An Evaluation of Replay Quality for Web-Archived Pages

Daniel Bicho, Fernando Melo and Daniel Gomes
Arquivo.pt - The Portuguese Web Archive
{daniel.bicho, fernando.melo, daniel.gomes}@fccn.pt

February 1, 2017

Abstract

Arquivo.pt is a research infrastructure that enables search and access to web pages preserved since 1996 [3]. Therefore, the replay quality of archived pages is crucial to provide an experience close as possible how it was provided by the original pages. A recurrent evaluation on the replay quality is needed to continually monitor and maintain the replay quality of Web Archives. In a previous study [10], the Arquivo.pt has made an experiment to evaluate the replay quality and performance of Wayback software. Inconsistency problems were detected with these first evaluation method using WebPageTest, so an alternative method was developed. The proposed technique evaluates the Quality Replay using the software QAReplayProxy [5], an in-house tool built to evaluate the replay system of Arquivo.pt. A new study was performed using this method and the obtained results showed that PyWB with CDX indexes was the Wayback Machine configuration that presented the best replay quality. Also, the Merged Flat CDX Indexes obtained the best average response and throughput replaying archived web pages, with an index of 800 million documents.

1 Introduction

Arquivo.pt is a research infrastructure that enables search and access to web pages preserved since 1996 [3]. As other Web Archives, Arquivo.pt must reproduce its web-archived pages. Therefore, this replay quality of archived pages is crucial to provide an experience close as possible to the one provided by the original pages.

In a previous study [10], the Arquivo.pt has made an experiment to evaluate the replay quality and performance of Wayback software. The study revealed that Arquivo.pt Wayback Machine was outdated and other Wayback Machine alternatives like OpenWayback or PyWB would significantly improve the replay quality of Arquivo.pt archived websites. Based on that study, Arquivo.pt migrated from Wayback 1.2.1 to PyWB, a python Wayback Machine implementation developed by Illya Kreymer [9]. While integrating this new Wayback, and developing Arquivo.pt platform, a recurrent evaluation on the Wayback Machine quality was needed to continually monitor replay

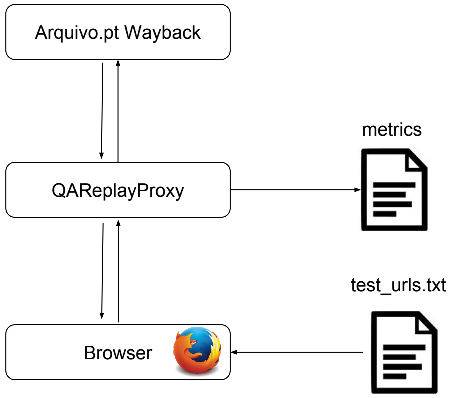

Figure 1: Overview of replay quality experimental setup.

quality provided by the service. The evaluation is also needed to detect and avoid possible development bugs that could be introduced accidentally.

Inconsistency problems were detected with the first evaluation method using WebPageTest [6]. While repeating the results were not reproducible. This inconsistency aligned with the fact that it is hard to know how the WebPageTest service is performing the tests, lead us to study other tools to measure replay quality. Therefore, in this study, a different technique based on a in-house tool was applied to evaluate the replay quality of Wayback Machines.

While evaluating the Wayback Machine replay quality of Arquivo.pt, a significant disparity between the replay quality with the CDX indexes [4] and the Lucene indexes [2] was detected, with the CDX indexes having a better score. This lead us to investigate how well the CDX indexes could perform in terms of speed and response scalability.

This report presents the obtained results as well as the problems identified while developing and executing the evaluation methodologies.

2 QAReplayProxy: a tool to measure Wayback Machines replay quality

In order to evaluate the replay quality an alternative method was developed. The proposed technique evaluates the Quality Replay using the software an QAReplayProxy [5], an in-house tool built to evaluate the replay system of Arquivo.pt.

Figure 1 presents a schematic overview about how metrics are gathered using QAReplayProxy.

This software acts as an invisible Proxy between the browser and the Waybak Machine. It inspects the requests and responses made between the browser and the Wayback Machine while an archived page is being replayed, and collects metrics about its

Wayback	Version	Indexes Type	Application Server
Arquivo Wayback	1.2.1	Lucene	Tomcat 8.0.30
PyWb CDX	0.10.7	CDX	uWSGI 2.0.11.1
PyWb Lucene	0.10.7	Lucene	uWSGI 2.0.11.1
OpenWayback	2.20	CDX	Tomcat 8.0.30

Table 1: Specifications for each tested Wayback Machine implementation.

replay by the Web Archive. Several metrics are gathered by the software, leaks to the live web, and URL requests made by the browser while replaying an archived website.

The URL requests are logged for further inspection, allowing to determine which resources were replayed or not. The QAReplayProxy also counts the number of resources that were loaded for each host. A request is a live-web leak if the origin host of a web file loaded is not the Wayback Machine host, meaning the it was fetched from the live web. This counter is used to report the number of live-web leaks and also their main origin.

3 Test Collection

A set of 400 websites was used to test the replay quality of several Wayback Machines with different configurations. This set of 400 websites were randomly select from .EU Collection crawled by Arquivo.pt [8]. This is the same set of URLs used in our previous study [10].

4 Evaluating Replay Quality

The replay quality was tested against several Wayback Machines implementations. Table 1 presents the Wayback Machines and indexes that were evaluated and shows the different Wayback Machine software, the type of indexes used and the application servers that hosted each Wayback Machine software.

4.1 Methodology

We ran the QAReplayProxy tool that gathers metrics from each system to evaluate the replay quality of several Wayback implementations over the test collection of 400 URLs of archived sites.

The metrics used to compare each Wayback Machine system were the HTTP response codes and the number of live-web leaks. The return codes provide us additional insights about replay quality. For instance, reaching a higher number of 200s return codes meant that more content was retrieved by the Wayback Machine system, that is, a better replay quality. Other return codes like 500s evidence problems on the replay system.

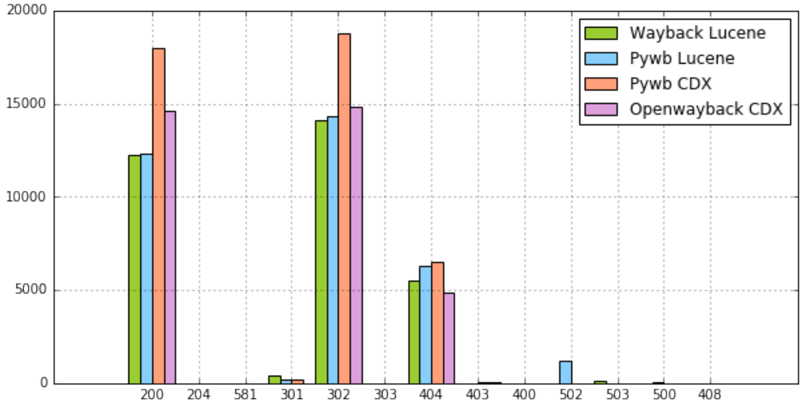

Figure 2: Waybacks setup return codes comparison.

4.2 Results Analysis

Figure 2 presents the distribution of the HTTP response codes obtained for each Wayback Machine system. The obtained results show that the system with best results regarding the amount of content rendered by the browser (HTTP response codes 200) is PyWB using CDXs indexes, with 18 009 fetched resources, followed by OpenWayback using CDX indexes with 14 629 fetched resources.

Both Waybacks using the Lucene Indexes implementation, namely Wayback Lucene (12 262) and PyWB Lucene (12 346) had a similar replay quality regarding the content rendered. Analysing other result codes responses, the PyWB Lucene implementation is the only one that presented 502 return code errors. These return codes appear because the PyWb Wayback Machine can detect redirect loops at the server side. When using the Lucene indexes a known issue caused that it didn't differentiate an URL like `www.jn.pt` from `jn.pt`. This problem had a big impact on the replay quality, and is one of the problems that contribute for the worse number of resources replayed using Lucene indexes in comparison with CDX indexes.

An other metric applied to compare Wayback implementations was the number of live-web leaks, the number of resources that are fetched from the live web instead of the Web Archive. That leads to temporal incoherences while rendering a Web page. So they must be minimized to improve replay quality. Figure 3 presents the obtained results, and it shows that the number of live leaks is very low on Pywb and OpenWayback Machines implementations, representing a huge improvement in comparison with the Arquivo.pt's old Wayback Machine (Wayback Lucene).

4.3 Problems Identified While Measuring Replay Quality

Some inconsistencies on the obtained results were identified while performing the described methodology to measure the replay quality of the different Wayback Machines and index implementations. The Wayback Machines using Lucene indexes were incho-

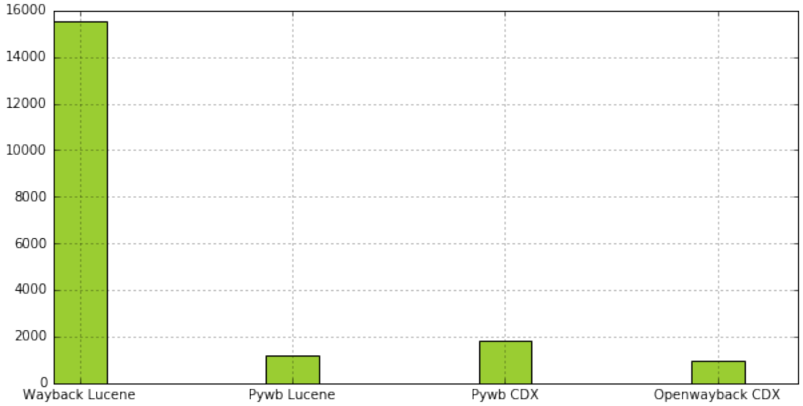

Figure 3: Waybacks setup live-web leaks comparison.

erent along several repeated tests. A high number of 500 HTTP response codes were obtained. But it decreased when the tests were repeated. The first clue was that this behaviour is caused by to many concurrent requests at the Query Servers hosting the Lucene indexes. The results improve when the cache was warmed with the resources that needed to be replayed, being able to serve more resources on the posterior tests. Despite the previous warming of the Query Servers, the number of 500s returned by the Waybacks using the Lucene indexes was still higher than when using CDX implementations, that did not suffer from the same issue.

5 Evaluation of Replay Speed & Throughput

The CDX index implementation reveals good results on the quality of the replay. Also, it does not have the problems of the 502 error return codes that occur with the Lucene indexes and have much more stable results, don't suffering from the inconsistent 500/404 errors that Lucene indexes suffers. This lead us to evaluate the replay performance of the CDX indexes, measuring its response time and throughput and comparing it with the Lucene indexes. The tested configurations were the following:

- Lucene Index: loaded with 600 million entries for web pages;
- Split Flat CDX Index: one index for each collection, loading a total of 800 million entries for web pages;
- Merged Flat CDX Index: one index with all collections, loading a total of 800 million entries for web pages;
- Split ZipNum Cluster Index [7]: 10 shards ZipNum index for each collection, loading a total of 800 million entries for web pages;
- Merged ZipNum Cluster Index: 10 shards ZipNum index with all collections, loading a total of 800 million entries for web pages;

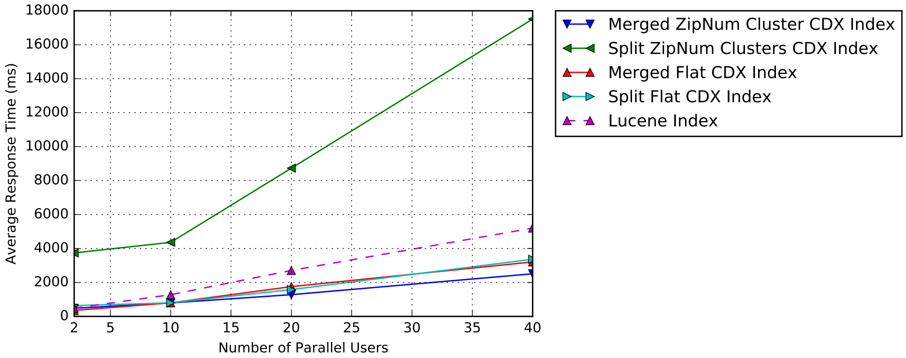

Figure 4: Response times measurements for each index configuration.

Ideally both Lucene and CDX indexes should had the same size regarding the number of documents indexed. But due to technical problems was not possibly to use exactly the same collections for both the indexes, resulting on CDX indexes with more documents. Despite this difference it should be enough to take conclusions about the performance of CDX indexes.

5.1 Methodology

Increasing levels of workload were applied over the several Wayback Machines systems using JMeter [1] hosted on 2 machines.

The load is expressed in terms of number of virtual users (simulated by JMeter).

For each index system, the replay performance was measured by running 5 minute tests with 2, 10, 20 and 40 parallel users, simulating replay requests for the archived web pages.

5.2 Results Analysis

Figure 4 displays the average Response Time measured for each configuration. The Merged Flat CDX Index presents the best response time with an average response time of 3 226 ms with a workload of 40 parallel users. The Merged ZipNum Index and the Split Flat CDX Indexes presented an overall similar response time. The configuration with the worst performance was the Split ZipNum cluster followed by the Lucene index.

The throughput capacity (web pages replay per second) was also measured for each Wayback Machine configuration (Figure 5). The Merged Flat CDX Index configuration has the best throughput, with an average of 11 web pages replays per second with a workload of 40 parallel users.

These results indicate that the Flat CDX Index configuration have a reasonable performance at least while loaded with about 800 Million documents, an equivalent of 409 GB index size. The ZipNum Indexes configuration presented the worst results,

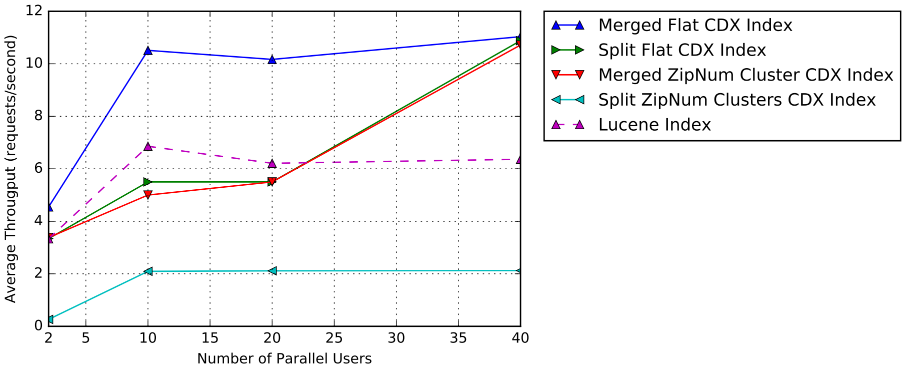

Figure 5: Throughput measurement for each index configuration.

probably because the size of the index is not big enough to compensate the overhead added by the ZipNum. It would be interesting to perform experiments with larger indexes in the future.

The compression and fragmentation of the ZipNum CDX index in several smaller parts adds more CPU load to the server. Also, the size of the index is low, 409 GB of Flat CDX Index and 59 GB of compressed CDX files on a server with 256 GB RAM. This index files are quickly added to the Linux page-cache, reducing the number of disk accesses. This behaviour is a limitation of the test collection, since it only uses 400 URL from the same collection, which will always trigger the usage of the same disk blocks, and consequently the same page-cache memory blocks.

The ZipNum Indexes usage will only pay-off with bigger collections, when the size of the CDX files is big enough to cause bottleneck on the Disk I/O.

With the increasing size of the CDX index, an assessment is required to determine which is the most adequate index configuration, since it will be dependent of the size of index and the performance capacity of the storage system. These factors will be crucial to determine if the CPU overhead added by the compression compensates the decrease of disk reads and the optimal size for ZipNum index shards, because for each additional shard there will be an additional read overhead for each request.

6 Conclusions

These new methodology adopted in this study to evaluate the replay quality of Wayback Machines presented more consistent results. It enabled more control on how the tests were performed instead of the uncontrolled and inconsistent results obtained through WebPageTest. The obtained results showed that PyWB with CDX indexes was the Wayback Machine configuration that presented the best replay quality. This result is mainly explained by the fact that the Lucene indexes used by Arquivo.pt currently have some issues that have a negative impact on the quality of the replay. The main problems identified are the occurrence of several redirect loop errors.

Other main problem was that the responses returned by our Lucene based indexes were very inconsistent. Often, the same archived web resource is available or originate a 500/404 error response. This error is caused by concurrent requests, the specific origin of the problem bottleneck location still needs to be investigated.

The PyWb can use several types of indexes. Since the CDX indexes performed so well in terms of replay quality, capacity tests were executed to measure their response time and throughput. The tests executed were made with several CDX indexes configurations, like ZipNum CDX indexes and flat CDX indexes. The results showed that the Merged Flat CDX Indexes had the best performance with an index of 800 million documents, but the tests performed have limitations in terms of size and scalability. Therefore, it would be interesting to test with larger indexes.

References

[1] Apache JMeter. http://jmeter.apache.org/.

[2] Apache Lucene Search Index. http://lucene.apache.org/.

[3] Arquivo.pt: pesquisa sobre o passado. http://arquivo.pt/.

[4] Internet Archive: CDX File Format Reference. https://archive.org/web/researcher/cdx{_}file{_}format.php.

[5] QAReplayProxy Tool Software. https://github.com/arquivo/QAReplayProxy.

[6] WebPagetest - Website Performance and Optimization Test. https://www.webpagetest.org/.

[7] Zipnum and cdx cluster merging — Further Explorations Into The Black Hole. http://aaron.blog.archive.org/2013/05/28/zipnum-and-cdx-cluster-merging/.

[8] D. Bicho and D. Gomes. A first attempt to archive the .EU domain Technical report. http://arquivo.pt/crawlreport/Crawling_Domain_EU.pdf, 2015.

[9] Illya Kreymer. pywb. https://github.com/ikreymer/pywb.

[10] F. Melo, D. Bicho, and D. Gomes. A Comparison Between The Performance of Wayback Machines. http://sobre.arquivo.pt/sobre/publicacoes-1/a-comparison-between-the-performance-of-wayback, 2016.