occurrence since the archived webpage is intended to present content from the past. When

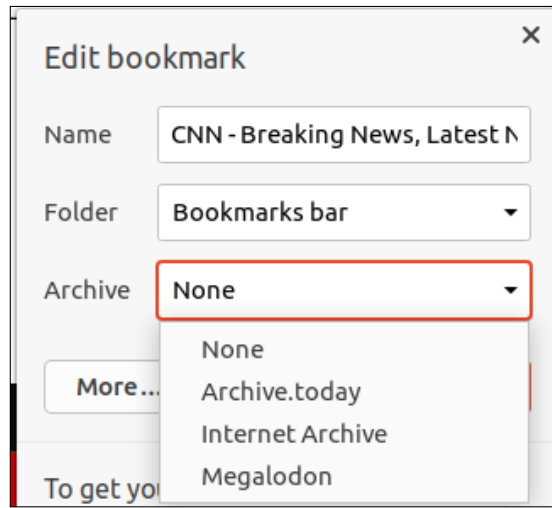
this occurs, it is not obvious to the user that the live Web is leaking into the archived page. Such occurrences have been called “zombies” [8] and “archive escapes” [16]. If frames within the archived content do not return the Memento-Datetime header, the user should be alerted that the archived content they are viewing contains elements from the live Web. When live content is detected in the archived page, the browser will display the message “Memento + live content” and the popup will include information for the containing archived page (Figure 5, left).

## 5 Bookmark as Archive

In addition to making users aware of any archived content they are viewing, users should also be able to easily archive any live content that they want to save [38]. Since bookmarking only saves the URI of the page, if the page were to be taken down, resolving the URI would result in an HTTP 404 `Not Found` status. Additionally, if page remained available but the content on the webpage were to change, the content that the user originally saw and wanted to save would no longer be present. A way to make these bookmarked pages constant is through the use of public web archives. In addition to the standard bookmark functionality, the Memento-aware Browser submits the webpage for capture by a user-selected public web archive. In this way, the user has access to both the traditional live Web bookmark and a bookmark to archived content. This feature incorporates archiving into the standard bookmarking process.

To implement the “bookmark as archive” feature, we added a dropdown to Chromium’s edit bookmark popup, shown in Figure 6. The user selects the desired archive, and the browser uses ArchiveNow [6,7] to submit the request to the web archive.

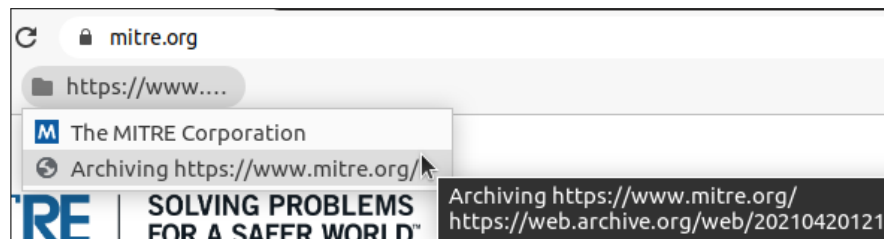
When the user clicks the bookmark star icon, a standard bookmark node for the URI is created. If the user selects an archive in the edit bookmark popup, then when they exit the popup, a request will be submitted to the selected archive and a bookmark node for the archived version will be created. Processing the archiving request at the web archive can take between a few seconds to several minutes to complete. All bookmark actions run on the UI thread, and we have implemented the archiving process on the background thread since it can take several minutes to complete. Because of this, when the archiving is complete, the bookmark node cannot be updated without crashing the browser with an invalid sequence error. Instead of creating a placeholder bookmark node that links to the URI-R, a bookmark node is created that links to a special URI-M with the datetime of the time of submission. The Internet Archive’s Wayback Machine and Archive.today both support URI-Ms with 14-digit date strings, as described in Section 2. For example, <https://web.archive.org/web/20100412125057/http://www.mitre.org/> is a URI-M for an archived webpage in the Wayback Machine. If you change the date string to one for which the Wayback Machine does not have a memento, it will redirect to the memento with the closest datetime that they do have. The same is true for Archive.today. But Megalodon.jp does not



**Fig. 6.** The “bookmark as archive” option appears in the edit bookmark popup.

redirect to the nearest datetime, so this method cannot be used. However, once the archiving process completes, the actual URI-M is returned and added to the file `archive-urls.txt` so that it can be accessed later.

To keep bookmarks for the live Web version and archived versions together (a user may create multiple archived versions for the same URI-R), we create a bookmark folder titled with the URI-R. The original bookmark node will be placed into this folder alongside the archive bookmark node. This way, the user will see their single bookmark folder on the bookmarks bar and when they click it they will see a dropdown showing all the versions of that page they saved (Figure 7).

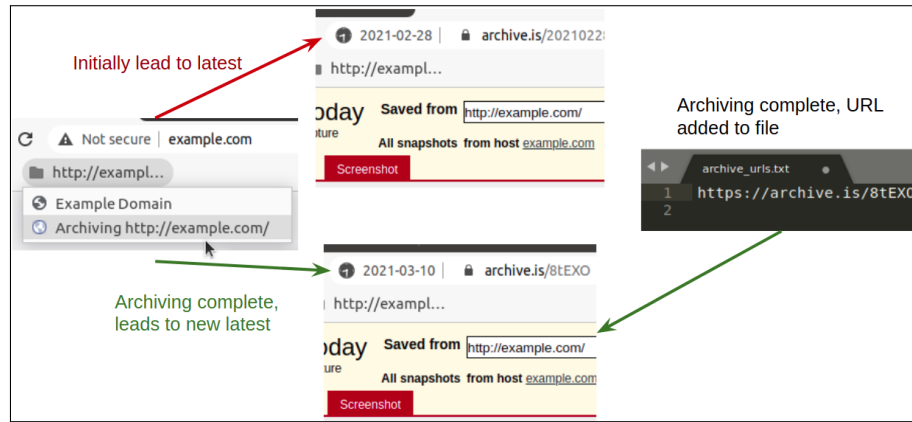


**Fig. 7.** When the user archives a page they have bookmarked, a folder gets created to hold all the bookmark nodes that lead to that webpage, either live or archived.

To walk through the example shown in Figure 8, say the user wanted to use the “bookmark as archive” feature to submit [example.com](https://example.com) to Archive.today on



March 10, 2021 at 4:18pm. The browser will immediately create a bookmark node folder for the webpage and place two bookmark nodes in the folder, the first being the standard bookmark that leads to the live webpage and the second bookmark node being the one for the archived version of the page. The URI-M for the archive bookmark node would be constructed using the method described above and would initially redirect to Archive.today's latest memento of [example.com](http://example.com), which at that time would be from February 28, 2021. When the archiving process completes a few minutes later, the archive bookmark node URI-M will redirect to the newly added URI-M of [example.com](http://example.com) that was archived a few minutes after the user submitted it.



**Fig. 8.** The archive bookmark node will redirect to the memento in the archive that is closest to the given datetime.

## 6 Evaluation

### 6.1 Memento Detection

Memento detection was tested with the web archives and webpages shown in Table 1. Each of the tested web archives was navigated to with the browser so that memento detection could be tested. The Memento icon does not display until an actual archived page was selected, meaning the memento detection was working as expected by not displaying the icon while selecting a memento from the archive. Figure 1 shows examples of testing with four different web archives. Three of the tested archives display the root memento in an iframe meant to make up the whole webpage (Section 4.2), and we verified that those displayed the YYYY-MM-DD datetime along with the Memento icon. We also tested live webpages with one and three iframe mementos, and these properly displayed the

“Mixed archival content” message. Finally, we tested with a memento that contained an embedded YouTube video, and the browser displayed the “Memento + live content” message.

**Table 1.** Memento Detection Evaluation

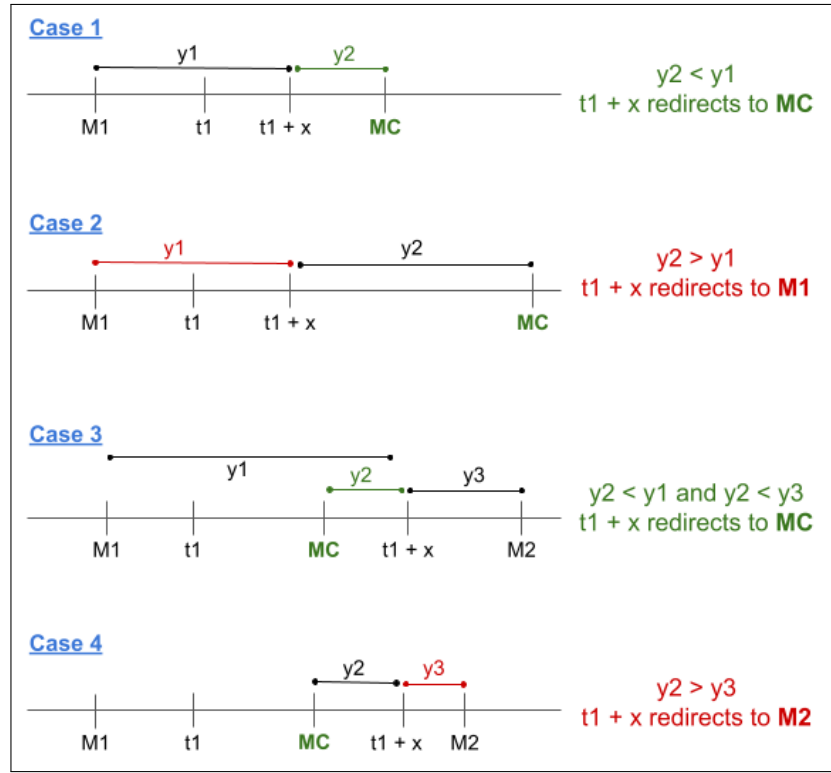
<b>cases</b>	<b>archives/webpages tested</b>
Root Memento	Archive-It [4], Archive.today [5], Icelandic Web Archive [11], Internet Archive [12], Library and Archives Canada [17], Library of Congress [18], National Records of Scotland [21], Portuguese Web Archive [26], Stanford Web Archive [28], UK National Archives Web Archive [34], UK Parliament Web Archive [35]
Root Memento in iframe	Perma Archive (Perma.cc) [25], Australian Web Archive (Trove) [32], UK Web Archive [36]
Root Live with Memento	Webpage with one iframe memento, webpage with three iframe mementos
Root Memento with Live	Memento with embedded YouTube video

## 6.2 Bookmark as Archive

Since the “bookmark as archive” URI-M does not link to the exact memento created, there is a chance it will be redirected to a memento that is not the one created from the user’s submission. Since archiving takes some time, we could add an offset to the date string to increase the likelihood that the correct memento is selected. Here we evaluate options for different offset values.

We illustrate the timeline for mementos and offsets in Figure 9. We show four cases, two that are successful (Case 1 and Case 3) and two that result in an incorrect memento being selected (Case 2 and Case 4). In the figure,  $M1$  is the latest memento before the bookmark is created,  $t1$  is the time the bookmark request is made,  $t1+x$  is the datetime used in the bookmark URI-M, with  $x$  being the offset value we are testing, and  $MC$  is when the bookmark memento is actually created. The difference between Case 1 and Case 2 is how long it takes for the archive to create the new memento. Just looking at Cases 1 and 2, we might think it best to set the offset  $x$  to be large. But we also have to consider the cases where another memento,  $M2$ , is created in the archive (by some other process) after the bookmarking process has started (Cases 3 and 4). If the offset results in the bookmark URI-M being too close to  $M2$  (Case 4), then the incorrect memento will be selected. The archive will redirect the URI-M to the closest existing datetime, so we want to set  $t1+x$  so that in most cases, it is closer to  $MC$  than any other mementos, without knowing exactly how long the archive will take to capture the page or what other mementos exist.

To determine an appropriate offset for popular websites, we analyzed 17 TimeMaps to see how well different offsets would return the correct memento.



**Fig. 9.** Four cases of timelines for mementos and offsets for “bookmark as archive”.

The 17 TimeMaps were of popular news websites [24] likely to have been archived frequently. For each TimeMap, a datetime for each second between January 1, 2021 and April 16, 2021 was generated and the memento closest to that datetime was found, representing MC. Then, an offset was added to the datetime ( $t1+x$ ) and the memento closest to that new datetime was found. If the memento closest to the original datetime was the same as the memento closest to the datetime plus the offset, this was counted as the offset leading to the correct memento. In this evaluation, only seconds were considered since the Memento-Datetimes have only second granularity. We tested offsets of {30, 60, 120} seconds for each of the 9 million seconds in our test period over all 17 TimeMaps. We obtained an accuracy of 99.1% with a 30-second offset, 98.4% with a 60-second offset, and 96.9% with a 120-second offset.

## 7 Future Enhancements

We see several features and enhancements that could be built on top of the existing features, specifically the memento detection feature since it stores the

Memento-Datetimes for all resources but is currently only considering the date-time returned by the main URI.

*“About This Memento” Popup.* Our current implementation has a placeholder for an “About This Memento” popup that could list all the archived embedded resources and their respective datetimes that make up the entire webpage, similar to the Internet Archive’s “About this capture” drop-down. Since the Memento-aware Browser stores the Memento-Datetime values for all resources, implementing this feature would mainly consist of completing the front-end. An added feature would be allowing the user to click on a listing for a resource within the popup and have that resource highlighted on the page.

*List of Known Web Archives.* It is possible for a live webpage to return the Memento-Datetime header and spoof a memento. A possible guard against this is to only acknowledge the header if its iframe is from a known web archive. This would require the browser to have a hard-coded list of known web archives. There are existing lists of known web archives, including those recognized by the MemGator memento aggregator [3], TimeTravel [31], and Webrecorder [15].

*Temporal Coherence.* Another addition would be to consider the temporal coherence of the memento [2]. An HTML webpage and all the embedded resources required for the full page are referred to as a *composite memento* [1]. It is not guaranteed that all of the resources in a composite memento will be archived at the same time, or even on the same day. If embedded resources are replayed from a datetime with a large offset from the main HTML page, composite mementos could be constructed that never existed on the live Web. When this occurs, the browser should alert the user that the page they are viewing is potentially temporally incoherent. This could be implemented as an infobar [30] that could be displayed at various times to alert the user without the obtrusiveness of a popup.

## 8 Conclusion

We have created a proof-of-concept Memento-aware Browser by extending Google’s Chromium open source web browser. We added memento detection, notification of mementos in the UI, and a “bookmark as archive” feature. We accounted for complex memento detection, such as iframe mementos and mementos that contain live web resources. For “bookmark as archive”, we found that since all bookmark actions run on the UI thread, bookmarks could not be updated after archiving was completed. We implemented a workaround taking advantage of Internet Archive and Archive.today redirecting links that contain datetimes they do not have to mementos closest to that datetime. The Memento-aware Browser source code [20] can be built and run on Linux and Windows. Additionally, there are release builds for Linux and Windows attached to the GitHub repository.

## References

1. Ainsworth, S.G., Nelson, M.L., Van de Sompel, H.: A framework for evaluation of composite memento temporal coherence. Tech. Rep. arXiv:1402.0928, arXiv (2014)
2. Ainsworth, S.G., Nelson, M.L., Van de Sompel, H.: Only one out of five archived web pages existed as presented. In: Proceedings of ACM Hypertext. pp. 257–266 (2015)
3. Alam, S., Nelson, M.L.: MemGator - a portable concurrent Memento aggregator: Cross-platform CLI and server binaries in Go. In: JCDL '16: Proceedings of the 16th ACM/IEEE-CS Joint Conference on Digital Libraries. pp. 243–244 (2016)
4. Archive-It: <http://www.archive-it.org>
5. Archive.today: <http://archive.vn/>
6. Aturban, M.: ArchiveNow. <https://github.com/oduwsdl/archivenow>
7. Aturban, M., Kelly, M., Alam, S., Berlin, J.A., Nelson, M.L., Weigle, M.C.: ArchiveNow: Simplified, extensible, multi-archive preservation. In: JCDL '18: Proceedings of the 17th ACM/IEEE-CS Joint Conference on Digital Libraries. pp. 321–322 (2018)
8. Brunelle, J.F.: Zombies in the Archives. <https://ws-dl.blogspot.com/2012/10/2012-10-10-zombies-in-archives.html> (Oct 2012)
9. Graham, M.: Wayback Machine Chrome extension now available. <http://blog.archive.org/2017/01/13/wayback-machine-chrome-extension-now-available/> (Jan 2017)
10. Graham, M.: Brave Browser and the Wayback Machine: Working together to help make the web more useful and reliable. <http://blog.archive.org/2020/02/25/brave-browser-and-the-wayback-machine-working-together-to-help-make-the-web-more-useful-and-reliable/> (Feb 2020)
11. Icelandic Web Archive: <https://wayback.vefsafn.is/>
12. Internet Archive: <http://www.archive.org>
13. Internet Archive: Wayback Machine web browser extension. <https://github.com/internetarchive/wayback-machine-webextension>
14. Kelly, M., Nelson, M.L., Weigle, M.C.: Mink: Integrating the live and archived web viewing experience using web browsers and memento. In: JCDL '14: Proceedings of the 14th ACM/IEEE-CS Joint Conference on Digital Libraries. pp. 469–470 (2014)
15. Kreymer, I.: Public web archives. <https://github.com/webrecorder/public-web-archives> (2021)
16. Lerner, A., Kohno, T., Roesner, F.: Rewriting history: Changing the archived web from the present. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. pp. 1741–1755 (2017). <https://doi.org/10.1145/3133956.3134042>
17. Library and Archives Canada: <https://webarchive.bac-lac.gc.ca/>
18. Library of Congress: <https://webarchive.loc.gov/>
19. Mabe, A.: A Chromium-based Memento-aware web browser. Tech. Rep. arXiv:2104.13361, arXiv (2021), <https://arxiv.org/abs/2104.13361>
20. Mabe, A.: Memento-aware Browser. <https://github.com/oduwsdl/Memento-aware-Browser> (2021)
21. National Records of Scotland: <https://webarchive.nrscotland.gov.uk/>
22. Nelson, M.L.: Right-click to the past – Memento for Chrome . <https://ws-dl.blogspot.com/2013/10/2013-10-14-right-click-to-past-memento.html> (Oct 2013)
23. Nelson, M.L., Van de Sompel, H.: Adding the dimension of time to HTTP. In: Brügger, N., Milligan, I. (eds.) The SAGE Handbook of Web History, chap. 14, pp. 189–214. SAGE Publishing (2018)

24. Nwala, A.C., Weigle, M.C., Nelson, M.L.: 365 dots in 2019: Quantifying attention of news sources. Tech. Rep. arXiv:2003.09989, arXiv (2020)
25. Perma.cc: <https://perma.cc/>
26. Portuguese Web Archive: <https://arquivo.pt/>
27. Sanderson, R., Shankar, H., Ainsworth, S., McCown, F., Adams, S.: Implementing time travel for the web. *Code4Lib Journal* (13) (2011)
28. Stanford Web Archive: <https://swap.stanford.edu/>
29. The Chromium Projects: Download Chromium. <https://www.chromium.org/getting-involved/download-chromium>, Accessed on 02.06.2021
30. The Chromium Projects: Infobars. <http://www.chromium.org/for-testers/enable-logging>, Accessed on 02.02.2021
31. TimeTravel: Time travel archive registry. [http://labs.mementoweb.org/aggregator\\_config/archivelist.xml](http://labs.mementoweb.org/aggregator_config/archivelist.xml) (2021)
32. Trove: <https://trove.nla.gov.au/>
33. Tweedy, H., McCown, F., Nelson, M.L.: A memento web browser for iOS. In: *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries (JCDL)*. pp. 371–372 (Jul 2013). <https://doi.org/10.1145/2467696.2467764>
34. UK National Archives Web Archive: <https://webarchive.nationalarchives.gov/>
35. UK Parliament Web Archive: <https://webarchive.parliament.uk/>
36. UK Web Archive: <https://webarchive.org.uk/>
37. Van de Sompel, H., Nelson, M., Sanderson, R.: HTTP framework for time-based access to resource states – Memento (Internet RFC 7089, December 2013)
38. Weigle, M.C.: Enabling personal use of web archives. Keynote talk presented at Web Archiving and Digital Libraries (WADL), <https://www.slideshare.net/mweigle/enabling-personal-use-of-web-archives> (June 2018)