What is a Web archive

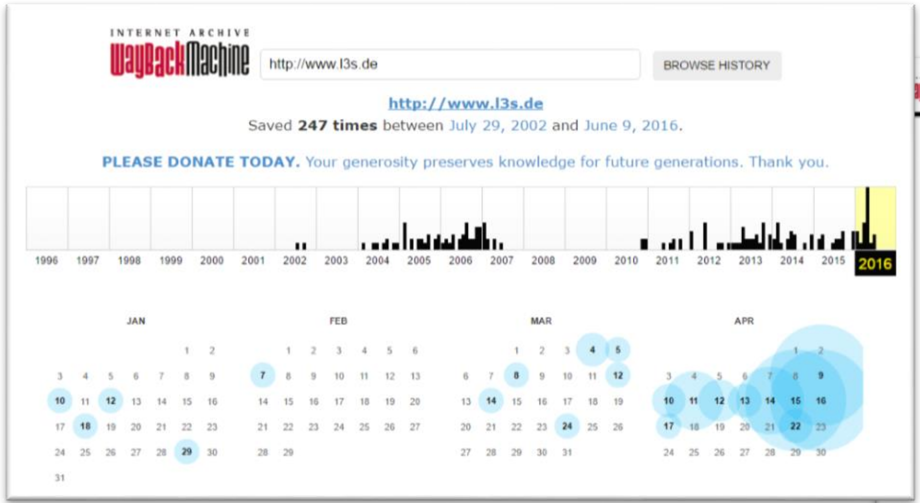
- Attempt to preserve the history of the Web
- Consisting of any kind of Web resources
  - e.g., HTML, images, video, scripts, …
- … stored in big files in the dedicated *WARC* format

INTERNET ARCHIVE

INTERNET ARCHIVE WayBackMachine

IIPC netpreserve.org INTERNATIONAL INTERNET PRESERVATION CONSORTIUM

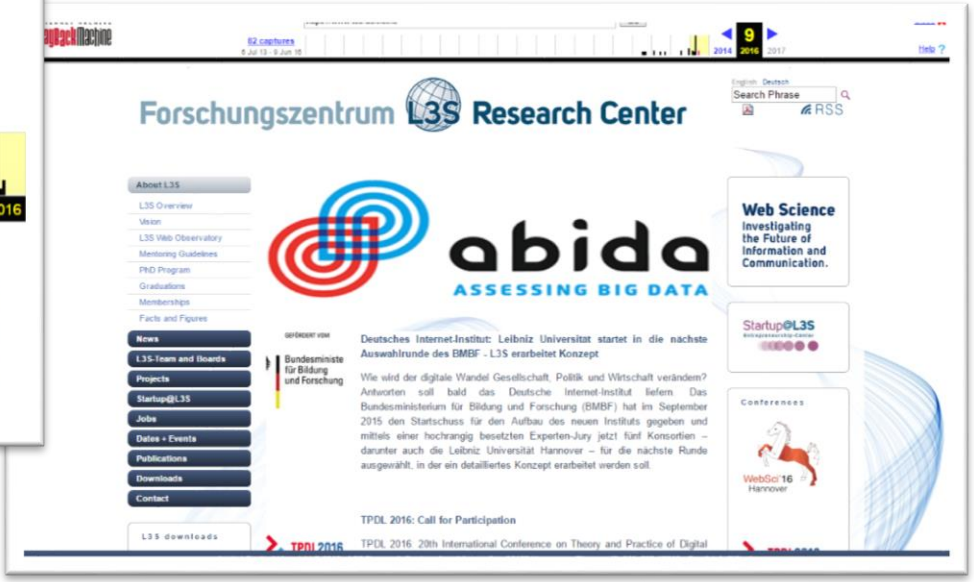
UK WEB ARCHIVE preserving uk websites



http://www.l3s.de

Saved **247 times** between July 29, 2002 and June 9, 2016.

PLEASE DONATE TODAY. Your generosity preserves knowledge for future generations. Thank you.



Deutsches Internet-Institut: Leibniz Universität startet in die nächste Auswahlrunde des BMBF - L3S erarbeitet Konzept

Wie wird der digitale Wandel Gesellschaft, Politik und Wirtschaft verändern? Antworten soll bald das Deutsche Internet-Institut liefern. Das Bundesministerium für Bildung und Forschung (BMBF) hat im September 2015 den Startschuss für den Aufbau des neuen Instituts gegeben und mittels einer hochrangig besetzten Experten-Jury jetzt fünf Konsortien – darunter auch die Leibniz Universität Hannover – für die nächste Runde ausgewählt, in der ein detailliertes Konzept erarbeitet werden soll.

TPDL 2016: Call for Participation

TPDL 2016: 20th International Conference on Theory and Practice of Digital

Web Archives are Big Data

- Processing requires computing clusters
  - i.e., Hadoop, YARN, Spark, …

Source: *Yahoo!*

- Distributed computing
- MapReduce or variants
  - Transform, aggregate, write
  - Homogeneous data
    - Well-defined format, clean
    - Easy to split and handle

- Web archive data is heterogeneous, may include text, video, images, …
  - Requires cleaning, filtering, selection, extraction and finally, processing

Web Archives as Scholarly Source

- Web archive data is heterogeneous, may include text, video, images, …
  - Requires cleaning, filtering, selection, extraction and finally, processing
- Data is temporal, time differences within a website, among websites
  - A resource may have been crawled and be included multiple times
    - With no or only slight changes, layout changes, content changes, …
- Users are often from various disciplines, not only computer scientists
  - Social scientists, political scientists, historians, …
  - Want to express their needs, define sub-sets, create research corpora
    - Require a format they know, which is readable and reusable
    - Technical persons can support, but need the tools

ArchiveSpark: Efficient Web Archive Access, Extraction and Derivation

Helge Holzmann, Vinay Goel, Avishek Anand

Alexandria Workshop 2016

https://github.com/helgeho/ArchiveSpark

Overview

- Goal: Easy research corpus building from Web archives
- Framework based on Apache Spark [1]
  - Supports efficient filter / selection operation
  - Supports derivations by applying third-party libraries
  - Output in a pretty and machine readable format
- Identified six objectives based on practical requirements
- Benchmarked against pure Spark and HBase
  - Both using Warcbase helpers [2]

[1] http://spark.apache.org
[2] https://github.com/lintool/warcbase

Example Use Case

- Political scientist wants to analyze sentiments and reactions on the Web from a previous election cycle.
- Five decisions to be taken to narrow down the dataset:
  1. Filter temporally to focus on the election period
  2. Select text documents by filtering on MIME type
  3. Only keep online captures with HTTP status code 200
  4. Choose a captured version, for instance the latest of each page
  5. Look for political signal terms in the content to get rid of unrelated pages
- Finally, extract relevant terms / snippets to analyze sentiments
  - Document lineage, e.g., the title might have more value than the body text

Objectives

1. A simple and expressive interface
2. Compliance to and reuse of standard formats
3. An efficient selection and filtering process
4. An easily extensible architecture
5. Lineage support to comprehend and reconstruct derivations
6. Output in a standard, readable and reusable format

O1: Simple and expressive interface

- Based on Spark
- Expressing instructions in Scala
- Simple accessors provided
- Easy extraction / enrichment mechanisms

```
val rdd = ArchiveSpark.hdfs(cdxPath, warcPath)
val onlineHtml = rdd.filter(r => r.status == 200 && r.mime == "text/html")
val entities = onlineHtml.enrich(Entities)
entities.saveAsJson("entities.gz")
```

O2: Standard formats

- (W)ARC and CDX widely used among the big Web archives
  - (Web) ARChive files (**ISO** 28500:2009)
  - Crawl index
- No additional pre-processing / indexing required

INTERNET ARCHIVE

ARCHIVE-IT

IIPC netpreserve.org INTERNATIONAL INTERNET PRESERVATION CONSORTIUM

Common Crawl

O3: Efficient selection and filtering

- Initial filtering purely based on meta data (CDX)
  - without touching the archive
- Seamless integration of contents / extractions / enrichments

```
val Title = HtmlText.of(Html.first("title"))
val filtered = rdd.filter(r => r.status == 200 && r.mime == "text/html")
val corpus = filtered.enrich(Title)
  .filter(r => r.value(Title).get.contains("science"))
```

O4: Easily extensible architecture

- **Enrich functions** allow integrating any Java/Scala code/library
  - with a unified interface
- Base classes provided
  - Implementations need to define the following only:
    1. Default dependency enrich function (e.g., StringContent)
    2. Function body (Java/Scala code / library calls, e.g., Named Entity Extractor)
    3. Result fields (e.g., persons, organizations, locations)

```
rdd.mapEnrich(StringContent, "length") {str => str.length}
```

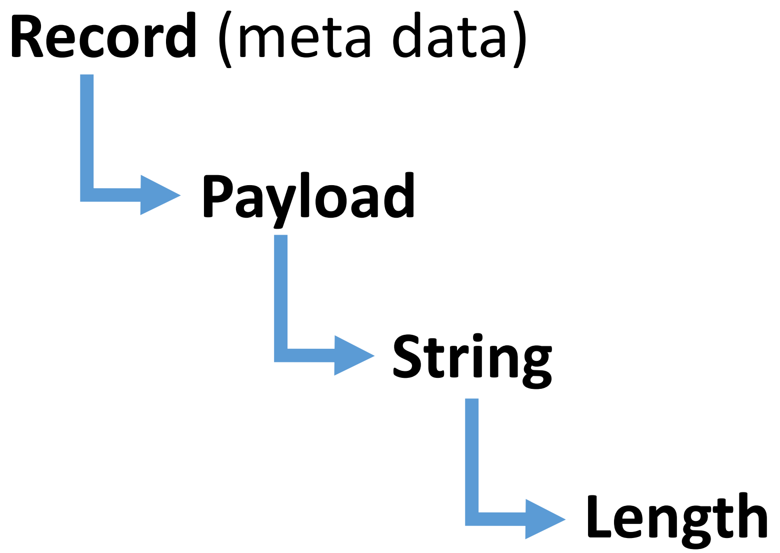
Dependency **Result field** **Body**

Helge Holzmann (holzmann@L3S.de)

O5: Lineage documentation

- Nested JSON objects represent enrichments / derivations

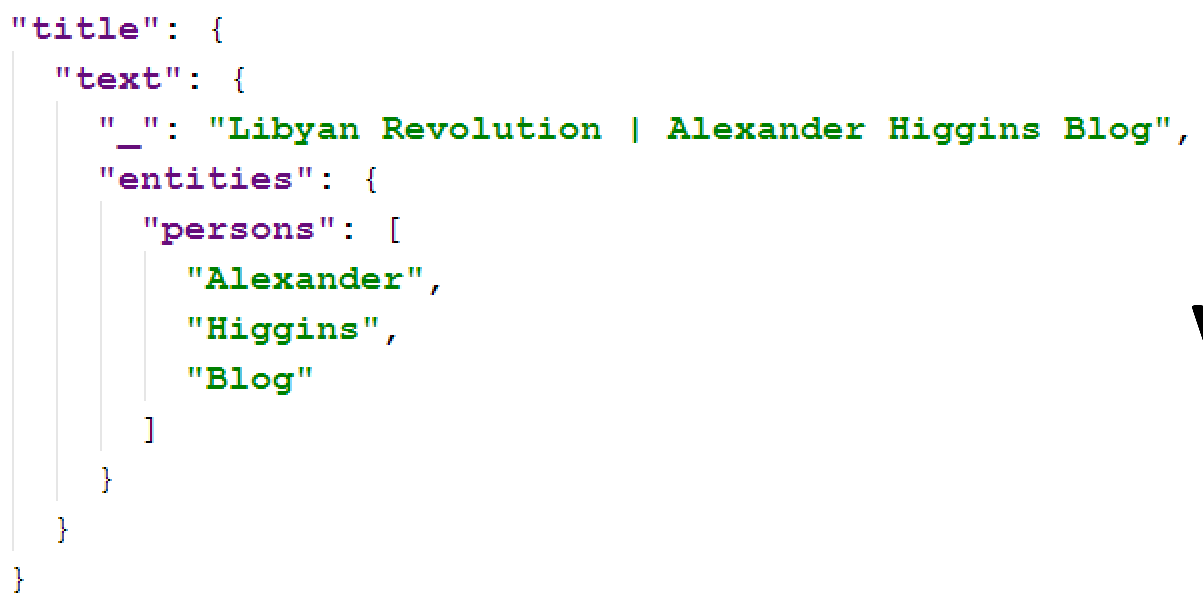
```
{
  "record": {
    "surtUrl": "com,example)/jcdl",
    "timestamp": "2016-01-17T11:32:53.000+01:00",
    "originalUrl": "http://example.com/jcdl",
    "mime": "text/html",
    "status": 200,
    "digest": "RKMS6XLYED4G8POFQUIN37WDEWYLD9Z",
    "redirectUrl": "-",
    "meta": "-"
  }
  "payload": {
    "string": {
      "_": "<html>...</html>",
      "length": 2345
    }
  }
}
```



Helge Holzmann (holzmann@L3S.de)

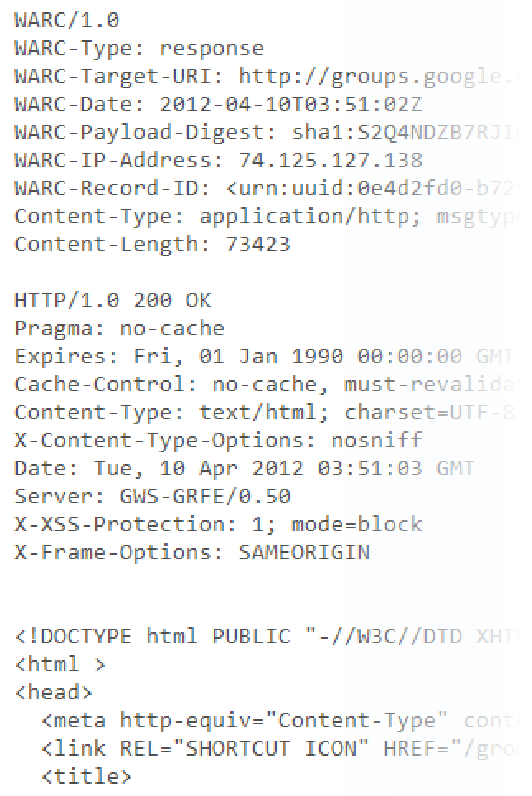
O6: Readable / reusable output

- Filtered, cleaned, enriched JSON vs. raw (W)ARC
  - widely used, structured, pretty printable, reusable



```
"title": {
  "text": {
    "_": "Libyan Revolution | Alexander Higgins Blog",
    "entities": {
      "persons": [
        "Alexander",
        "Higgins",
        "Blog"
      ]
    }
  }
}
```

vs.



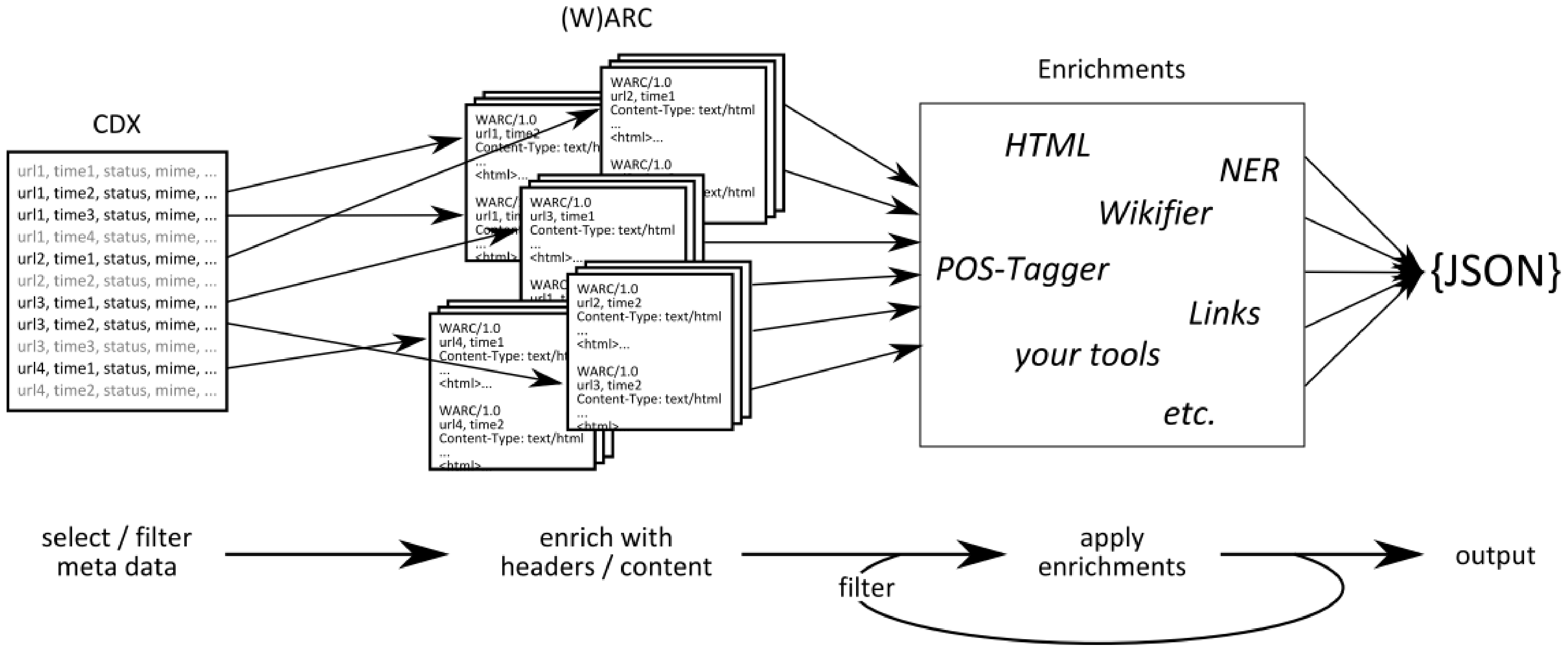
```
WARC/1.0
WARC-Type: response
WARC-Target-URI: http://groups.google.
WARC-Date: 2012-04-10T03:51:02Z
WARC-Payload-Digest: sha1:S2Q4NDZB7RJI
WARC-IP-Address: 74.125.127.138
WARC-Record-ID: <urn:uuid:0e4d2fd0-b72
Content-Type: application/http; msgtyp
Content-Length: 73423

HTTP/1.0 200 OK
Pragma: no-cache
Expires: Fri, 01 Jan 1990 00:00:00 GMT
Cache-Control: no-cache, must-revalida
Content-Type: text/html; charset=UTF-8
X-Content-Type-Options: nosniff
Date: Tue, 10 Apr 2012 03:51:03 GMT
Server: GWS-GRFE/0.50
X-XSS-Protection: 1; mode=block
X-Frame-Options: SAMEORIGIN


<!DOCTYPE html PUBLIC "-//W3C//DTD XHT
<html >
<head>
  <meta http-equiv="Content-Type" cont
  <link REL="SHORTCUT ICON" HREF="/gro
  <title>
```

The ArchiveSpark Approach

- Filter as much as possible on meta data before touching the archive
- Enriching instead of mapping / transforming data


(W)ARC
CDX
url1, time1, status, mime, ...
url1, time2, status, mime, ...
url1, time3, status, mime, ...
url1, time4, status, mime, ...
url2, time1, status, mime, ...
url2, time2, status, mime, ...
url3, time1, status, mime, ...
url3, time2, status, mime, ...
url3, time3, status, mime, ...
url4, time1, status, mime, ...
url4, time2, status, mime, ...
WARC/1.0 url2, time1 Content-Type: text/html ... <html>...
WARC/1.0 url1, time2 Content-Type: text/html ... <html>...
WARC/1.0 url3, time1 Content-Type: text/html ... <html>...
WARC/1.0 url2, time2 Content-Type: text/html ... <html>...
WARC/1.0 url3, time2 Content-Type: text/html ... <html>
WARC/1.0 url4, time1 Content-Type: text/html ... <html>...
WARC/1.0 url4, time2 Content-Type: text/html ... <html>
Enrichments
HTML
NER
Wikifier
POS-Tagger
Links
your tools
etc.
{JSON}
select / filter meta data
enrich with headers / content
filter
apply enrichments
output

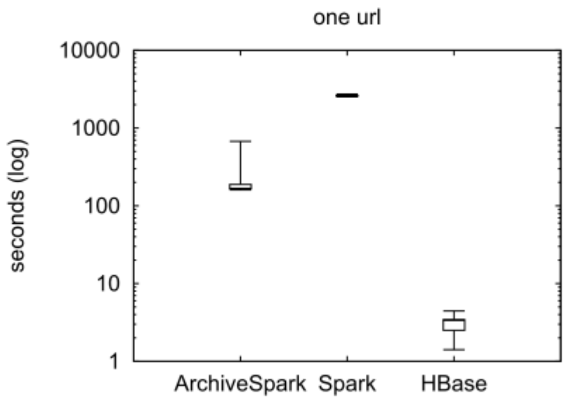
Experimental Setup

- Dataset: Occupy Movement 2011/2012 [1]
  - 10,089,668 unique URLs, 17,478,067 captures
  - 470.9 GB of compressed WARC files, 24.4 GB of CDX files
- Cluster: 25 nodes, 256 CPU cores, 2560 GB of RAM
  - Spark run with 10 executors, 4GB of memory each
- Compared ArchiveSpark with:
  - Pure Spark, using Warcbase's WARC input format for Spark [2]
  - HBase, using Warcbase to ingest the data [2] (~24 hours)

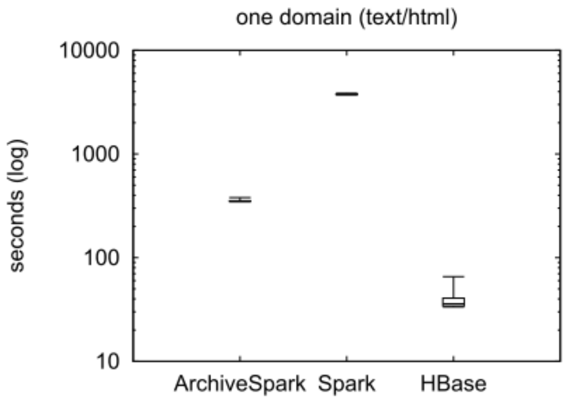
[1] http://archive-it.org/collections/2950
[2] https://github.com/lintool/warcbase

Benchmarks

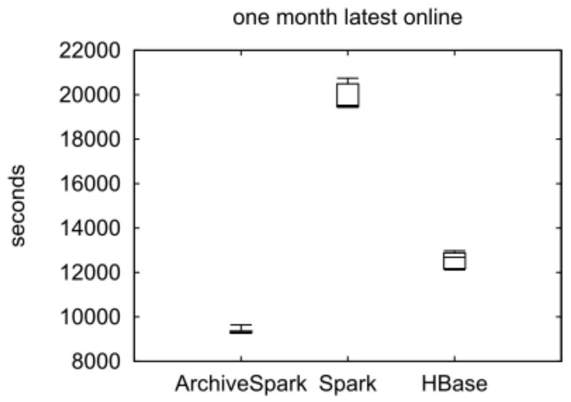
- Three scenarios, from basic to more sophisticated:
  a) Select one particular URL
  b) Select all pages (MIME type text/html) under a specific domain
  c) Select the latest successful capture (HTTP status 200) in a specific month
- Benchmarks do not include derivations
  - Those are applied on top of all three methods and involve third-party libraries



(a)



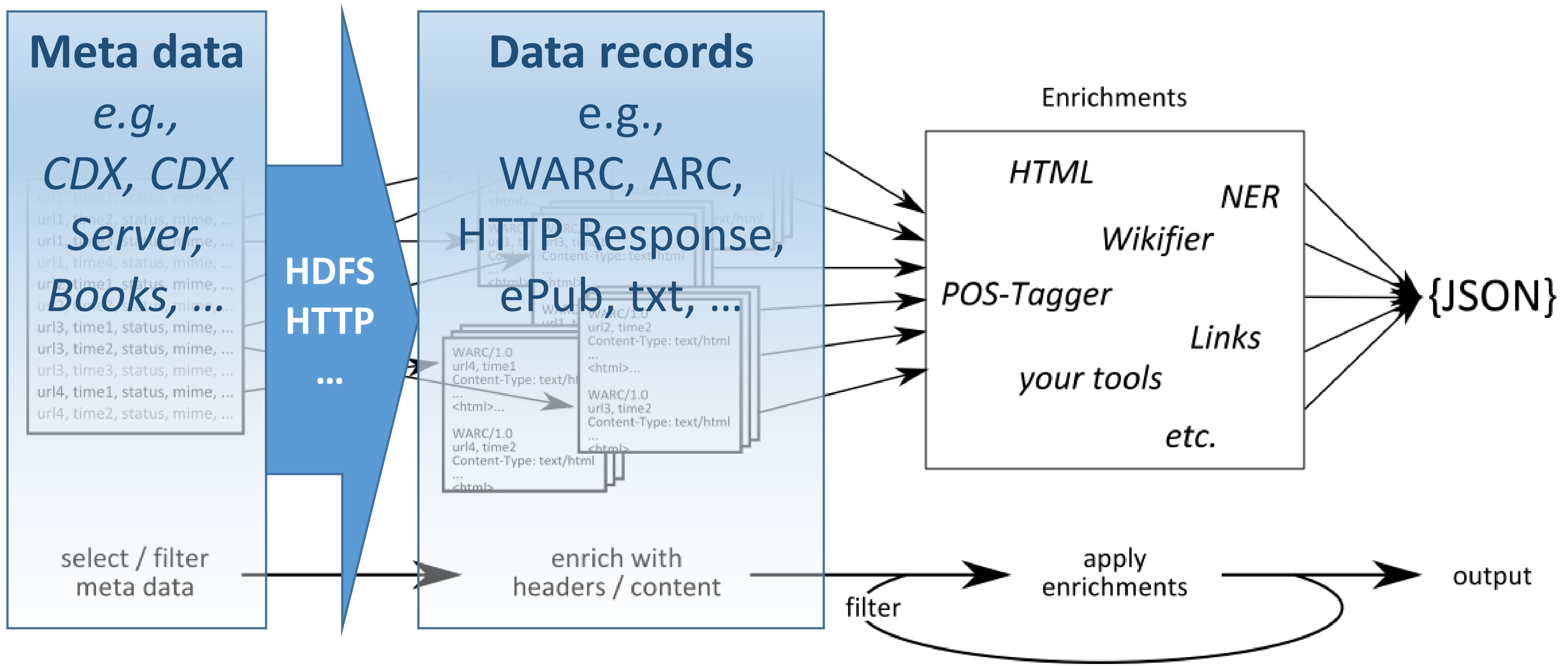
(b)



(c)

ArchiveSpark 2.0

- Generalize the concept to any index / data
- Enable any data sources, e.g., HTTP, …



Conclusion

- **Expressive** and **efficient** corpus creation from Web archives
- Easily **extensible**
  - Please provide enrich functions of your own tools / third-party libraries
- **Open source**
  - Fork us on **GitHub**: https://github.com/helgeho/ArchiveSpark
  - Star, contribute, fix, spread, **get involved!**
- Please **cite**
  - *Helge Holzmann, Vinay Goel, Avishek Anand.*
    ArchiveSpark: Efficient Web Archive Access, Extraction and Derivation.
    In Proceedings of JCDL, Newark, New Jersey, USA, 2016.

Helge Holzmann (holzmann@L3S.de)

Thank you!

https://github.com/helgeho/ArchiveSpark