

Introduction to Digital Libraries Assignment #2

James Tate II

March 04, 2015

1 Introduction

The first part of this assignment required using several tools to create WARC¹ files from approximately 100 of the final URIs identified in assignment one. A sample of the WARC files were *replayed* in two WARC replay tools, to examine their archiving accuracy compared to a web browser's rendering of the original URI representation. The tools performed wildly differently on different URIs, and the two replay tools even had different outputs from the same WARC files.

The second part of this assignment required me to setup SOLR² and index some of the downloaded WARC files. SOLR is an open-source indexing and search tool — sample queries and results are given later in this document.

2 Methodology

This assignment required four tools be used to attempt to create WARC files for 100 URIs. Unfortunately, one of the tools, WAIL³, could not be coerced into properly creating WARC files for the URIs it was given. In short, I ran out of time and patience while trying to make Heritrix, the web crawler part of WAIL, download only the given URI and not hundreds of other pages. The other three tools were WARCreate, wget and WebRecorder.io. Before any of these tools could be used, I had to select 100 URIs to process.

2.1 URIs

To select the URIs, I first ordered all the unique final URIs from the first assignment by descending frequency. Then, from the 200 most frequently-occurring URIs, I excluded the URIs that were not text/html webpages, appeared to be spam or did not render correctly in a web browser. This left me with 144 URIs of the original 200. Then, I started using the three tools to create WARC files from the URIs until I had successfully created a WARC file using at least one method from a few more than 100 URIs.

¹<http://www.digitalpreservation.gov/formats/fdd/fdd000236.shtml>

²<http://lucene.apache.org/solr/>

³<http://matkelly.com/wail/>

2.2 wget

The first, and most simple, tool for creating WARC files from URIs was the *nix utility, wget. In a bash script, I called the below wget command once for each URI. This command downloads the given URI and supporting pages, then combines that content into a WARC file. This invocation ignores robots.txt and stores the downloaded web content in the garbage directory to keep it away from the output WARC files.

```
wget --warc-file="$2/$id" -p -l 1 -H -e robots=off \
-P "garbage/" "$uri" > "$2/${id}.wget.output" 2>&1"
```

Output from this command is saved in the *X.wget.output* files where *X* is the id of the URI. The generated WARC files are compressed with gzip and saved in *X.warc.gz* files. wget was by far the easiest tool used to create WARC files, mostly due to its usability in simple bash scripts.

2.3 WARCreate

WARCreate is a Google Chrome extension that allows the user to create a WARC file of the currently visible webpage. Using WARCreate was troublesome and produced interesting results (see Section 3.2). Most frustratingly, Chrome had to be restarted frequently to counteract the continual slowdowns of WARCreate. By design, I had to manually click the “Generate WARC” button on each of over 100 webpages after navigating to the URI in the web browser. I had to wait for the extension to give me a WARC file to save after clicking the button. Sometimes, after waiting 30 or more seconds, I would give up and move on to the next URI without ever getting a WARC.

2.4 WebRecorder.io

WebRecorder.io is a website available at <http://webrecorder.io>. It allows the user to, at no cost, “record” webpages and download the resulting WARC files. It also allows users to upload WARC files created by it and other tools to “replay” them in the web browser. Both of these functions were used in this assignment. See the Section 3.2 for details on playback using WebRecorder.io.

Using WebRecorder.io required me to paste the URI into a text field on the website, and click the “Record” button. Then, after the page rendered for a couple seconds, I could immediately download a WARC file. Although a few pages did not render correctly on WebRecorder.io, which required those URIs to be skipped, WebRecorder.io was a relatively pain-free tool to use. WARC files downloaded from WebRecorder.io were also compressed with gzip.

2.5 SOLR

Apache SOLR was the search platform used to test indexing and searching of the generated WARC files. It was simple to run on Linux by downloading the binary from the Apache website, along with the Maven binary also from Apache. The commands below were used to run SOLR from the pre-compiled binaries.

```
export PATH=$PATH:/home/jtate/src/apache-maven-3.2.5/bin/
mvn jetty:run-exploded -Djetty.port=10770
java -jar warc-indexer-2.0.1-20150116.110435-2-jar-with-dependencies.jar
```

```
-s http://localhost:10770/discovery -t ../../cs751/warc/wget/*.warc.gz
```

The last command (split across two) lines, was used to add all WARC files generated by wget to SOLR's search index/database. The web interface to SOLR allowed queries to be entered and would display the responses in JSON format. See section 3.3 for SOLR results.

3 Results

This section describes the observed performance of WARC creation and playback, as well as searching using SOLR.

3.1 WARC Files

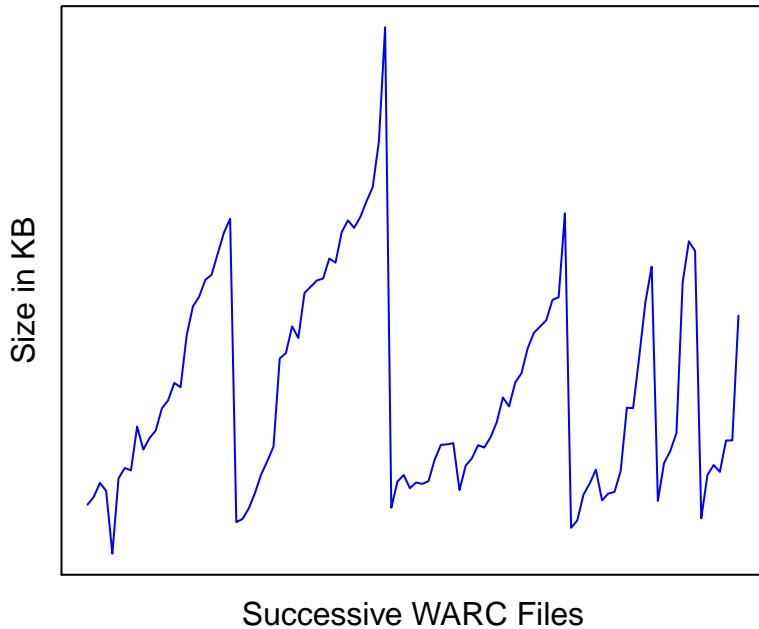
The WARC files generated by the three tools each had different properties. Some quantitative properties of the WARC files generated by the three tools are shown in the table below.

WARC Tool	wget	WARCreate	WebRecorder.io
Number of WARC Files	166	106	113
Average WARC Size	6.23MB	8.27MB	2.41MB
Average Number of Requests	103.3	1048	86.73
Average Number of Responses	103.2	331.1	82.74

The table lists the number of WARC files downloaded using each tool, the average size of each WARC file in megabytes and the average number of HTTP requests and responses in each WARC file.

An interesting pattern emerged when creating WARC files using the WARCreate tool. It appears WARCs generated by the tool get progressively larger as long as the Chrome web browser is running. Once the browser is restarted (because WARCreate is taking too long to generate WARC files), the typical WARC size drops significantly. See the below line graph for an illustration.

Size of Successive WARCreate WARC Files



Each “crash” in the graph corresponds to a restart of the web browser before the next WARC file was generated. This suggests Successive WARC files are including content from previous WARC files, which could explain the relatively high average number of requests and responses in the WARCreate WARC files.

3.2 Playback

Playback of WARC files was conducted using two tools: the Wayback Machine included in the WAIL utility, and the “Replay” feature of WebRecorder.io. Screen captures of the original webpages for each of three sample URIs are shown in the Appendix B. The three sample URIs are listed in Appendix A.

Using the Wayback Machine was a bit of a hassle because I had to be careful to know which set of WARC files were being played-back. There was no clear output of the source filename, so I decided to only load one set of WARCs into the Wayback Machine at a time. The Wayback Machine also did not appear to display pages from a gzip compressed WARC file, although it seemed to index these files correctly. Screen captures of the Wayback Machine’s rendering of the sample WARC files are in the second Appendix C.

WebRecorder.io’s Replay feature simply required me to upload each sample WARC file and click the “Replay” button. A screen capture of each rendering of the sample WARCS is shown in the third Appendix D.

Overall, both playback tools preformed decently. However, all three sample WARC files produced using WebRecorder.io failed catastrophically when rendered in the Wayback Machine. The rendered pages appeared to have a strange encoding composed primarily of characters not displayable by Google Chrome. Other than that combination of WARC generation and playback tools, everything worked well enough to read any text on the page except one URI saved by WARCreate would not work at all when replayed

using WebRecorder.io

Of the 18 rendered WARC files, 12 displayed correctly with the exception of missing multimedia. Four rendered WARCs did not display any of the content as the original webpage, and two appeared to be missing stylesheets and were a chore to read.

3.3 SOLR Queries

I made four queries of the SOLR database after I loaded it with all my WARC files created by wget. The first two queries just test the generic search terms “linux” and “marathon”. The third query searches for only documents that contain “linux” in the document title, as determined by SOLR. The fourth query searches only for documents with the “keywords” field set. This field appeared to be populated only in documents that contained the below HTML tag.

```
<meta name="keywords" content="keyphrases" />
```

In the tag, *keyphrases* is a comma-separated list of keywords, that may contain spaces. A common keyphrase, was “boston marathon”.

The results of the queries are shown abridged in screen captures in the fourth Appendix E. They are also shown in complete form in text in my GitHub repository ⁴ for this project.

⁴https://github.com/jamesbtate/cs851-s15/tree/master/hw2_report/queries

Appendices

A Sample URIs

The three below URIs are the ones used when testing rendering WARC files. They are numbered for their frequency in my list of final URIs in assignment one. URI #1 was the most frequent URI.

1. <https://represent.com/connor>
6. <http://www.nba.com/kings/news/cousins-2015-allstar>
23. <https://www.youtube.com/watch?v=UC49bpMs4zk>

B Original Webpage Screen Captures

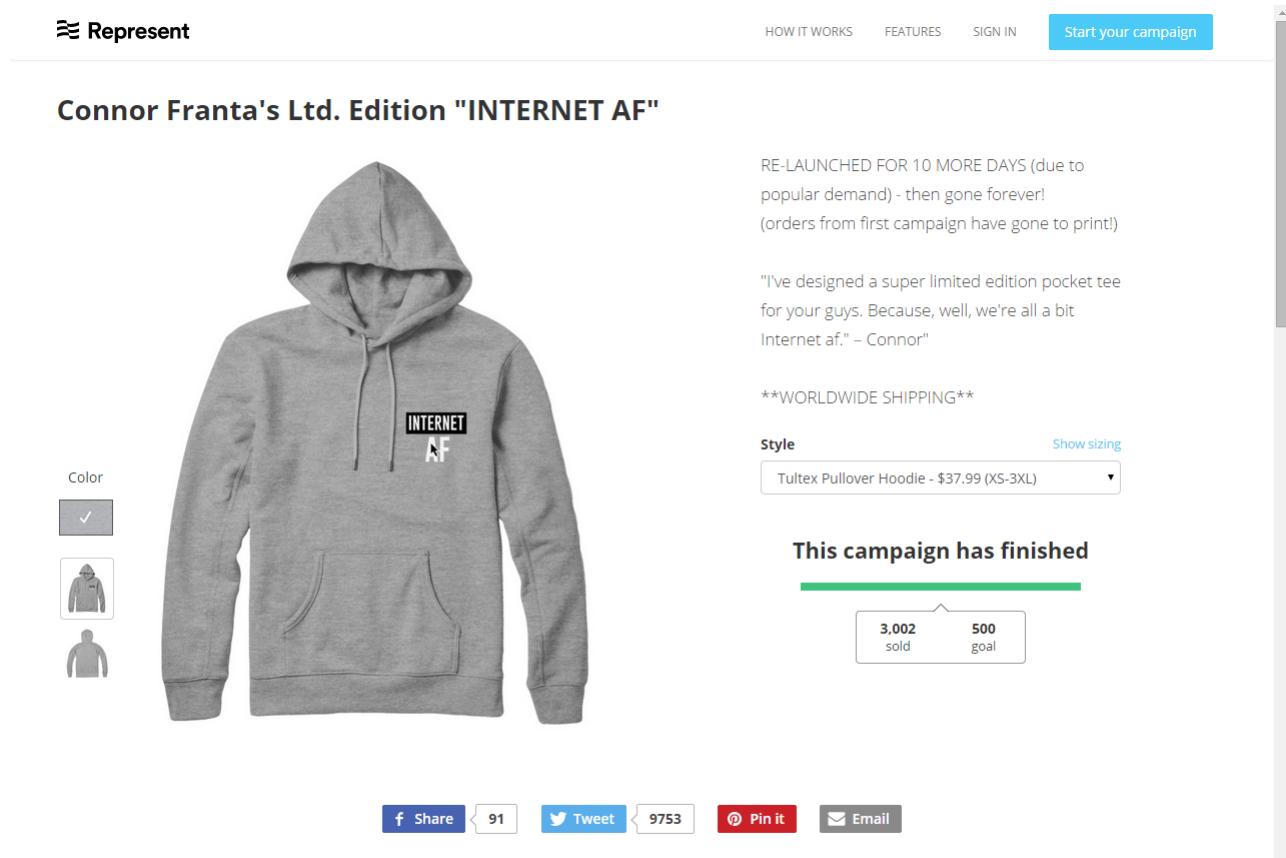


Figure 1: Web browser render of representation of URI #1.

NBA.COM GLOBAL TEAMS D-LEAGUE WNBA STORE

PRESENTED BY KAISER PERMANENTE.

LANGUAGE EN

KINGS Team Tickets Schedule News Multimedia Dancers Shop Community ESC f t g+ d Search |

Next Story>>

RELATED CONTENT

Kings

Sacramento Kings Foundation Accepting Nominations for First Pete Saco Awards March 04, 2015

Gallery: Kicks of the Week 3/4/15 March 04, 2015

DeMarcus Cousins

Kings All-Stars Through The Years February 09, 2015

Internet Reacts to #DMCtoNYC January 30, 2015

Press Release

Darren Collison Undergoes Successful Surgery March 03, 2015

Kings Name Vlade Divac Vice President of Basketball and

SACRAMENTO, Calif. - Sacramento Kings center DeMarcus Cousins was named by NBA Commissioner Adam Silver to replace injured Western Conference All-Star Kobe Bryant of the Los Angeles Lakers, it was announced today by the NBA. Cousins' selection marks the first time a Kings player will participate in an All-Star game since Brad Miller and Peja Stojakovic represented Sacramento in 2004. He will become the sixth player during the Sac-era, and 23rd in franchise history to represent the Kings in the All-Star Game.

"I'm extremely excited to play in my first All-Star game," said Cousins. "I appreciate the recognition and want to also thank my teammates and the fans for their support throughout the season. Representing the Kings and the city of Sacramento is a great honor."

"Everyone in the Kings organization is thrilled for DeMarcus and happy that his diligence and commitment to becoming

Posted: Jan 30, 2015 f g+ t d

Figure 2: Web browser render of representation of URI #6.

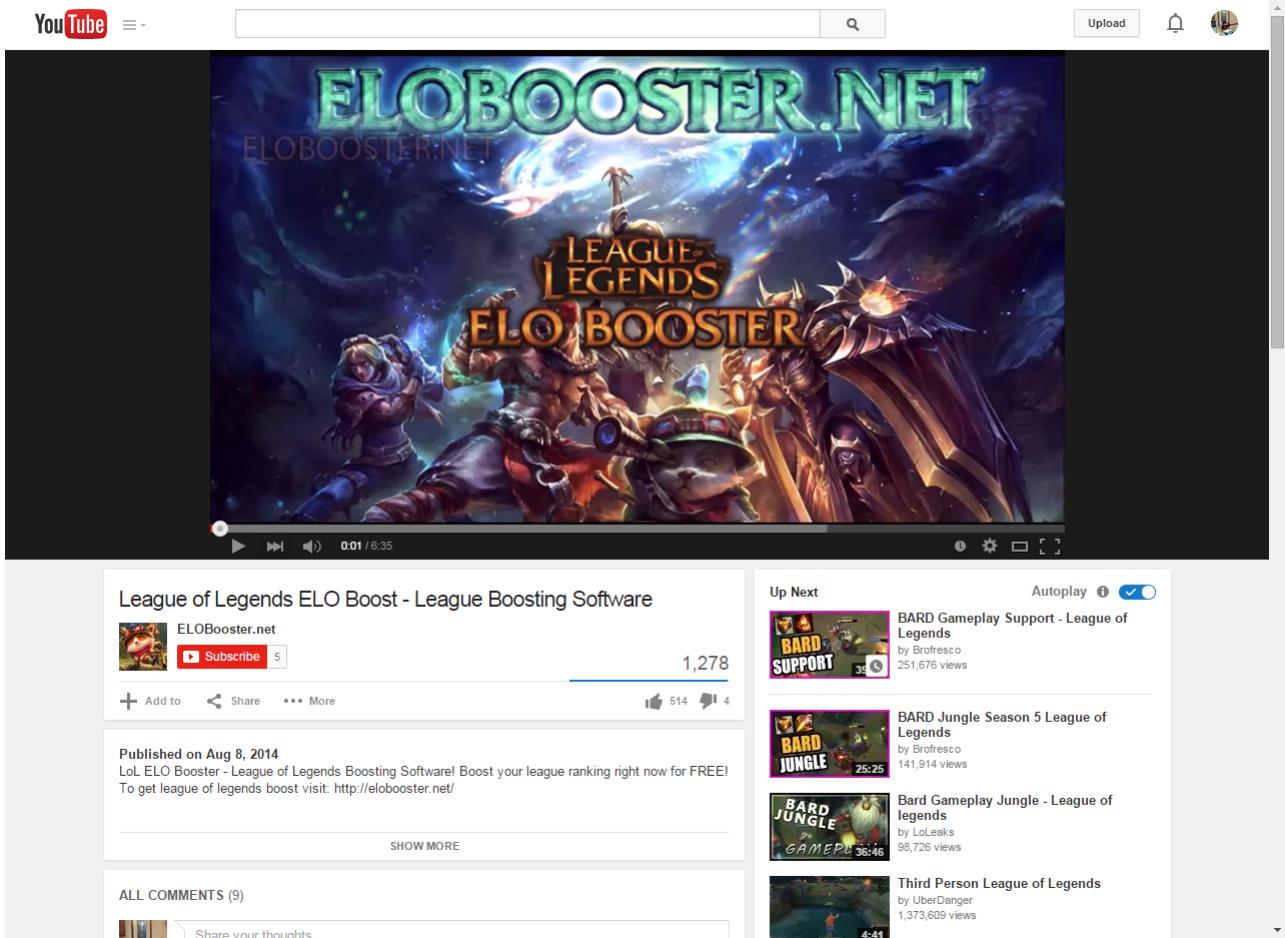


Figure 3: Web browser render of representation of URI #23.

C Wayback Machine Playback Screen Captures

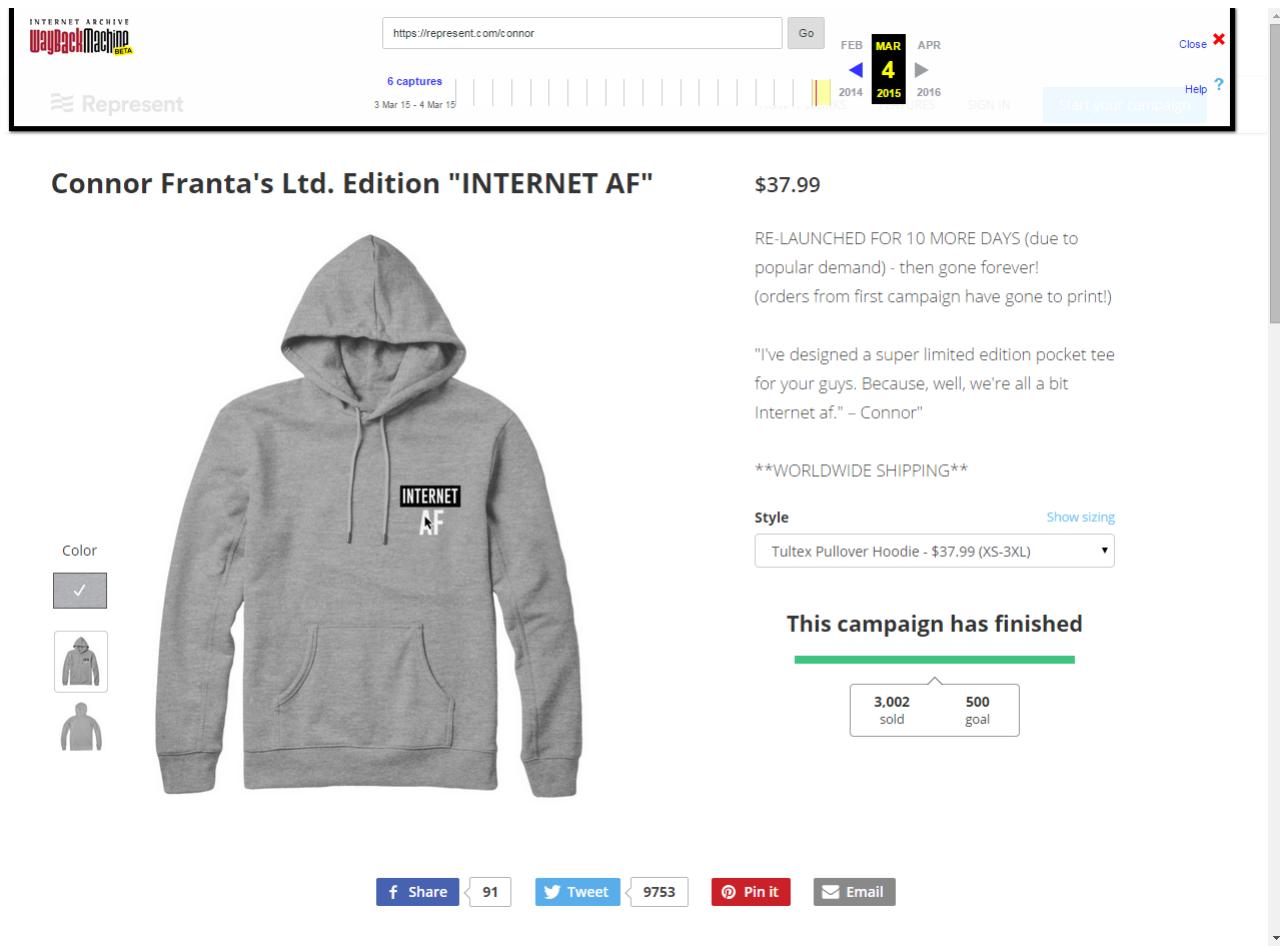


Figure 4: Wayback Machine playback of wget WARC of URI #1.

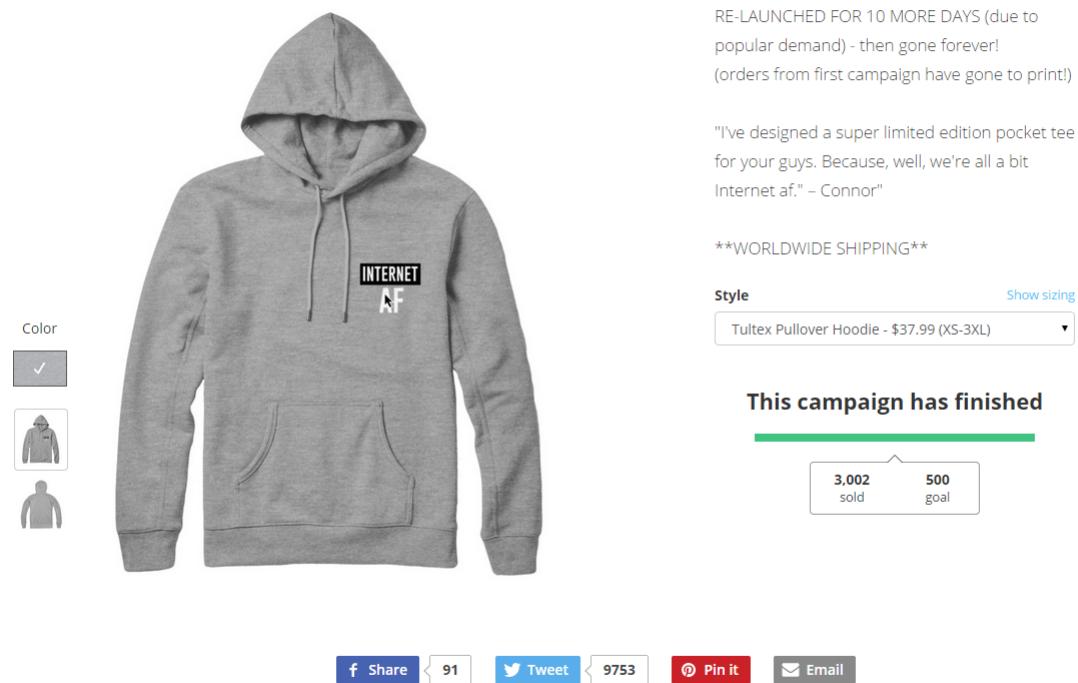
Connor Franta's Ltd. Edition "INTERNET AF"

Figure 5: Wayback Machine playback of WARCreate WARC of URI #1.

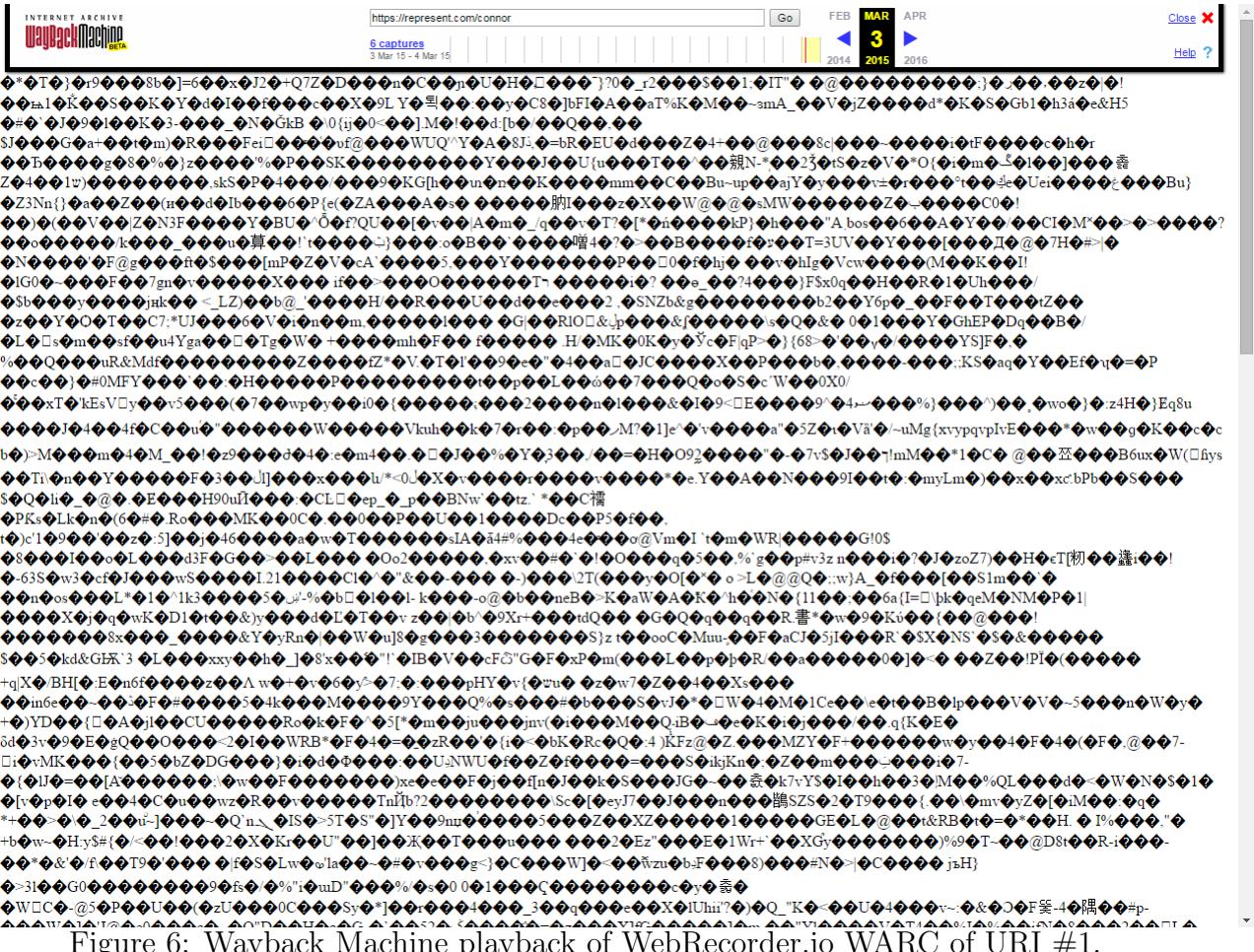


Figure 6: Wayback Machine playback of WebRecorder.io WARC of URI #1.

NBA.COM GLOBAL TEAMS D-LEAGUE WNBA STORE

PRESENTED BY KAISER PERMANENTE.

KINGSTeam Tickets Schedule News Multimedia More LANGUAGE EN

Search |

DeMarcus Cousins Named to 2015 NBA All-Star Team

SACRAMENTO, Calif. - Sacramento Kings center DeMarcus Cousins was named by NBA Commissioner Adam Silver to replace injured Western Conference All-Star Kobe Bryant of the Los Angeles Lakers, it was announced today by the NBA. Cousins' selection marks the first time a Kings player will participate in an All-Star game since Brad Miller and Peja Stojakovic represented Sacramento in 2004. He will become the sixth player during the Sac-era, and 23rd in franchise history to represent the Kings in the All-Star Game.

"I'm extremely excited to play in my first All-Star game," said Cousins. "I appreciate the recognition and want to also thank my teammates and the fans for their support throughout the season. Representing the Kings and the city of Sacramento is a great honor."

Waiting for s869.t.eloqua.com...

Posted: Jan 30, 2015

Next Story>>

RELATED CONTENT

Kings

Sacramento Kings Foundation Accepting Nominations for First Pete Saco Awards March 04, 2015

Gallery: Kicks of the Week 3/4/15 March 04, 2015

DeMarcus Cousins

Kings All-Stars Through The Years February 09, 2015

Internet Reacts to #DMCtoNYC January 30, 2015

Press Release

Darren Collison Undergoes Successful Surgery March 03, 2015

Kings Name Vlade Divac Vice

Figure 7: Wayback Machine playback of wget WARC of URI #6.

NBA.COM GLOBAL TEAMS D-LEAGUE WNBA STORE PRESENTED BY LANGUAGE EN

Team Tickets Schedule News More

Next Story>>

RELATED CONTENT

Kings

- Vivek Recalls Trip with Obama
March 03, 2015
- Darren Collison Undergoes Successful Surgery
March 03, 2015

DeMarcus Cousins

- Kings All-Stars Through The Years
February 09, 2015
- Internet Reacts to #DMCtoNYC
January 30, 2015

Press Release

- VLADE DIVAC Kings Name Vlade Divac Vice President of Basketball and Franchise Operations
March 03, 2015
- CITY OF SACRAMENTO AND SACRAMENTO KINGS ANNOUNCE

DeMarcus Cousins Named to 2015 NBA All-Star Team

SACRAMENTO, Calif. - Sacramento Kings center DeMarcus Cousins was named by NBA Commissioner Adam Silver to replace injured Western Conference All-Star Kobe Bryant of the Los Angeles Lakers, it was announced today by the NBA. Cousins' selection marks the first time a Kings player will participate in an All-Star game since Brad Miller and Peja Stojakovic represented Sacramento in 2004. He will become the sixth player during the Sac-era, and 23rd in franchise history to represent the Kings in the All-Star Game.

"I'm extremely excited to play in my first All-Star game," said Cousins. "I appreciate the recognition and want to also thank my teammates and the fans for their support throughout the season. Representing the Kings and the city of Sacramento is a great honor."

"Everyone in the Kings organization is thrilled for DeMarcus and happy that his diligence and commitment to becoming

Figure 8: Wayback Machine playback of WARCcreate WARC of URI #6.

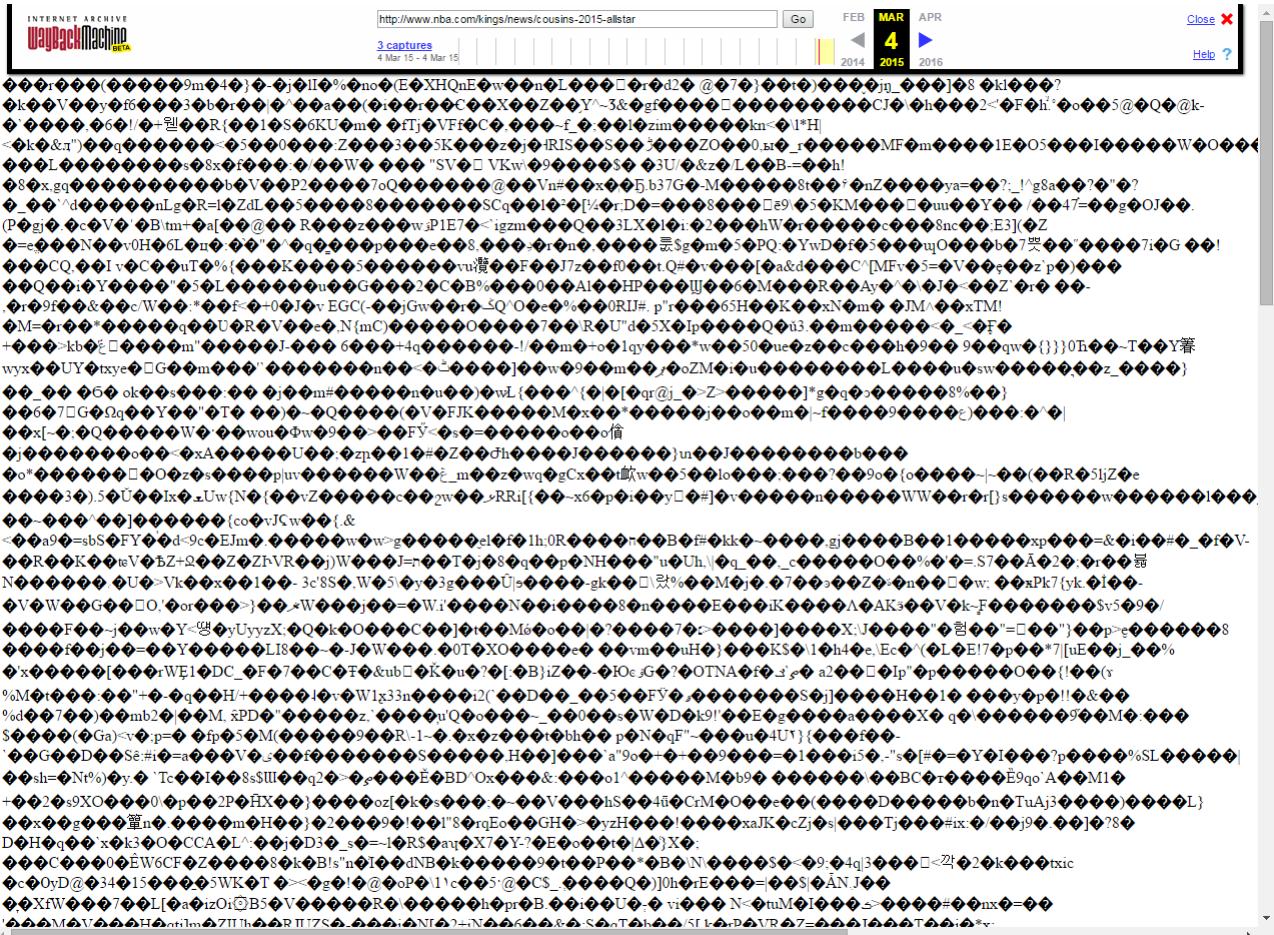


Figure 9: Wayback Machine playback of WebRecorder.io WARC of URI #6.

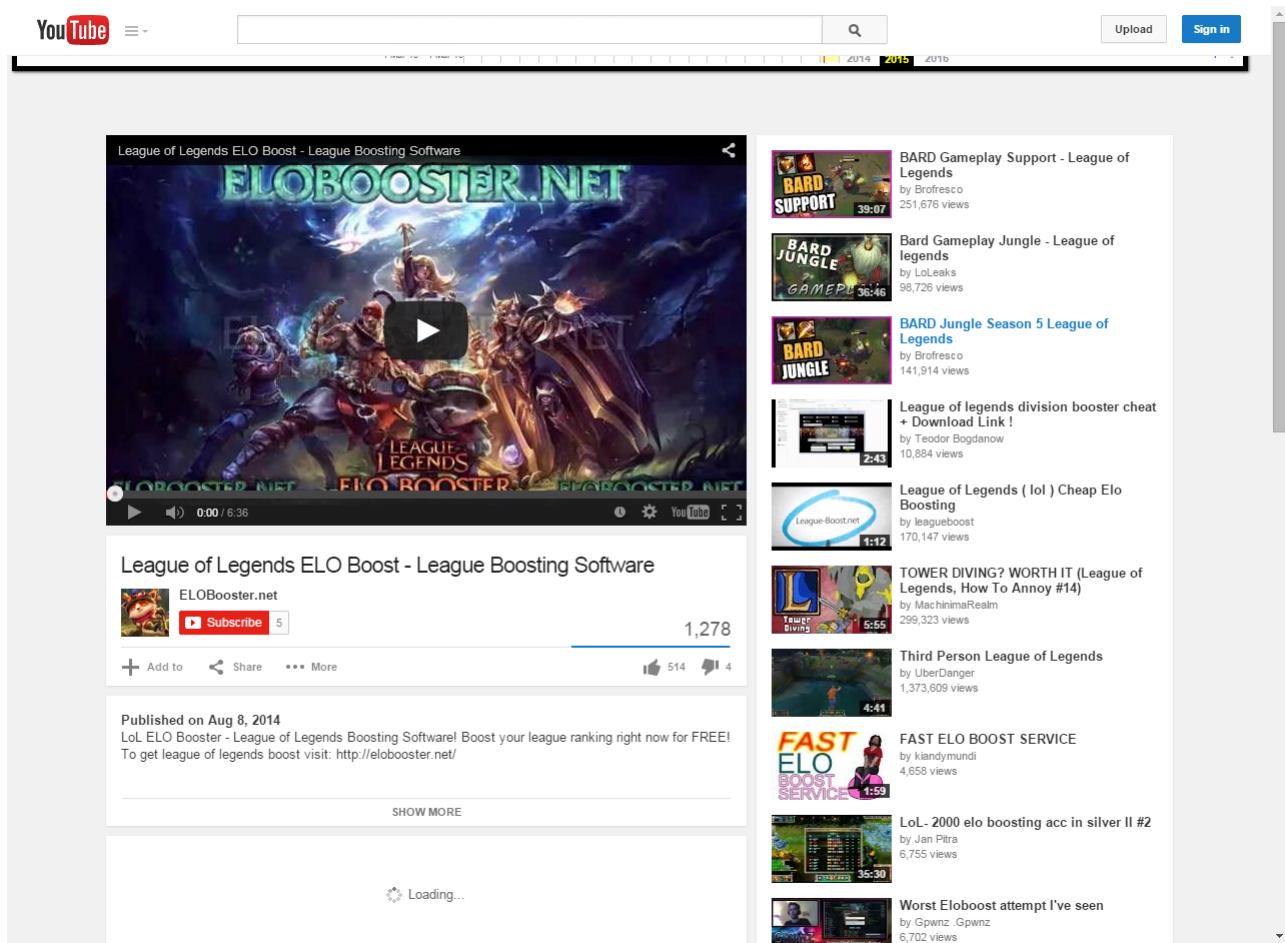


Figure 10: Wayback Machine playback of wget WARC of URI #23.

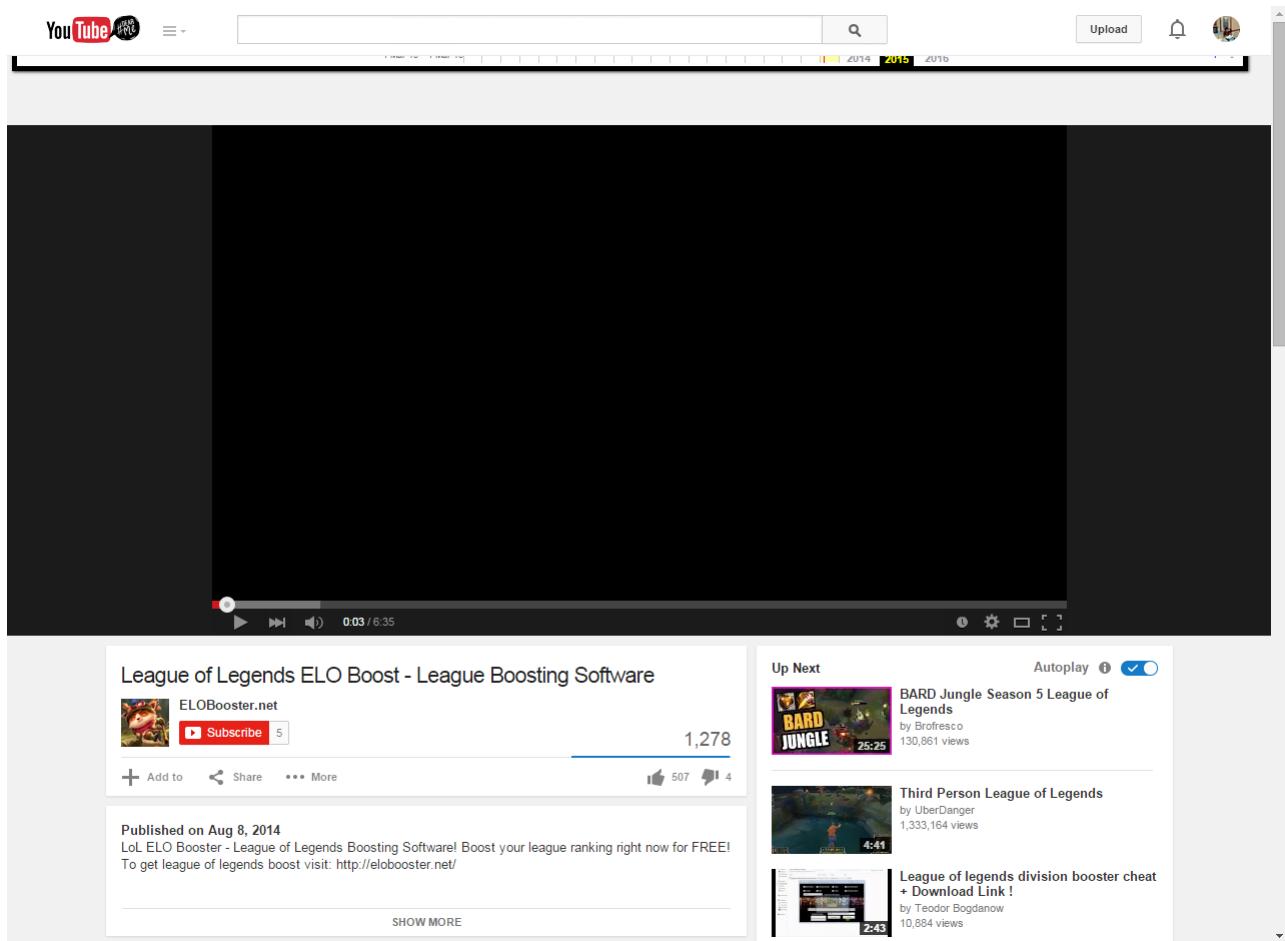


Figure 11: Wayback Machine playback of WARCreate WARC of URI #23.



Figure 12: Wayback Machine playback of WebRecorder.io WARC of URI #23.

D WebRecorder.io Playback Screen Captures

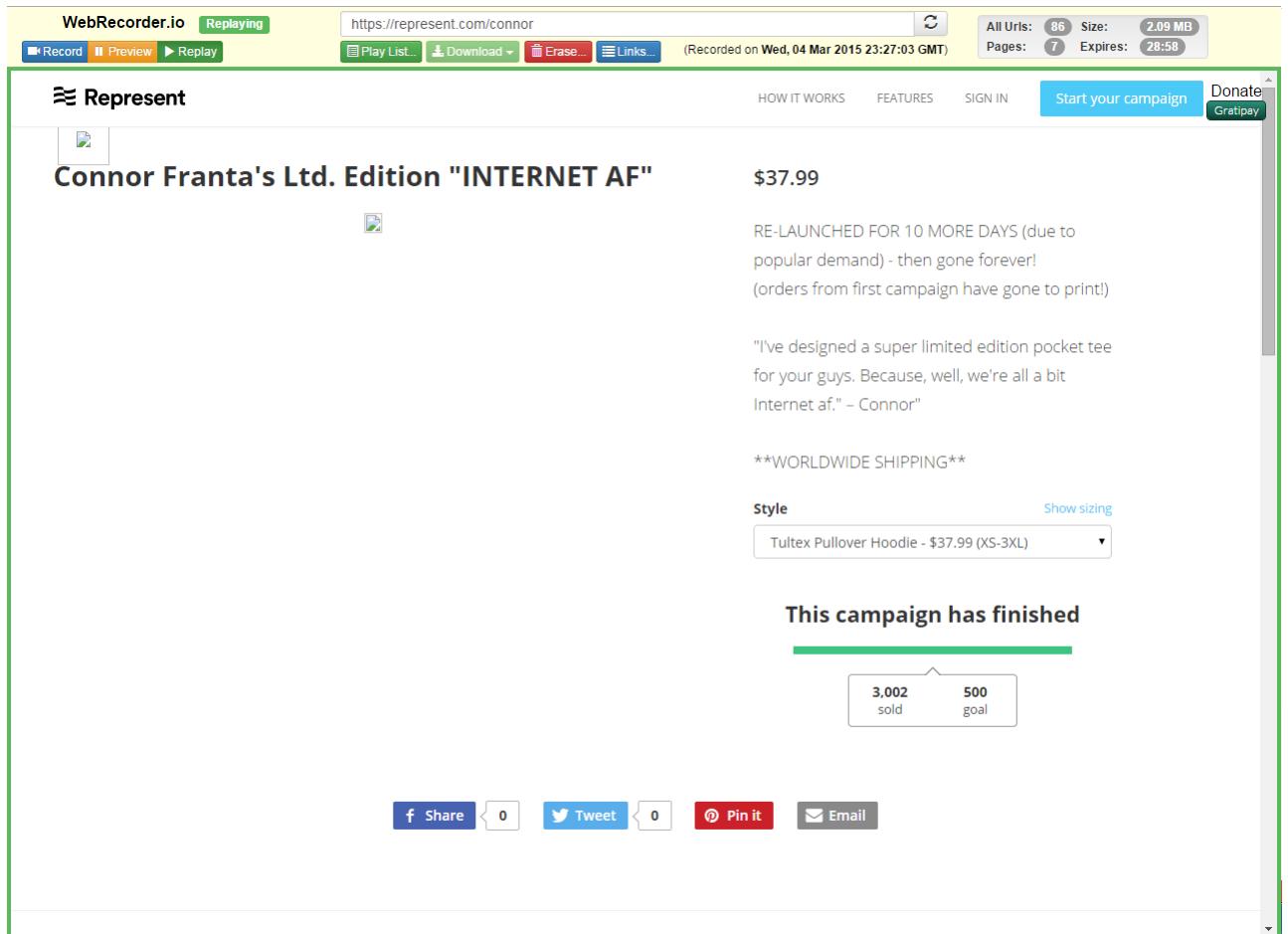


Figure 13: WebRecorder.io playback of wget WARC of URI #1.

WebRecorder.io REPLAY Expires In: 28:56

Note: Some or all of this archived data was not created in WebRecorder.io and its authenticity can not be verified by WebRecorder.io

Recorded Pages [Url Search](#) Total Archive Size: 2.92 MB

Search:

Showing 0 to 0 of 0 entries

Page	Recorded At
No pages have been recorded yet. Start Recording!	

For any questions, comments, inquiries or feature requests,
contact: info@webrecorder.io

Donations graciously accepted: [Gratipay](#)

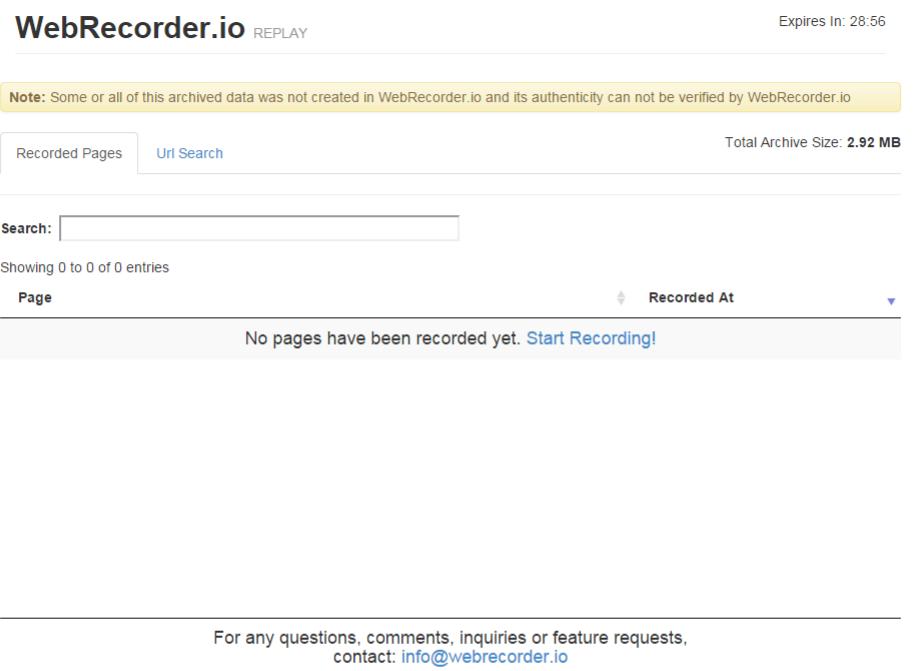
The screenshot shows the WebRecorder.io interface. At the top, it displays the title "WebRecorder.io REPLAY" and a note about the authenticity of the archived data. It also shows the total archive size as 2.92 MB. Below this is a search bar and two navigation links: "Recorded Pages" and "Url Search". A message indicates that no pages have been recorded yet, with a link to "Start Recording!". At the bottom, there is contact information and a link to Gratipay for donations.

Figure 14: WebRecorder.io playback of WARCreate WARC of URI #1.

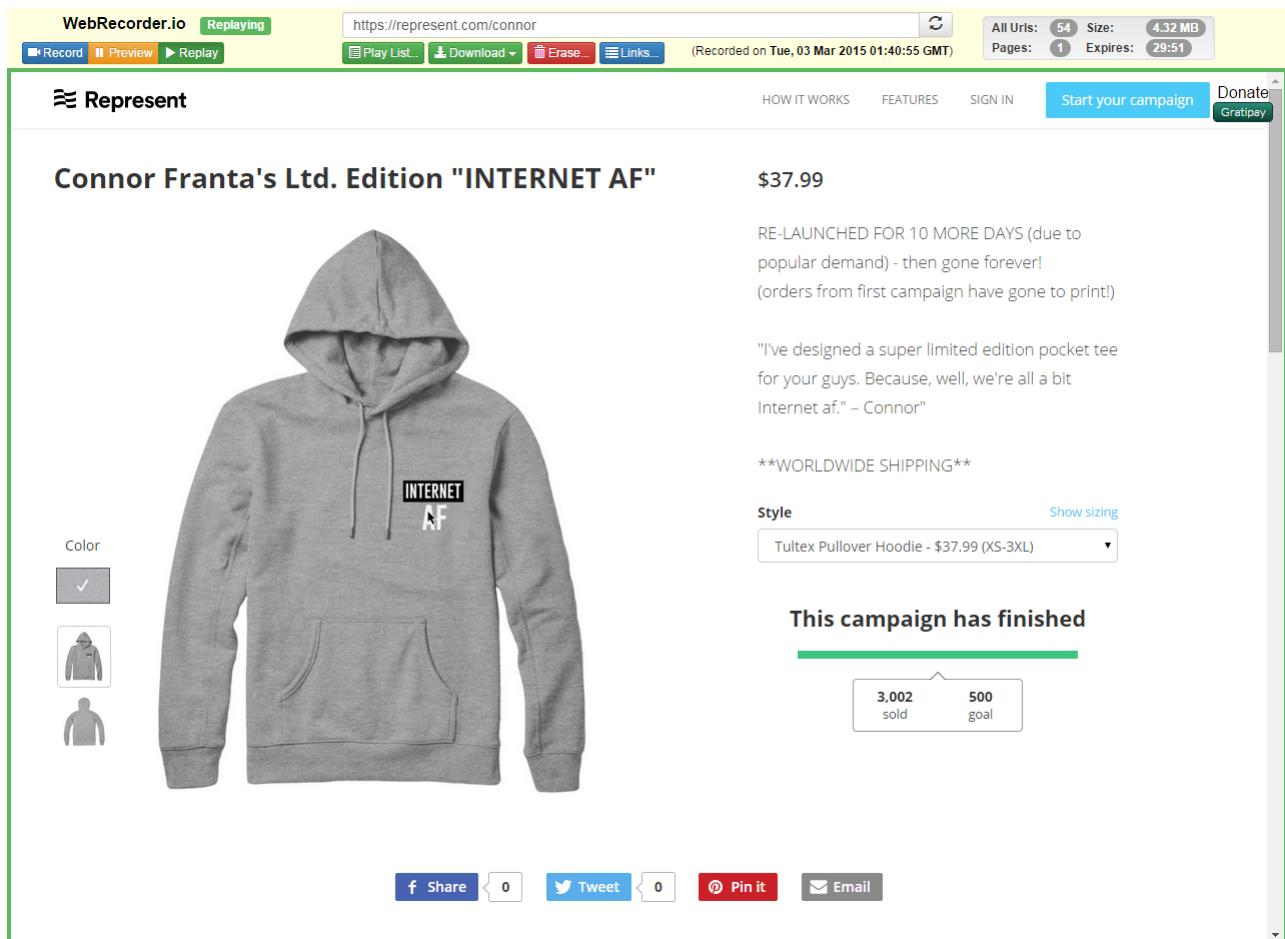


Figure 15: WebRecorder.io playback of WebRecorder.io WARC of URI #1.



Figure 16: WebRecorder.io playback of wget WARC of URI #6.

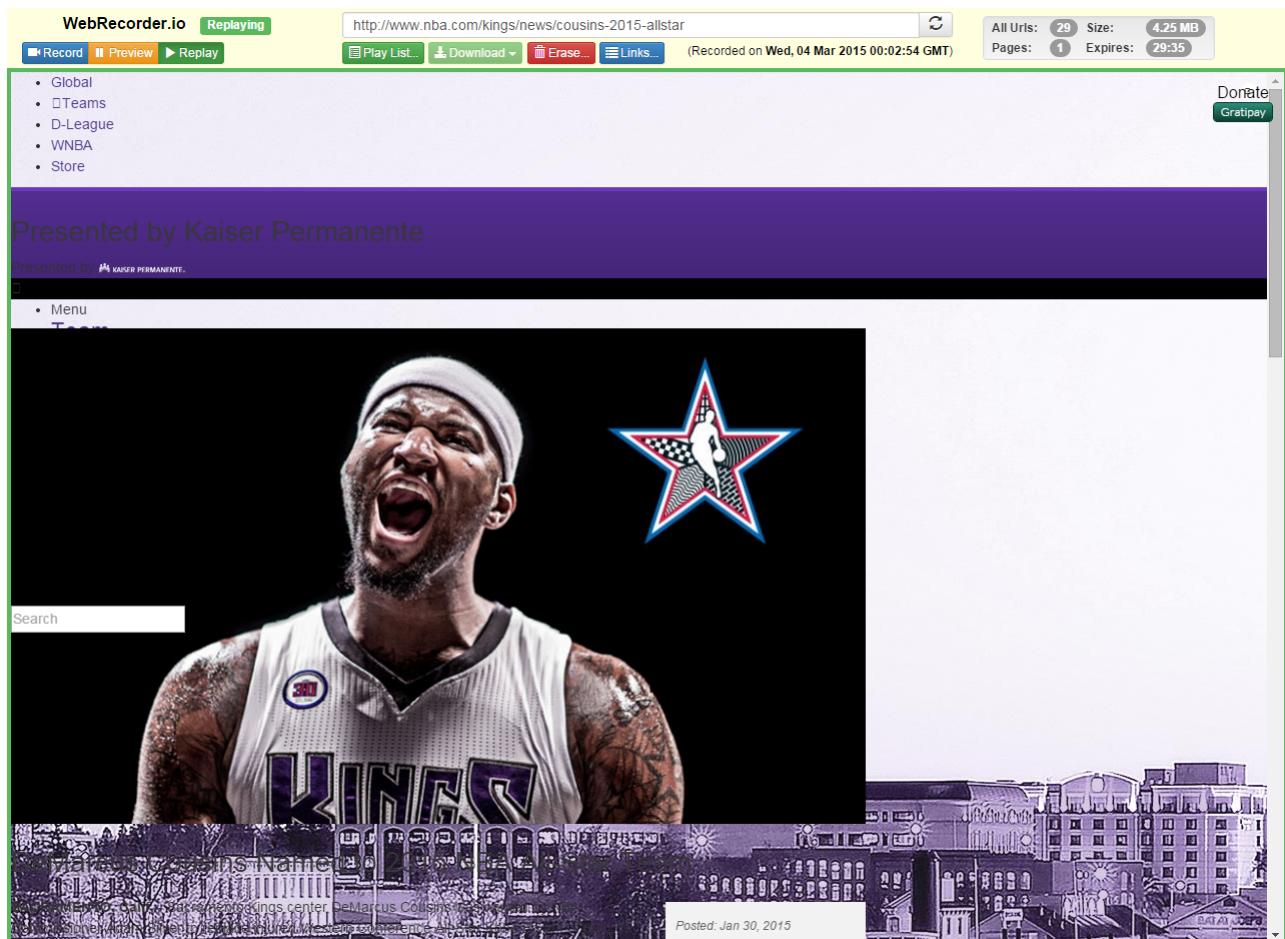


Figure 17: WebRecorder.io playback of WARC of URI #6.



Figure 18: WebRecorder.io playback of WebRecorder.io WARC of URI #6.

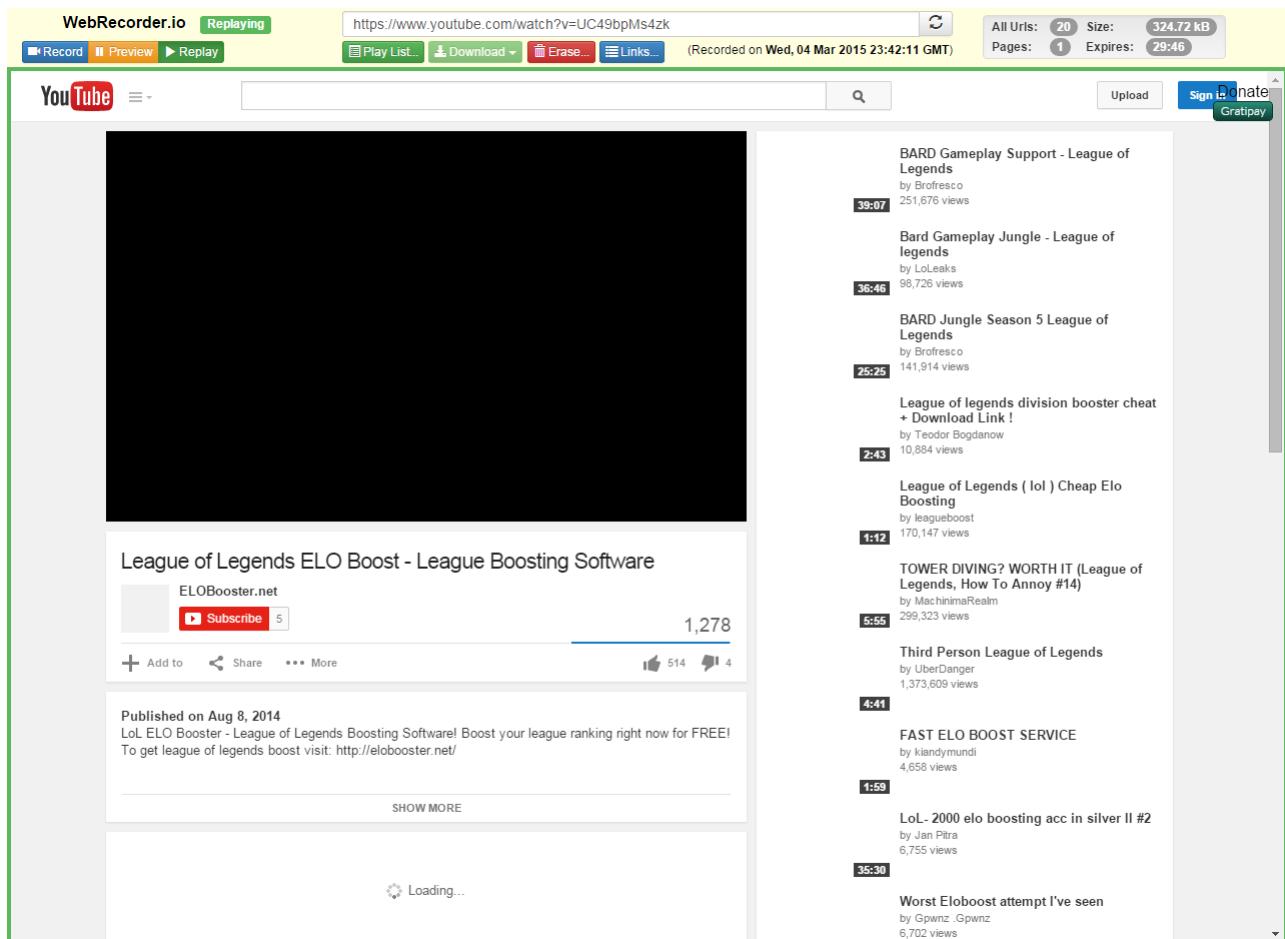


Figure 19: WebRecorder.io playback of wget WARC of URI #23.

WebRecorder.io **Replaying** <https://www.youtube.com/watch?v=UC49bpMs4zk> All URLs: 447 Size: 15.55 MB
 Recorded on Wed, 04 Mar 2015 00:17:47 GMT
 Pages: 1 Expires: 29:41

WebRecorder.io

The url
https://plus.google.com/_/notifications/frame?origin=https%3A%2F%2Fwww.youtube.com&US&jsh=m%3B%2F%2Fscs%2Fabc-static%2F%2Fjs%2Fk%3Dgapi.gapi.en.ehm/-L6Y_snQHC2igwIIAwLSbYBg has not been recorded yet.

[Record it now!](#)

(Note: If you just recorded this page moments ago, please wait a few seconds and [reload this page](#).)

Skip navigation


 Upload 0
 jamesbate@gmail.com
[Change](#)
 James Tate

Creator Studio
 Other accounts

 jessicagtate@gmail.com
 jessicagtate@gmail.com
[Add account](#) [Sign out](#)
 Search

 What to Watch
 My Channel
 My Subscriptions 14
 History
 Watch Later 8
 Playlists
 Liked videos

Subscriptions

- YouAlwaysWin 5
- TheDiamondMinecart 10
- HuskyStarcraft
- TobyGames 19
- Husky Plays

Figure 20: WebRecorder.io playback of WARCreate WARC of URI #23.

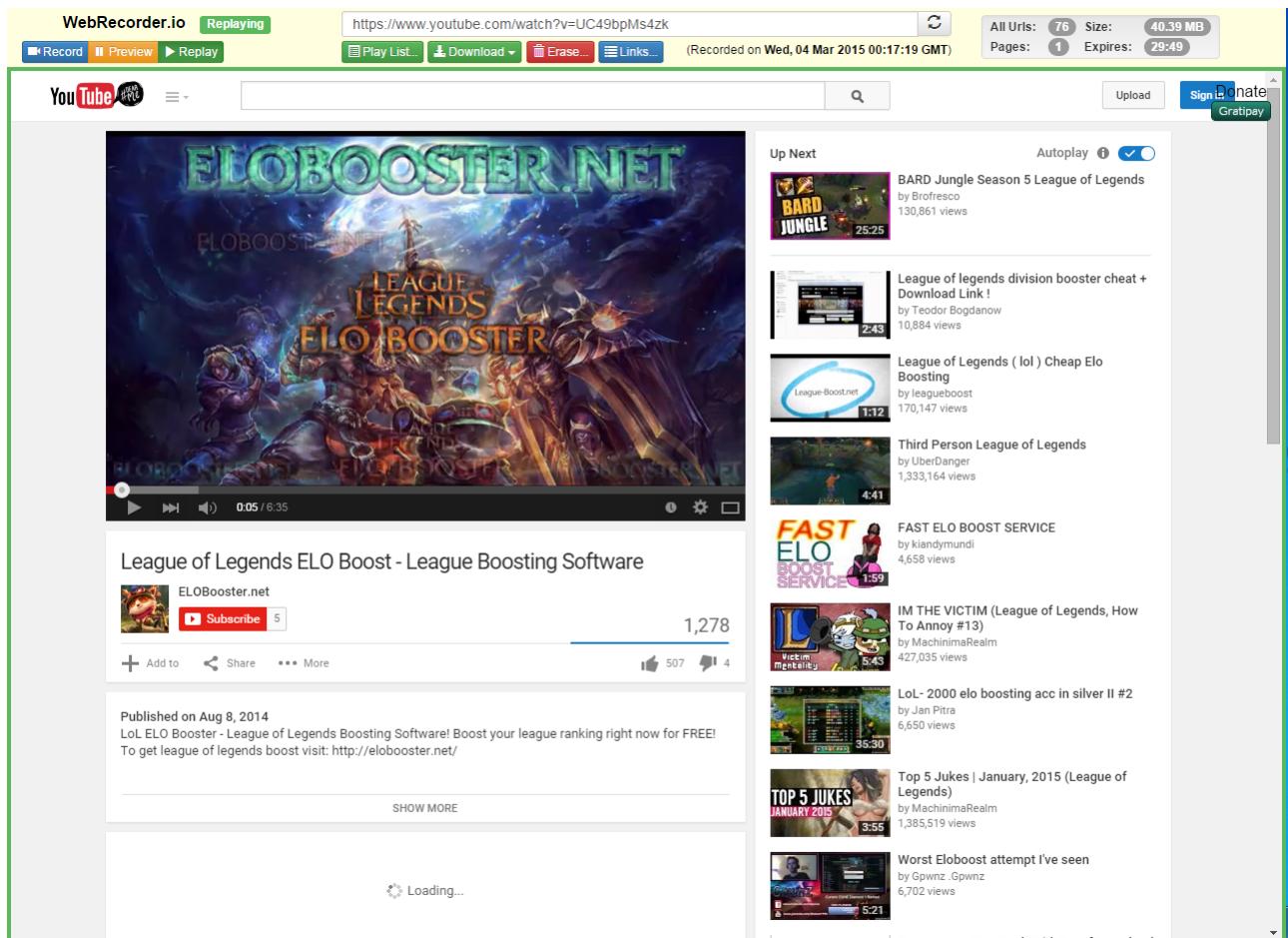


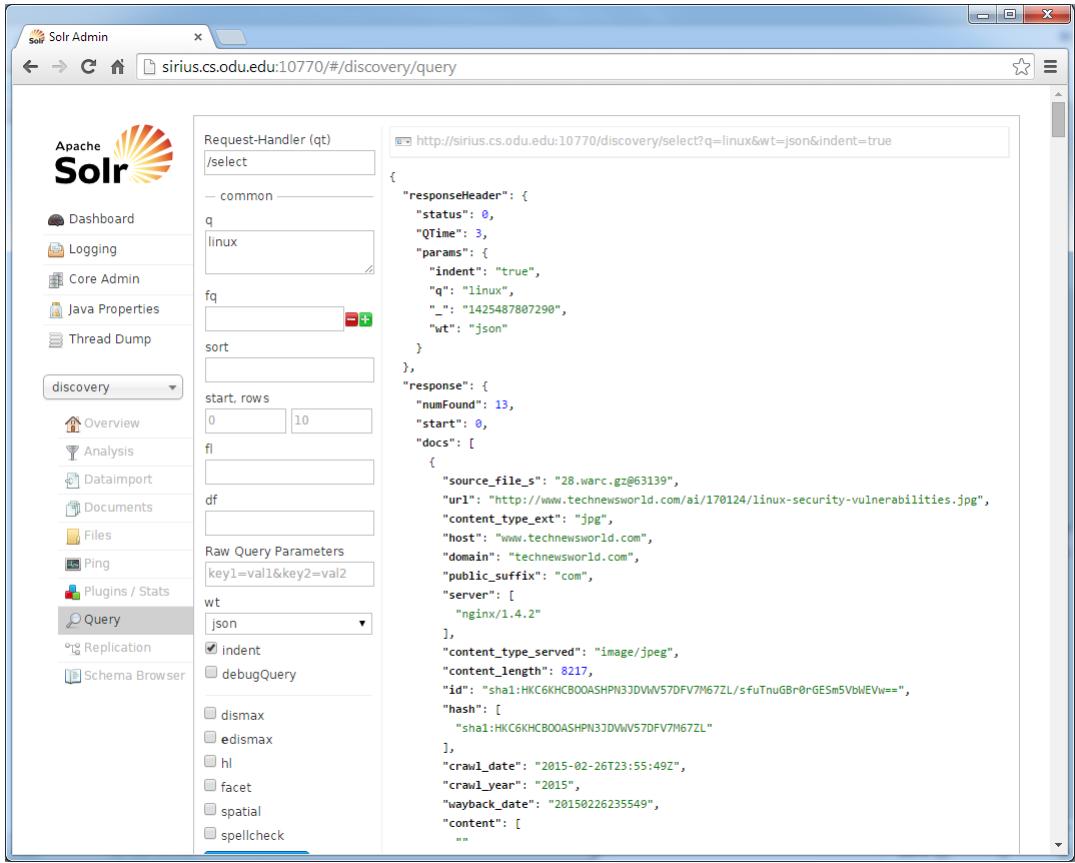
Figure 21: WebRecorder.io playback of WebRecorder.io WARC of URI #23.

E SOLR Queries

The screenshot shows the Apache Solr Admin interface. On the left, there's a sidebar with various navigation links like Dashboard, Logging, Core Admin, Java Properties, Thread Dump, discovery (selected), Query, Replication, and Schema Browser. The main area has a title bar "sirius.cs.odu.edu:10770/#/discovery/query". Below the title bar, there's a "Request-Handler (qt)" dropdown set to "/select". The query parameters are: q: marathon, fq: (empty), sort: (empty), start: 0, rows: 10, fl: (empty), df: (empty). Under "Raw Query Parameters", there's a key1=val1&key2=val2 entry. The "wt" dropdown is set to "json". A checkbox for "indent" is checked. Below these are checkboxes for dismax, edismax, hl, facet, spatial, and spellcheck. The right side of the interface displays the JSON response for the query. The URL in the browser is http://sirius.cs.odu.edu:10770/discovery/select?q=marathon&wt=json&indent=true. The response shows a status of 0, QTime of 3, and 15 documents found. One document is shown in detail:

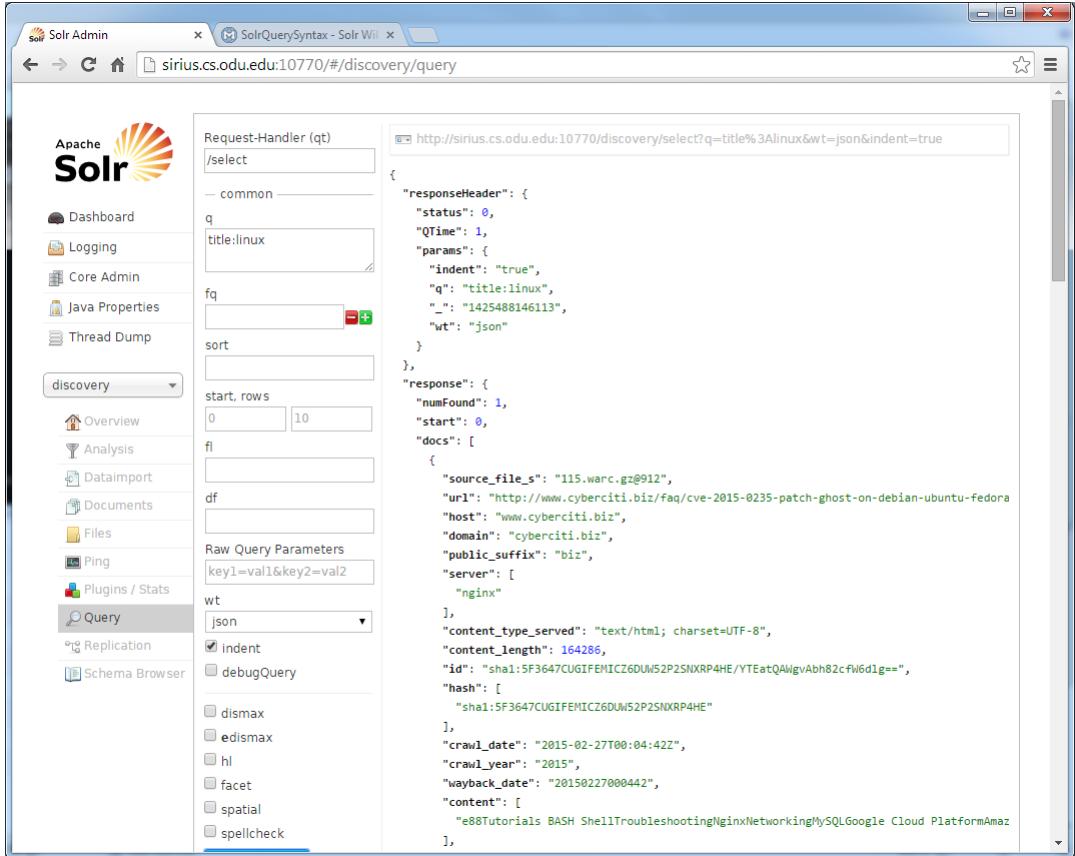
```
{  
  "responseHeader": {  
    "status": 0,  
    "QTime": 3,  
    "params": {  
      "indent": "true",  
      "q": "marathon",  
      "_": "1425487429572",  
      "wt": "json"  
    }  
  },  
  "response": {  
    "numFound": 15,  
    "start": 0,  
    "docs": [  
      {  
        "source_file_s": "102.warc.gz@966",  
        "url": "http://www.washingtonpost.com/news/post-nation/wp/2015/01/28/a-blizzard-myster",  
        "host": "www.washingtonpost.com",  
        "domain": "washingtonpost.com",  
        "public_suffix": "com",  
        "content_type_served": "text/html; charset=UTF-8",  
        "server": [  
          "nginx"  
        ],  
        "content_length": 98045,  
        "id": "sha1:ZL6LB73ANDBFIE2LQNM6DR7KB64Q3PVY/o6EbxEVjPf+Jg1bh6I7FxA==",  
        "hash": [  
          "sha1:ZL6LB73ANDBFIE2LQNM6DR7KB64Q3PVY"  
        ],  
        "crawl_date": "2015-02-27T00:02:27Z",  
        "crawl_year": "2015",  
        "wayback_date": "20150227000227",  
        "content": [  
          "A blizzard mystery: Who shoveled the Boston Marathon finish line? - The Washington",  
        ]  
      }  
    ]  
  }  
}
```

Figure 22: SOLR query for word “marathon”.



The screenshot shows the Apache Solr Admin interface. On the left, there's a sidebar with various navigation options like Dashboard, Logging, Core Admin, Java Properties, Thread Dump, discovery (selected), Overview, Analysis, Dataimport, Documents, Files, Ping, Plugins / Stats, Query (selected), Replication, and Schema Browser. The main panel has a title bar "Solr Admin" and a URL "sirius.cs.odu.edu:10770/#/discovery/query". It contains a "Request-Handler (qt) /select" form with fields for "q" (set to "linux"), "fq", "sort", "start", "rows", "fl", "df", and "Raw Query Parameters" (key1=val1&key2=val2). Below the form are checkboxes for "wt" (set to "json") and "indent" (checked), and several other unselected checkboxes: dismax, edismax, hl, facet, spatial, and spellcheck. To the right of the form is a large text area showing the JSON response to the query. The response includes the "responseHeader" (status 0, QTime 3, params with indent=true, q=linux, _id=1425487807290, wt=json), the "response" (numFound 13, start 0, docs array), and the first document's details: source_file_s (28.warc.gz@63139), url (http://www.technewsworld.com/ai/170124/linux-security-vulnerabilities.jpg), content_type_ext (jpg), host (www.technewsworld.com), domain (technewsworld.com), public_suffix (com), server (nginx/1.4.2), content_type_served (image/jpeg), content_length (8217), id (sha1:HKK6KHCBOOAASHPN3JDWV57DFV7M67ZL/sfuTnuGBr0rGESm5VbhEVw==), hash (sha1:HKK6KHCBOOAASHPN3JDWV57DFV7M67ZL), crawl_date (2015-02-26T23:55:49Z), crawl_year (2015), wayback_date (20150226235549), and content (multiple lines of file content).

Figure 23: SOLR query for word “linux”.



This screenshot is similar to Figure 23, showing the Solr Admin interface. The sidebar and form are identical, with "Query" selected. The "q" field is now set to "title:linux". The JSON response on the right shows one result: a document with source_file_s (115.warc.gz@912), url (http://www.cyberciti.biz/faq/cve-2015-0235-patch-ghost-on-debian-ubuntu-fedora), host (www.cyberciti.biz), domain (cyberciti.biz), public_suffix (biz), server (nginx), content_type_served (text/html; charset=UTF-8), content_length (164286), id (sha1:5F3647CUGIFEMICZ60UW52P2SNXRP4HE/YTEatQAlgvAbh82cfW6d1g==), hash (sha1:5F3647CUGIFEMICZ60UW52P2SNXRP4HE), crawl_date (2015-02-27T00:04:42Z), crawl_year (2015), wayback_date (2015022700442), and content (multiple lines of file content related to BASH Shell Troubleshooting Nginx Networking MySQL Google Cloud Platform Amazon).

Figure 24: SOLR query for documents with “linux” in the title.

The screenshot shows the Apache Solr Admin interface. On the left, there's a sidebar with various navigation options like Dashboard, Logging, Core Admin, Java Properties, Thread Dump, discovery (selected), Query (selected), Replication, and Schema Browser. The main area has tabs for 'Solr Admin' and 'SolrQuerySyntax - Solr Wiki'. The URL in the browser is `sirius.cs.odu.edu:10770/#/discovery/query`. The main panel is titled 'Request-Handler (qt)' and contains fields for 'q' (set to 'keywords:[* TO *]'), 'fq' (empty), 'sort' (empty), 'start', 'rows' (set to 10), 'fl' (empty), and 'df' (empty). Below these are 'Raw Query Parameters' with 'key1=val1&key2=val2' and 'wt' set to 'json'. There are also checkboxes for 'indent' (checked) and 'debugQuery' (unchecked). A list of checkboxes on the right includes 'dismax', 'edismax', 'hl', 'facet', 'spatial', and 'spellcheck'. To the right of the form is a large text area showing the JSON response from the Solr server. The response header includes status 0, QTime 0, and params for indent, q, fq, start, rows, fl, df, wt, and key parameters. The response body shows a list of 60 documents, each with fields like source_file_s, url, host, domain, public_suffix, content_type_served, server, content_length, id, hash, crawl_date, crawl_year, wayback_date, and content. One document is highlighted with a red border.

```

{
  "responseHeader": {
    "status": 0,
    "QTime": 0,
    "params": {
      "indent": "true",
      "q": "keywords:[* TO *]",
      "fq": "1425488314857",
      "start": 0,
      "rows": 10
    }
  },
  "response": {
    "numFound": 60,
    "start": 0,
    "docs": [
      {
        "source_file_s": "102.warc.gz@966",
        "url": "http://www.washingtonpost.com/news/post-nation/wp/2015/01/28/a-blizzard-mystery-snow-shoveling-at-the-boston-marathon-finish-line/",
        "host": "www.washingtonpost.com",
        "domain": "washingtonpost.com",
        "public_suffix": "com",
        "content_type_served": "text/html; charset=UTF-8",
        "server": [
          "nginx"
        ],
        "content_length": 98045,
        "id": "sha1:ZL6LB73ANDBFIE2LQNM6DR7KB64Q3PVY/o6EbxIVjPf+Jg1bh6I7FxA==",
        "hash": [
          "sha1:ZL6LB73ANDBFIE2LQNM6DR7KB64Q3PVY"
        ],
        "crawl_date": "2015-02-27T00:02:27Z",
        "crawl_year": "2015",
        "wayback_date": "20150227000227",
        "content": [
          "A blizzard mystery: Who shoveled the Boston Marathon finish line? - The Washington"
        ]
      }
    ]
  }
}

```

Figure 25: SOLR query for documents with keyword field set..