Cost of a WARC
Analyzing Web Archives in the Cloud

Ryan Deschamps, Samantha Fritz, Jimmy Lin, Ian Milligan, and Nick Ruest
Why do we care about web archives?

Born-digital sources have the potential to reshape research in the humanities and social sciences;

Research access has lagged (beyond Wayback Machine, analysis ecosystem is mostly command-line-based tools)

As we plan for research access, we need to understand the economics associated with providing this sort of access
How much does it cost to analyze a WARC (the standard container file format of web archives) in the cloud?
US$7 per TB
What do we mean by the “Cloud”?

We conduct our work on the Compute Canada Cloud, which is an OpenStack instance supported by a research grant.

As OpenStack is a popular open-source cloud platform, our findings should be generalizable.

We translated all of our compute time into Amazon Web Services costs as it is the most popular commercial provider.
What are we performing “analysis” with?

Analysis using the **Archives Unleashed Toolkit** or **AUT**

**AUT** is a Scala domain-specific language on top of the Apache Spark platform.
What do we mean by “Analysis”?

The Filter - Analyze - Aggregate - Visualize (FAAV) Cycle

Common analytics task: crawl statistics to visualizing web graphs to exploring text at scale

Informed by extensive hands-on collaboration
ANALYSIS AT SCALE
What do we mean by “Analysis”?  

**Extract all URLs** to compute the frequency of domains appearing in a given collection (domain distribution);  

**Extract all plain text** from all pages, along with metadata such as crawl date, domain name, and URL (full text); and  

**Extract all hyperlinks** to create a domain-to-domain network graph (webgraph);
On to the experiment!
We decided to use a 16 core, 64GB memory virtual machine.

Powerful, but struck the balance between expensive and power.

Why not a cluster?
Analysis based on analyzing the cost of processing **48 Archive-It collections** from six Canadian universities (Toronto, Victoria, Simon Fraser University, Manitoba, Dalhousie, and Winnipeg).

A variety of **sizes** – smallest at 1.2GB was Victoria’s academic calendar; largest at 4.3TB was Canadian Government Information Collection.

<table>
<thead>
<tr>
<th>Size</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>≥ 1 GB, &lt; 10 GB</td>
<td>10</td>
</tr>
<tr>
<td>≥ 10 GB, &lt; 100 GB</td>
<td>18</td>
</tr>
<tr>
<td>≥ 100 GB, &lt; 1 TB</td>
<td>15</td>
</tr>
<tr>
<td>≥ 1 TB</td>
<td>5</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>48</strong></td>
</tr>
</tbody>
</table>
The Experiment (Workflow)
Findings

We then took all the times for each job (Domain, Full Text, Webgraph) and found processing time per GB in seconds.

Webgraph is most computationally intensive, but not too much so.

Processing times drop as size increases, as startup costs are amortized.

<table>
<thead>
<tr>
<th>Derivative</th>
<th>all</th>
<th>L</th>
<th>M</th>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>domain distribution</td>
<td>32</td>
<td>25</td>
<td>27</td>
<td>36</td>
</tr>
<tr>
<td>full text</td>
<td>34</td>
<td>28</td>
<td>35</td>
<td>34</td>
</tr>
<tr>
<td>webgraph</td>
<td>36</td>
<td>34</td>
<td>36</td>
<td>36</td>
</tr>
<tr>
<td>total</td>
<td>102</td>
<td>87</td>
<td>98</td>
<td>106</td>
</tr>
</tbody>
</table>

Figure: Processing times per GB in seconds
Scatter plot between collection size and total processing time, illustrating a linear relationship.
Findings

Derivative files are much smaller

Researcher can usually work with these derivative files on their own systems in a way they could not work with their WARC

<table>
<thead>
<tr>
<th>Derivative</th>
<th>all</th>
<th>L</th>
<th>M</th>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>domain distribution (KB)</td>
<td>0.95</td>
<td>0.51</td>
<td>0.98</td>
<td>1.01</td>
</tr>
<tr>
<td>full text (MB)</td>
<td>78.5</td>
<td>97.6</td>
<td>102.1</td>
<td>62.4</td>
</tr>
<tr>
<td>webgraph (KB)</td>
<td>76.9</td>
<td>85.8</td>
<td>122.6</td>
<td>50.9</td>
</tr>
</tbody>
</table>

**Figure**: Derivative sizes per GB
So we know the times to compute these derivatives. Show me the money!
<table>
<thead>
<tr>
<th>Derivative</th>
<th>all</th>
<th>L</th>
<th>M</th>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>domain distribution</td>
<td>$6.51</td>
<td>$4.67</td>
<td>$5.05</td>
<td>$7.63</td>
</tr>
<tr>
<td>full text</td>
<td>$6.73</td>
<td>$5.24</td>
<td>$6.65</td>
<td>$7.04</td>
</tr>
<tr>
<td>webgraph</td>
<td>$7.19</td>
<td>$6.46</td>
<td>$6.82</td>
<td>$7.52</td>
</tr>
<tr>
<td>total</td>
<td>$20.43</td>
<td>$16.37</td>
<td>$18.52</td>
<td>$22.19</td>
</tr>
</tbody>
</table>

Processing cost per TB in US $
Cost of a WARC

C5.4xlarge (16 core, 68 GB memory) is $0.68/hour in US East (Ohio)

The previous results show a **macro-average**

The bottom line: US$7/TB for a typical analytics operation such as generating **domain frequency** reports, extracting **full text of a collection**, or extracting the **link-to-link webgraph** of hyperlinks.
Cost of a WARC

This is *cost-competitive*

Google BigQuery costs US$5 per TB — *but* is SQL based and prices on uncompressed size whereas our calculations were on compressed WARC files (which are roughly 60% the size of uncompressed WARC files).

Archives Unleashed is price competitive with commercial services, albeit without any profit margin.
Proposed Workflow

Cheaper download server (ex. t3.medium)

Expensive processing server (ex. c5.4xlarge)
Limitations: Storage

We did not include storage in this discussion. 1TB of data costs US$23 per month. Our preferred workflow would be to transfer WARC, analyze, and then delete them quickly.

At 30 MB/s data transfer speed, transferring a TB costs US$0.40; less than the per-day cost of S3 data storage.

As long as the preservation copy is secure, the “processing copy” can be created and deleted on a whim.
Conclusions
Conclusions
We share the beginnings of an economic analysis and believe the costs to be quite affordable; whether institutions or individual scholars find these costs palatable remains to be seen.
US$7 per TB
Thanks to our supporters!