InterPlanetary Wayback: The Permanent Web Archive

Sawood Alam, Mat Kelly, Michele C. Weigle, and Michael L. Nelson
Old Dominion University
Department of Computer Science
Norfolk, VA 23529, USA
{salam,mkelly,mweigle,mln}@cs.odu.edu

Project Description

To facilitate permanence and collaboration in web archives, we built InterPlanetary Wayback (*ipwb*) to disseminate the contents of WARC files into the content addressable Inter-Planetary File System (IPFS)[1]. Our software prototype[2] partitions, indexes, and deploys the payloads of archival data records into the IPFS peer-to-peer "permanent web" for sharing and offsite redundant preservation and replay.

We leverage the Python-based pywb[3] for its support of CDXJ and WARC[4] manipulation libraries already included in the replay system's codebase. CDXJ allows *ipwb* to use the arbitrary JSON data field to store metadata about WARC records within IPFS (i.e., the content digest needed for lookup in IPFS). A modified pywb allows the Memento[5] URI-R and Memento-Datetime to resolve from the CDXJ to WARC content in IPFS instead of a local WARC file, as normally occurs with pywb.

The *ipwb* prototype extracts the HTTP response body (payload) from the records within a WARC file, which it then uses IPFS to generate a signature uniquely representative of this content. This payload is pushed into the IPFS system and retrieved at a later date when the URI-M is queried. Content addressability allows for network-wide deduplication of the content. The digest of the content is used as the key to locate the content in the peer-to-peer network.

Implementation

Each line in the CDXJ file holds one index record (Figure 1). The line begins with a SURTed URI[6] and datetime followed by a single-line JSON block that stores reference to the content and other arbitrary metadata. We utilize the last field in a CDXJ record (a JSON object) to store the HTTP response headers and payload digests, original status code when the URI-R was crawled, the MIME-type of the content, and a UUID to identify a memento. The two digests that are used to locate the contents from the IPFS system and build the response are encoded into a single field called "locator" using a URN scheme[7].

In designing *ipwb*, it was critical to consider the HTTP

[1] http://arxiv.org/abs/1407.3561
[2] https://github.com/oduwsdl/ipwb
[3] https://github.com/ikreymer/pywb
[4] http://www.digitalpreservation.gov/formats/fdd/fdd000236.shtml
[5] https://tools.ietf.org/html/rfc7089
[6] http://crawler.archive.org/articles/user_manual/glossary.html#surt
[7] https://www.w3.org/TR/uri-clarification/

```
SURT_URI DATETIME {
  "id": "WARC-Record-ID",
  "url": "ORIGINAL_URI",
  "status": "3-DIGIT_HTTP_STATUS",
  "mime": "Content-Type",
  "locator": "urn:ipfs/HEADER_DIGEST/PAYLOAD_DIGEST"
}
```

Figure 1: A single-line CDXJ record template, shown on multiple lines for readability.

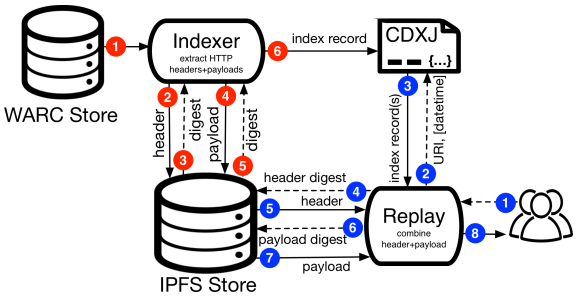

Figure 2: The *ipwb* indexer and replay workflow.

headers returned at crawl time separately from the HTTP response body. The HTTP response headers' content will change with every capture, as the datetime returned from a server is temporally dependent. Compare this to the response body, which very often contains the same content on each access, more often for static resources. Were the HTTP header and response body combined then added to IPFS, every IPFS hash would be unique, nullifying the potential for deduplication of identical content. Further, *ipwb* only retains response records. The rationale for this design decision is that the state of the art of web archive replay systems do not consider the WARC request record upon replay. While including request records may be useful in the future (for instance, to take into account the user-agent originally used to view the live website), WARC content is currently fully replayable without preserving the request records.

Our prototype works in two phases, illustrated in Figure 2 with red and blue annotations:

- *Indexing* – extracts records from the WARC Store one record at a time, splits each record into HTTP header and payload, stores the two pieces into IPFS (compressing before storing, if necessary), and generates a CDXJ record using the returned references and some other metadata from the WARC record.
- *Replay* – receives request from users containing a lookup URI and optionally a datetime, queries for matching record in the CDXJ, fetches the corresponding header and payload from the IPFS Store (using references returned from the index record), combines them, and performs necessary transformation to build the response to the user.

Future Work

Because of the novelty of IPFS, particularly relative to web archiving, there are numerous ways to expand this work. Collection builders can share their collections by just exchanging the index while keeping the data in the IPFS network and others can optionally replicate the data in their storage for redundancy. Further considerations of access control can also be addressed to encrypt and restrict content based on privacy and security mechanisms. Another model of IPFS-based archiving system can be built entirely using IPFS and IPNS technologies without the need of external indexes.