Experiments using a Distributed Web Crawler to Process and Index Web Archives

Sebastian Nagel
sebastian@commoncrawl.org

2nd International Symposium on Open Search Technology, OSSYM 2020,
hosted by CERN, Geneva, Switzerland, Oct 12–14, 2020

Index Web Archives using a Web Crawler

- web archives collect parts of the World Wide Web as cultural heritage and for research
- a web crawler "browses" the WWW to
  - feed a search index (web search engine)
  - mine information from the data
  - archive the data in web archives

Objective: make a web crawler read web pages from web archives

- utilize a single architecture to process and index web pages, no matter whether pages are from the web or from archives
- reproducible experiments independent from time and location

About Common Crawl

- we're a non-profit that makes web data accessible to programmers and data scientists
- for natural language processing, web science, semantic web, internet security research, …
- hosted as Open Data set on Amazon Web Services
- web archives 2008 – 2020:
  200 billion web pages captured (HTML only)
- plus secondary formats (text, metadata, URL index)
- 5 Petabytes of data in total (summer 2020)
- per month: 4–6 Petabytes of data requested ("downloaded"),
  2–6 billion requests (files or chunks)

The WARC format (Web ARChive)

- "freezes" the internet traffic between a client (a web crawler or browser) and web servers at the HTTP protocol level
  - content payload
  - HTTP headers
  - connection metadata (datetime, IP address)
- ISO standard since 2009 [1,2]
- WARC I/O modules for many programming languages [3,4]
- text header + payload
- per-record gzipped: extract single records if offsets are known

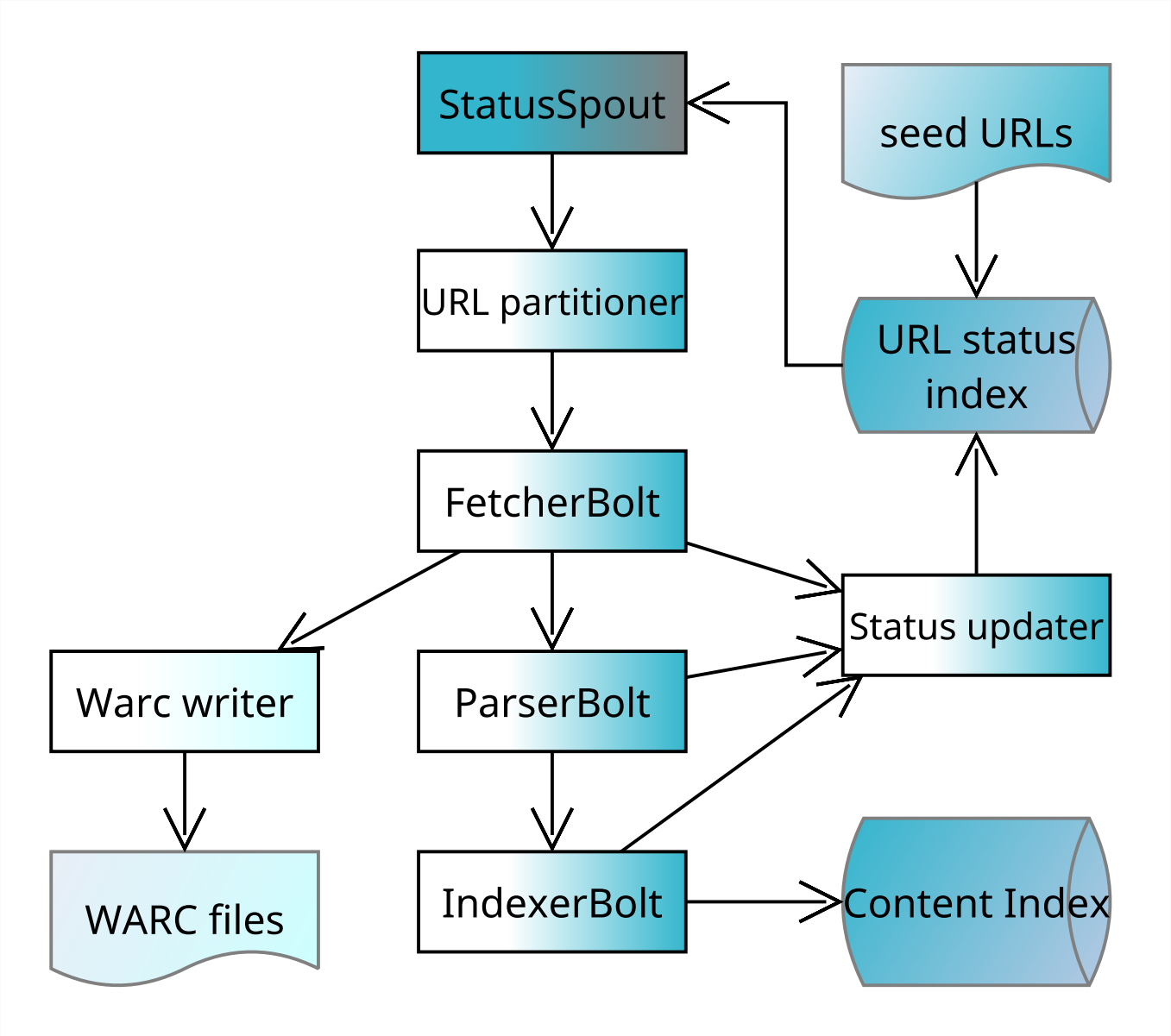
The WARC format (Web ARChive)

```
% curl -s -r441780722-$((441780722+10599-1)) \
   "https://commoncrawl.s3.amazonaws.com/crawl-data/CC-MAIN-2020-34/segments/1596439738819.78/warc/CC-MAIN-
20200811180239-20200811210239-00091.warc.gz" \
   | gzip -dc
WARC/1.0
WARC-Type: response
WARC-Date: 2020-08-11T19:24:35Z
WARC-Target-URI: https://opensearchfoundation.org/en/2nd-international-symposium-on-open-search/
...
WARC-Identified-Payload-Type: text/html

HTTP/1.1 200 OK
Date: Tue, 11 Aug 2020 19:24:34 GMT
Server: Apache
...
Content-Type: text/html; charset=UTF-8
Content-Length: 47869

<!DOCTYPE html>
...
<title>2nd International Symposium on Open Search, 12-14 October 2020 &#8211; Open Search Foundation</title>
...
```

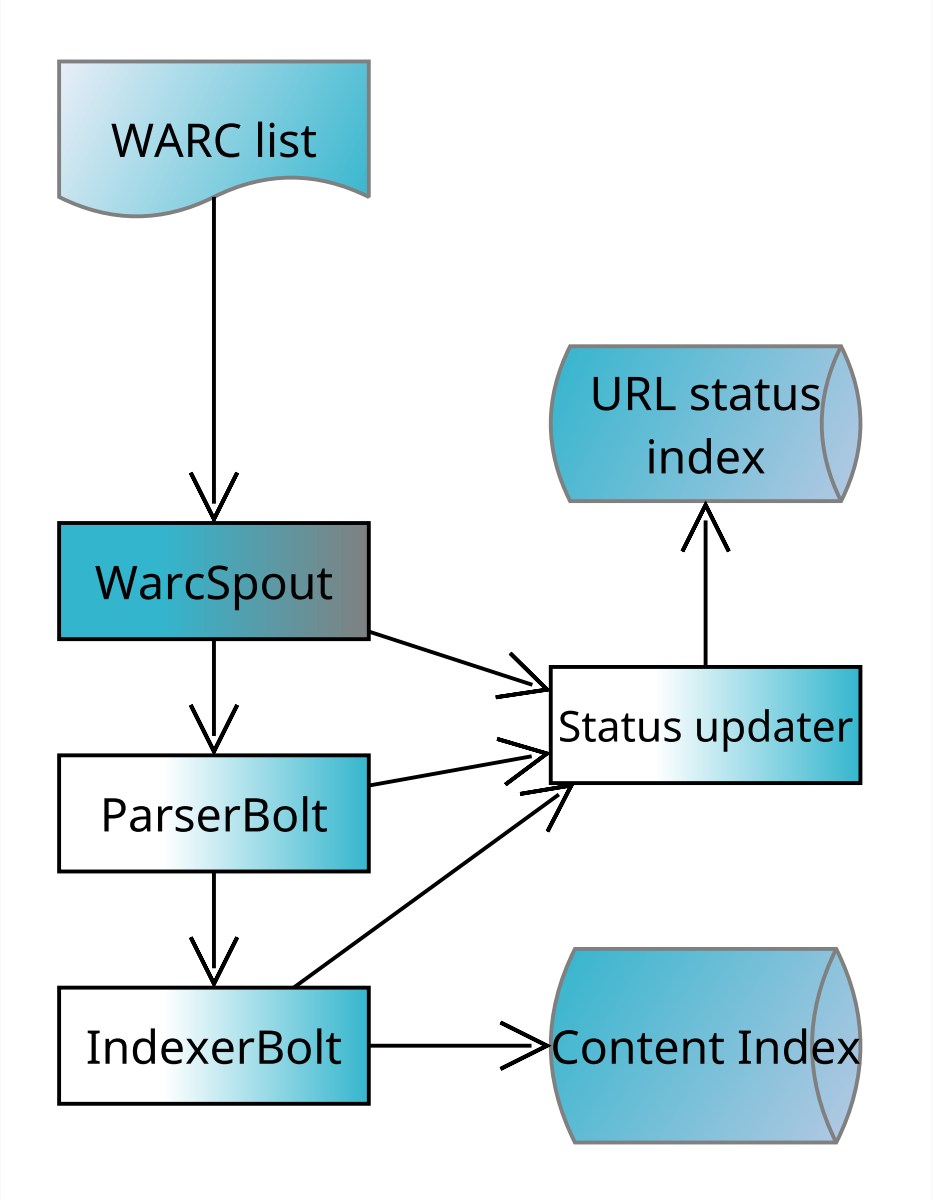
StormCrawler

- a software library, API and program [5,6]
- to build low latency, distributed (scalable) web crawlers
- written in Java
- based on Apache Storm [7], a distributed stream processing framework
- open source (Apache license)
- highly modular and flexible
  - parsers for HTML, PDF, Office documents, RSS, sitemaps, …
  - indexers for Elastics, Solr, CloudSearch

Topology Web Crawler



Topology Web Archive Crawler



Experiments

- baseline: process WARC files, emit captured web pages into topology, do not use content
- fidelity: read WARC files and write captures again into WARCs
- index: parse HTML pages from WARC files, extract text and metadata and index documents into Elasticsearch

- source code and instructions on github [8]
- run your own experiments!

  If you like it, consider using StormCrawler also for web crawling!

References

1. https://en.wikipedia.org/wiki/Web_ARChive
2. https://iipc.github.io/warc-specifications/specifications/warc-format/warc-1.1/
3. https://www.archiveteam.org/index.php?title=The_WARC_Ecosystem
4. https://github.com/iipc/awesome-web-archiving
5. https://stormcrawler.net/
6. https://github.com/DigitalPebble/storm-crawler/
7. https://storm.apache.org/
8. https://github.com/sebastian-nagel/warc-crawler