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.

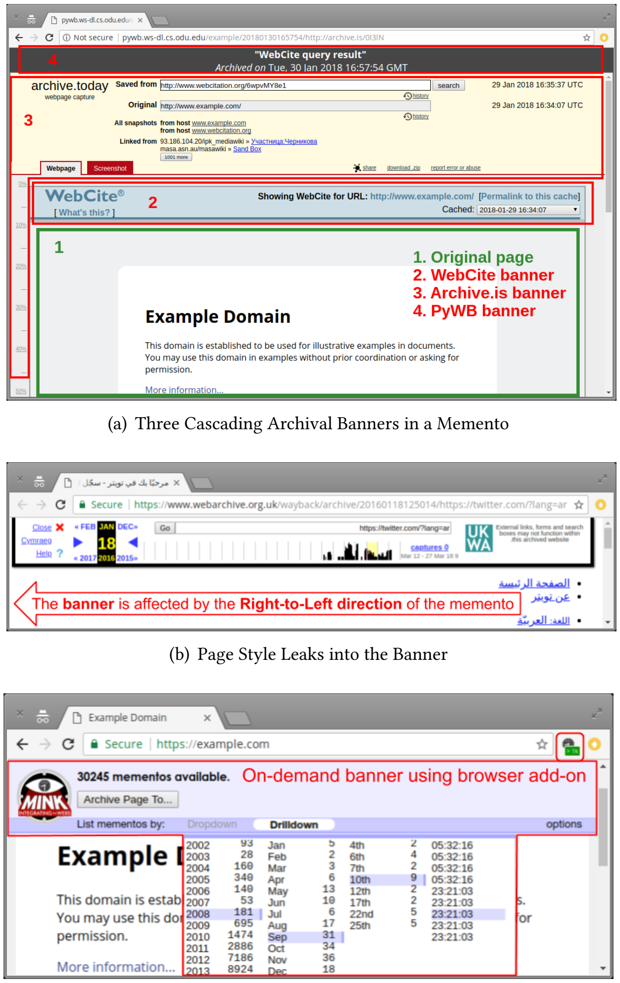
2.2 Frame/Iframe Banners

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.


pywb.ws-dl.cs.odu.edu/
Not secure | pywb.ws-dl.cs.odu.edu/example/20180130165754/http://archive.is/0I3lN
4
"WebCite query result"
Archived on Tue, 30 Jan 2018 16:57:54 GMT
archive.today
webpage capture
3
Saved from http://www.webcitation.org/6wpvMY8e1 search 29 Jan 2018 16:35:37 UTC
history
Original http://www.example.com/ 29 Jan 2018 16:34:07 UTC
history
All snapshots from host www.example.com
from host www.webcitation.org
Linked from 93.186.104.20/ipk_mediawiki » Участница:Черникова
masa.asn.au/masawiki » Sand Box
1001 more
Webpage
Screenshot
share
download .zip
report error or abuse
0%
10%
20%
30%
40%
50%
WebCite®
[What's this?]
2
Showing WebCite for URL: http://www.example.com/ [Permalink to this cache]
Cached: 2018-01-29 16:34:07
1

1. Original page
2. WebCite banner
3. Archive.is banner
4. PyWB banner

Example Domain

This domain is established to be used for illustrative examples in documents. You may use this domain in examples without prior coordination or asking for permission.

More information...

(a) Three Cascading Archival Banners in a Memento

مرحبًا بك في تويتر - سجّل
Secure | https://www.webarchive.org.uk/wayback/archive/20160118125014/https://twitter.com/?lang=ar
Close
Cymraeg
Help
« FEB
JAN
DEC»
18
« 2017
2016
2015»
Go
https://twitter.com/?lang=ar
captures 0
Mar 12 - 27 Mar 18 9
UK WA
External links, forms and search boxes may not function within this archived website.
الصفحة الرئيسة
عن تويتر
اللغة: العربيّة

The **banner** is affected by the **Right-to-Left direction** of the memento

(b) Page Style Leaks into the Banner

30245 mementos available.

On-demand banner using browser add-on

2002	93	Jan	5	4th	2	05:32:16
2003	28	Feb	2	6th	4	05:32:16
2004	160	Mar	3	7th	2	05:32:16
2005	340	Apr	6	10th	9	05:32:16
2006	140	May	13	12th	2	23:21:03
2007	53	Jun	10	17th	2	23:21:03
2008	181	Jul	6	22nd	5	23:21:03
2009	695	Aug	17	25th	5	23:21:03
2010	1474	Sep	31			23:21:03
2011	2886	Oct	34			
2012	7186	Nov	36			
2013	8924	Dec	18			

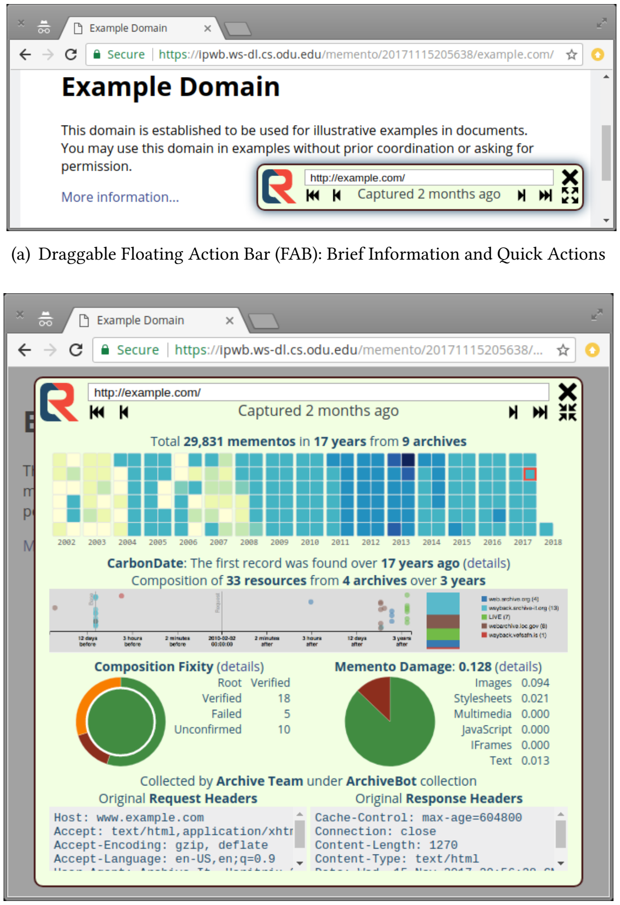
(c) Mink Banner: Injected On-demand by the Chrome Extension

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.



Example Domain

This domain is established to be used for illustrative examples in documents. You may use this domain in examples without prior coordination or asking for permission.

Captured 2 months ago

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

Captured 2 months ago

Total **29,831 mementos** in **17 years** from **9 archives**

CarbonDate: The first record was found over **17 years ago** (details)

Composition of **33 resources** from **4 archives** over **3 years**

Composition Fixity (details)

	Root	Verified
Verified		18
Failed		5
Unconfirmed		10

Memento Damage: **0.128** (details)

Images	0.094
Stylesheets	0.021
Multimedia	0.000
JavaScript	0.000
IFrames	0.000
Text	0.013

Collected by **Archive Team** under **ArchiveBot** collection

Original **Request Headers**

```
Host: www.example.com
Accept: text/html,application/xhtr
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
```

Original **Response Headers**

```
Cache-Control: max-age=604800
Connection: close
Content-Length: 1270
Content-Type: text/html
```

(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. InterPlanetary 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.