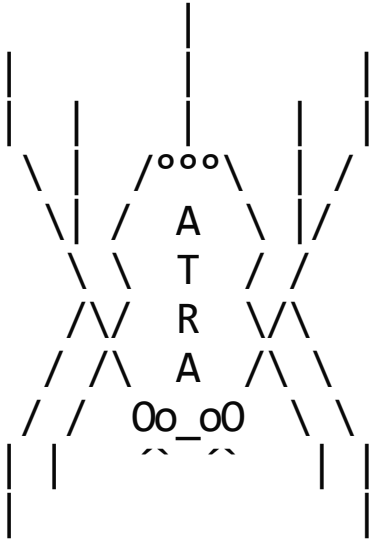


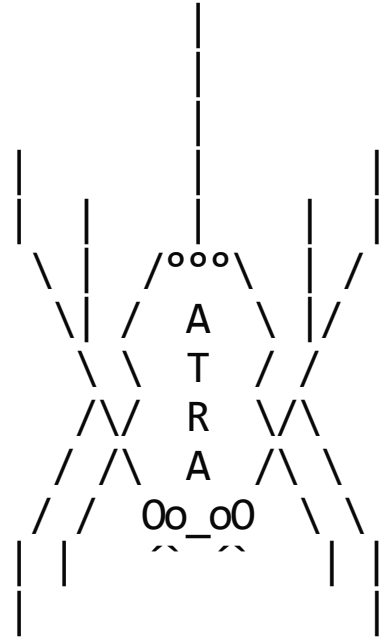
Atra

A powerful, lightweight approach to crawling

Universität Bamberg
Lehrstuhl für Medieninformatik

Atra

- Motivation
- Key Features
- Roadmap
- Demo

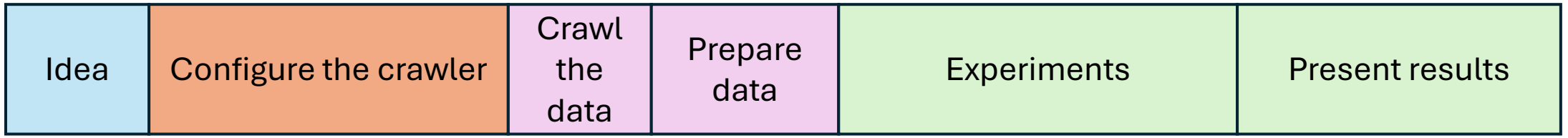


Motivation

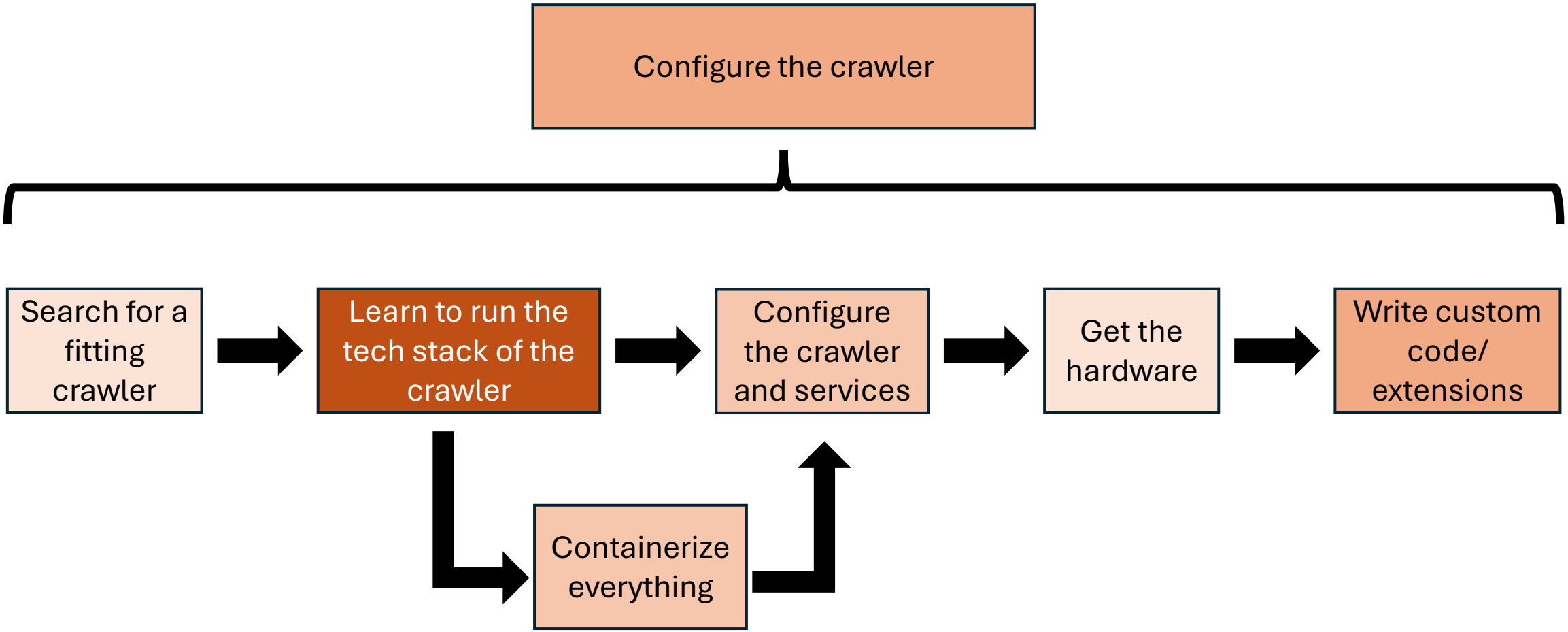
Perfect World

The usual for experts

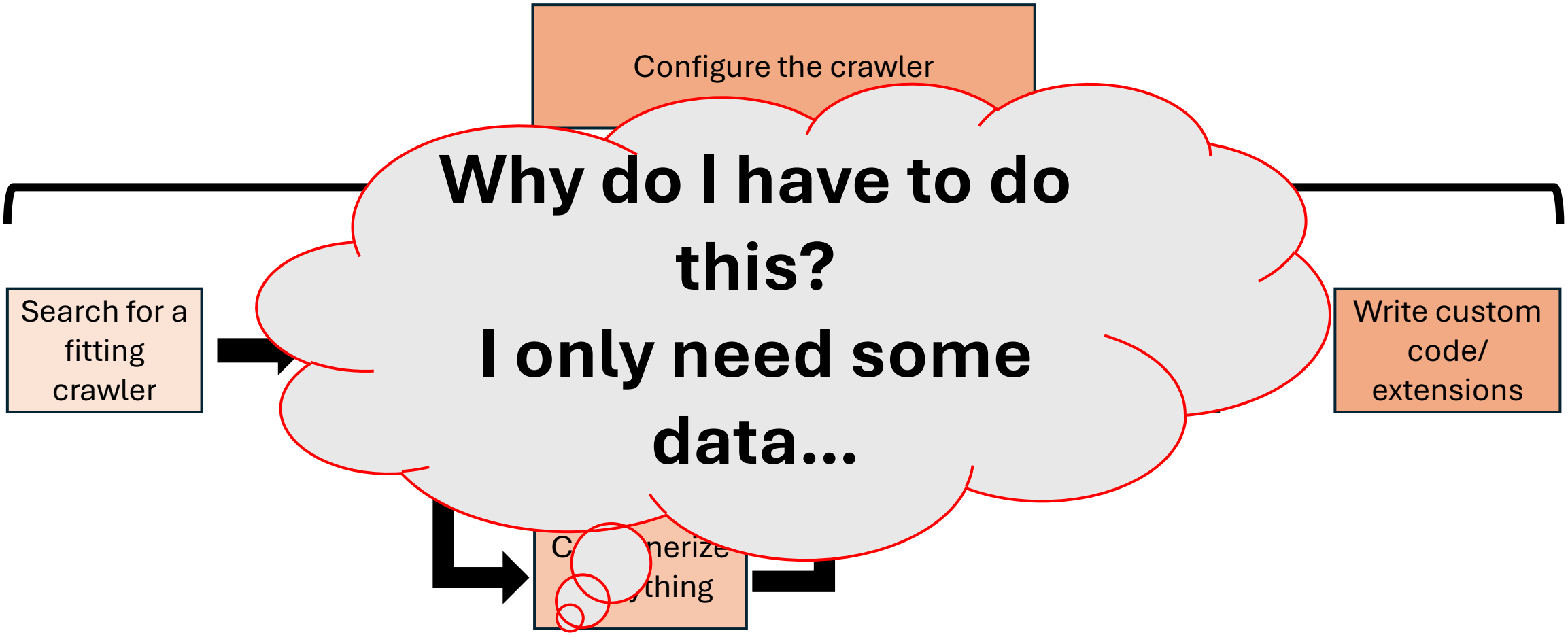
New to crawling



Motivation


Configure the crawler
Search for a fitting crawler
Learn to run the tech stack of the crawler
Configure the crawler and services
Get the hardware
Write custom code/ extensions
Containerize everything

Motivation


Configure the crawler
Why do I have to do this?
I only need some data...
Search for a fitting crawler
Write custom code/ extensions
C nerize
thing

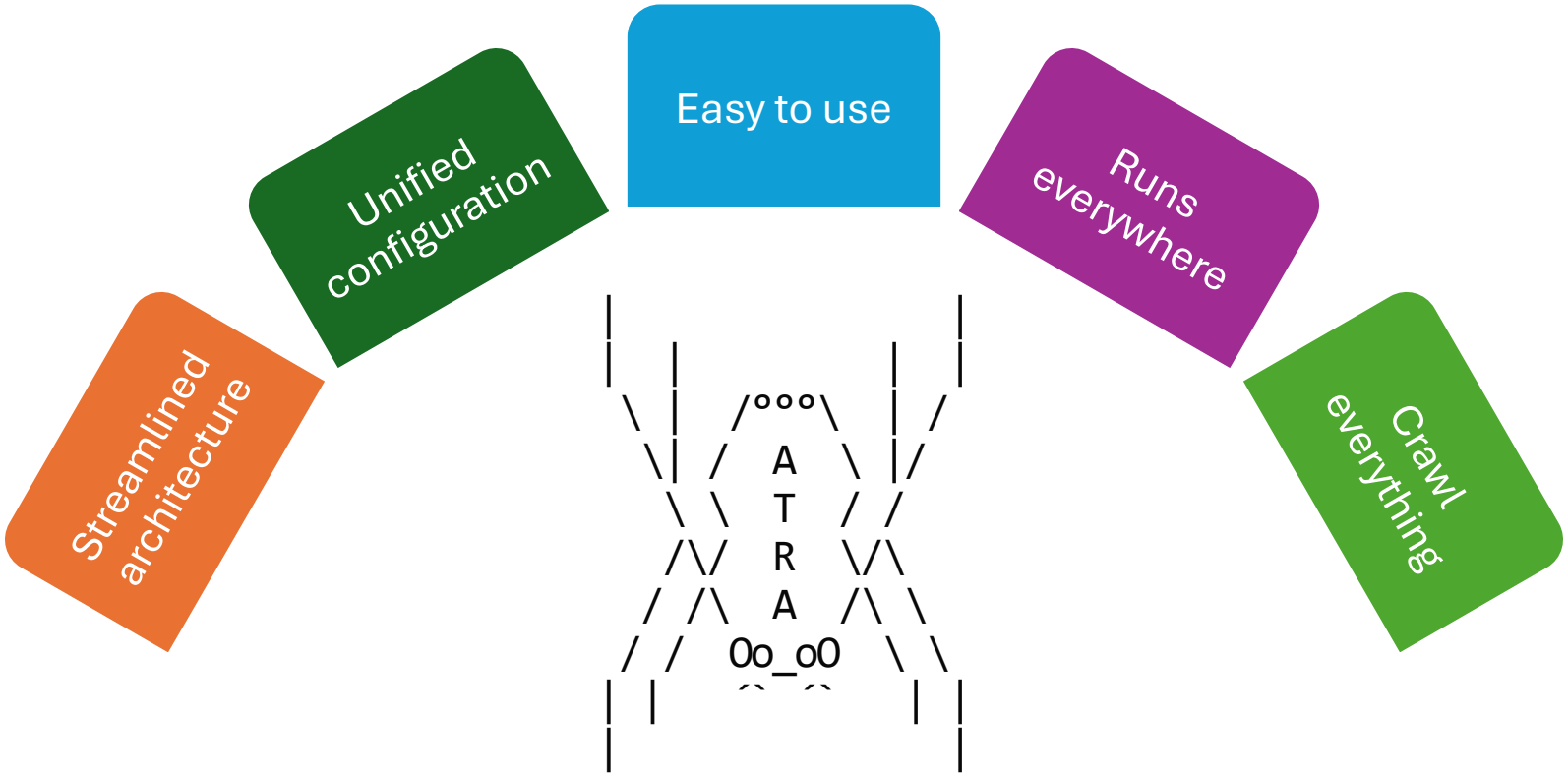
Motivation

Perfect World

The usual for experts

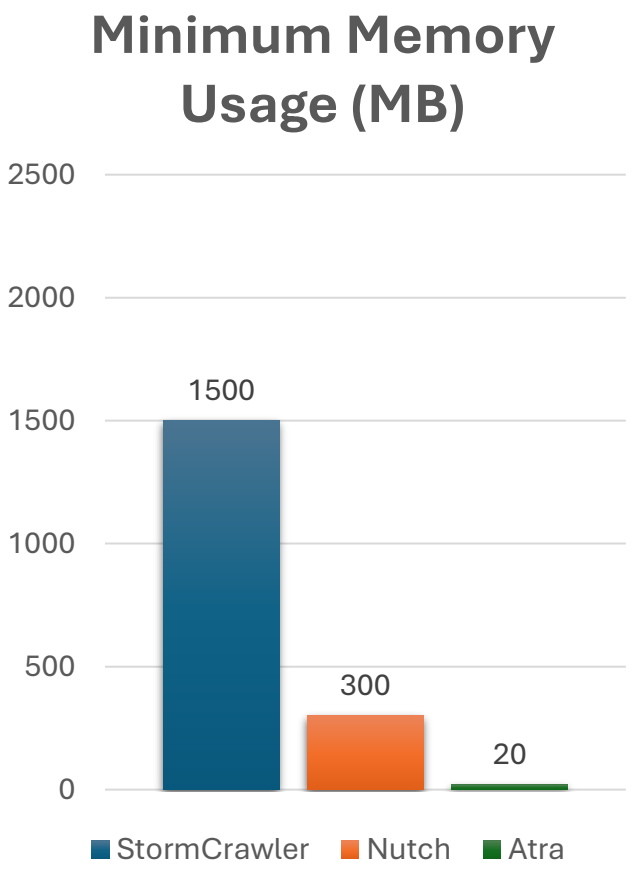
New to crawling, but with Atra

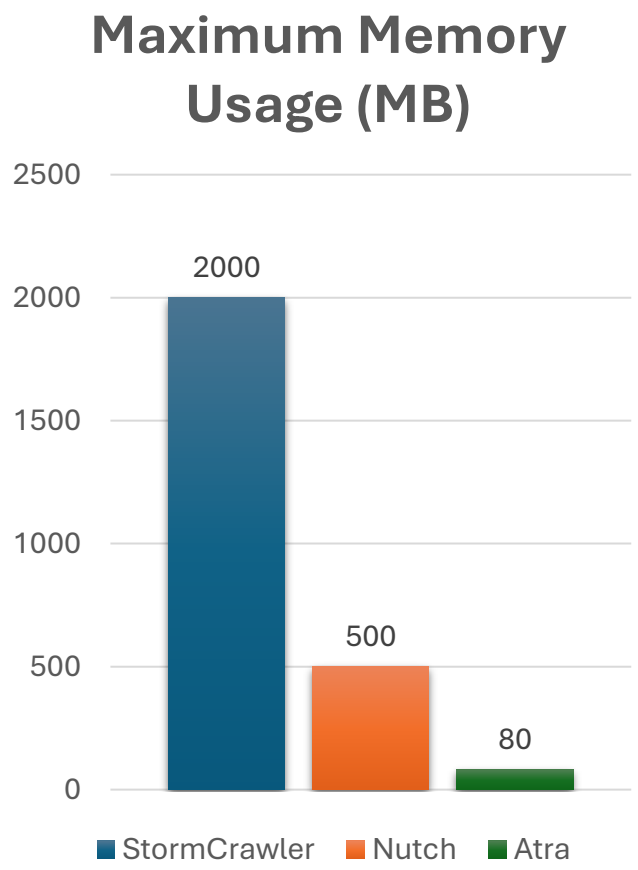
Atra

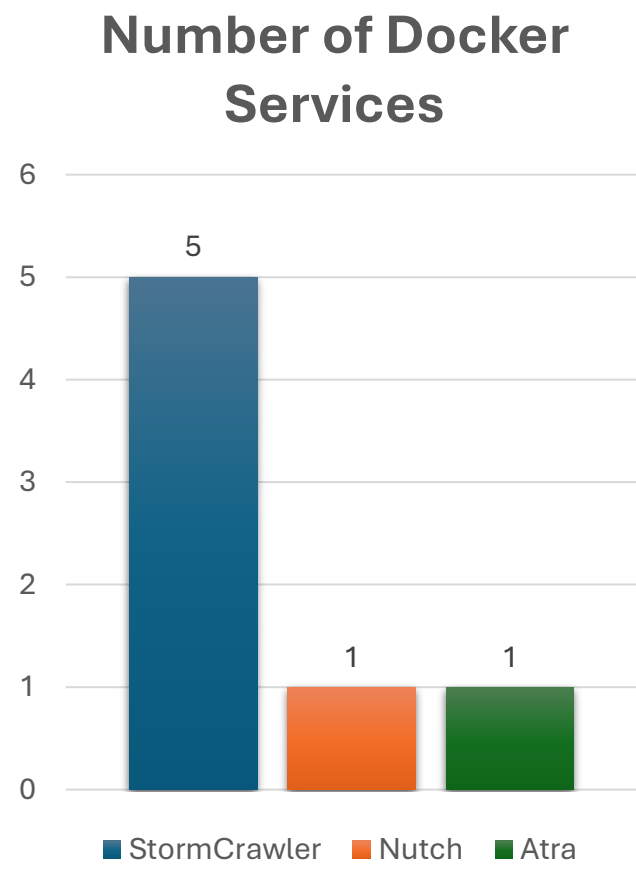

Easy to use
Unified configuration
Runs everywhere
Streamlined architecture
Crawl everything
A T R A
0o_o0

Key Features

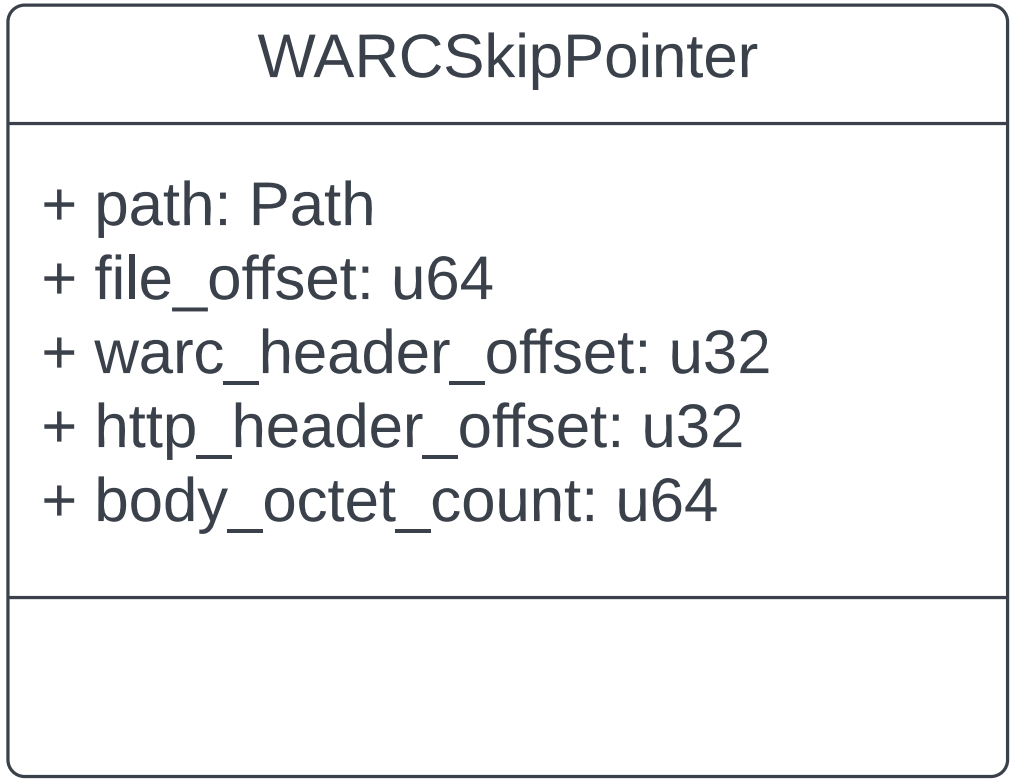
Efficient Resource Usage



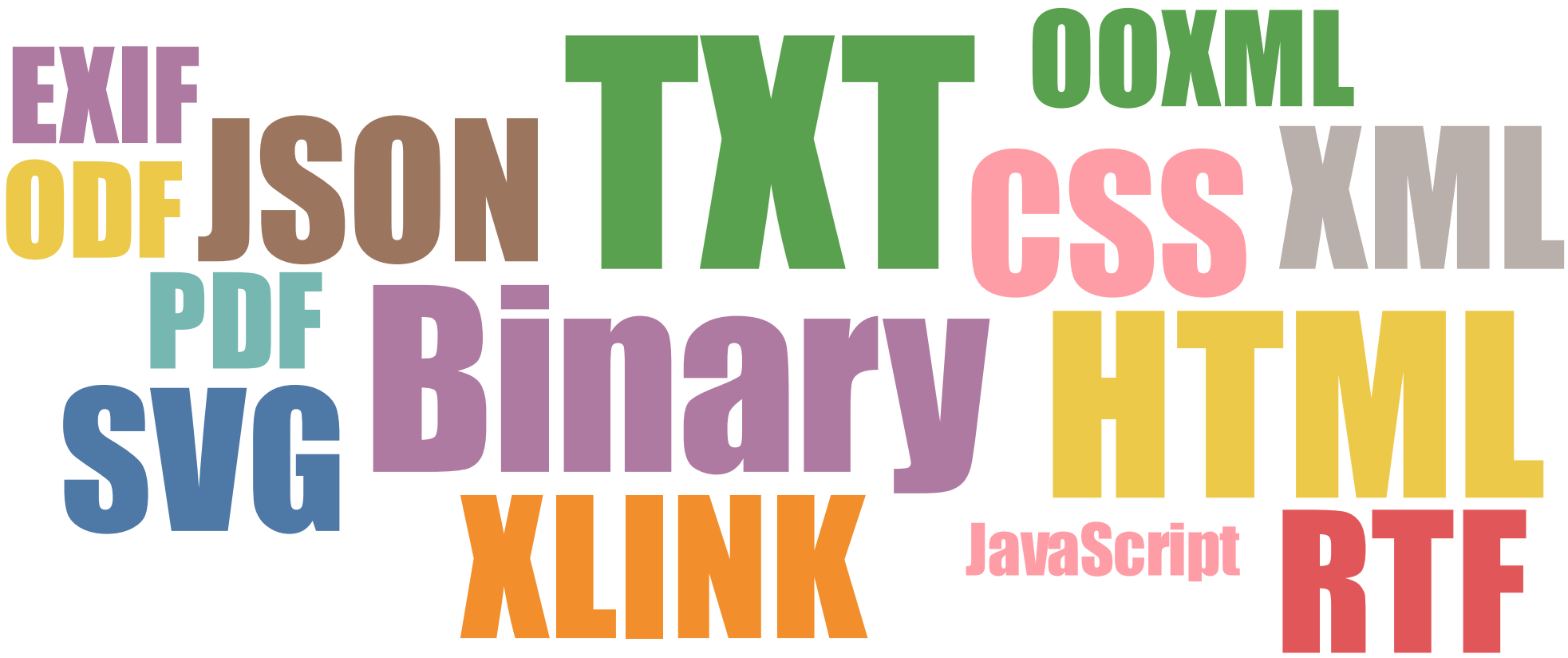




WARC Skip-Pointer



Link Extraction



Link Extraction – HTML

The usual way

Apache StormCrawler

```
final Elements links = jsoupDoc.select("a[href]");
```

Apache Nutch

<a>, <area>, <link>
<frame>, <iframe>, <script>, <img>, <source>
<form>

Atra

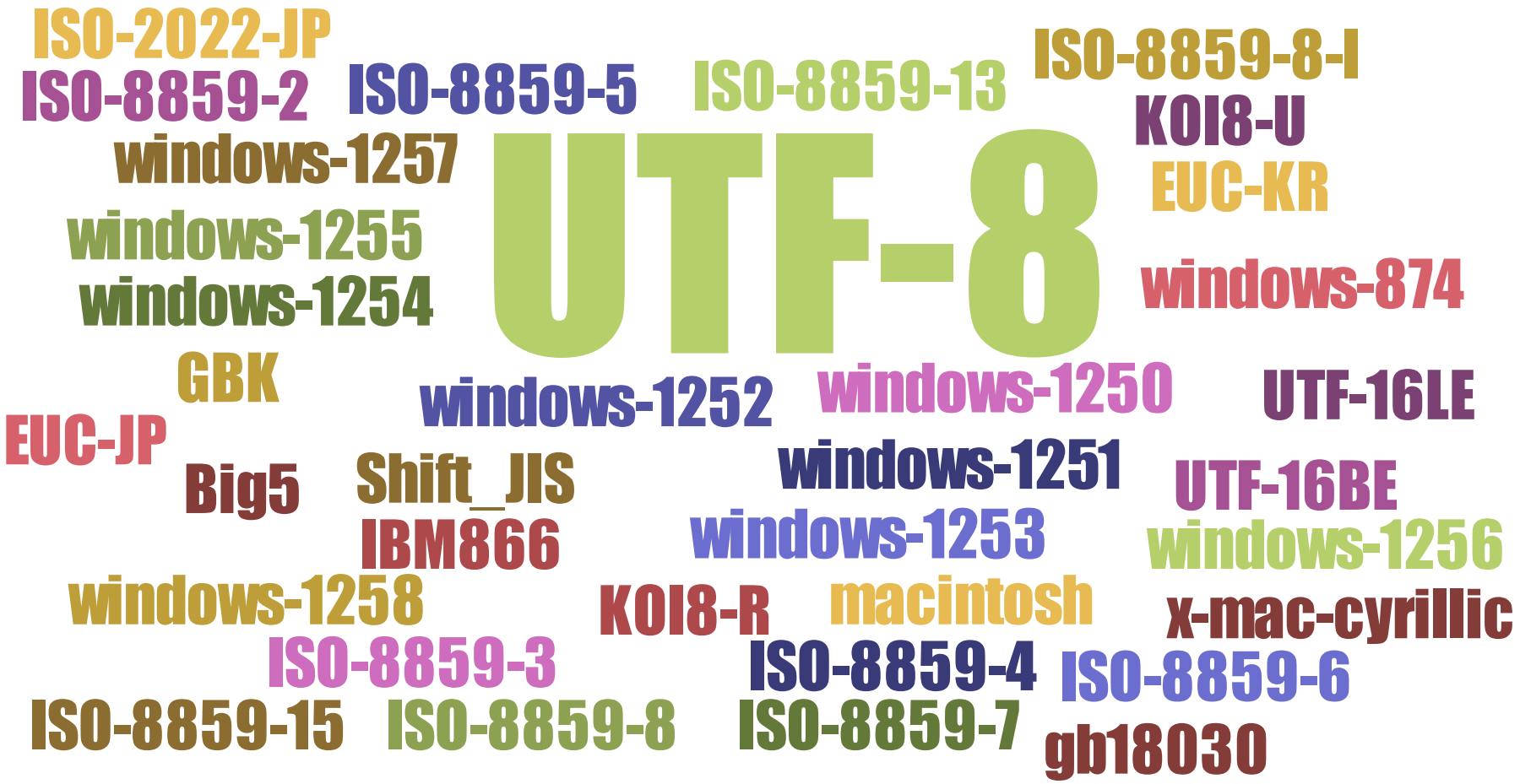
```
/// A matcher for href locations
2 usages   Felix Engl +1
pub static HREF_LOCATION_MATCHER: Lazy<Regex> =
    Lazy::new(|| Regex::new( re: "location\\s*\\.\\s*href\\s*=\\s*'\\s*([^']*)\\s*'\\s*;?")

// Ignore [ping] of area/a
static_selectors! {
    pub [
        BASE = "base"
        HREF_HOLDER = "a,area,link"
        SRC_HOLDER = "audio,embed,iframe,img,input,source,track,video"
        SCRIPT_HOLDER = "script"
        ON_CLICK = "[onclick]"
        FORM_HOLDER = "form[action]"
        META_NO_FOLLOW = "meta[name=\"robots\"][content=\"nofollow\"]"
    ]
}
```

Link Extraction - Binary

- Phase 1:
  - Scan the data source and keep consecutive valid UTF-8 bytes in a **buffer**.
  - If the **buffer** fulfills one of these conditions, go to Phase 2.
    - End with "://".
    - Contains "." and has at least 4 chars.
  - If the scan encounters invalid codepoints clear the **buffer**.
  - Continue scan.
- Phase 2:
  - Consume valid UTF-8 codepoints from the source until reaching a whitespace or an invalid codepoint.
  - Use linkify to extract valid links.
  - Clear the **buffer**.
  - Go to Phase 1.

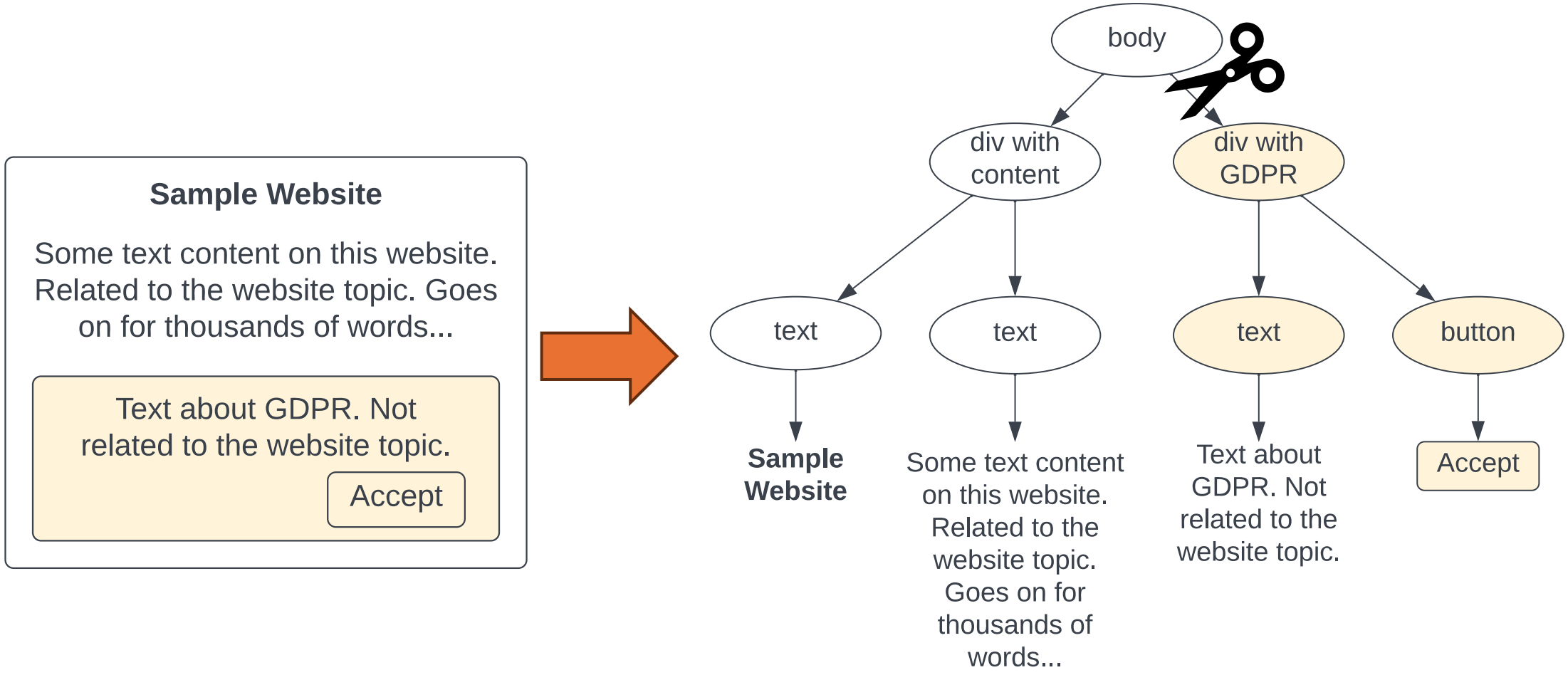
Encoding support



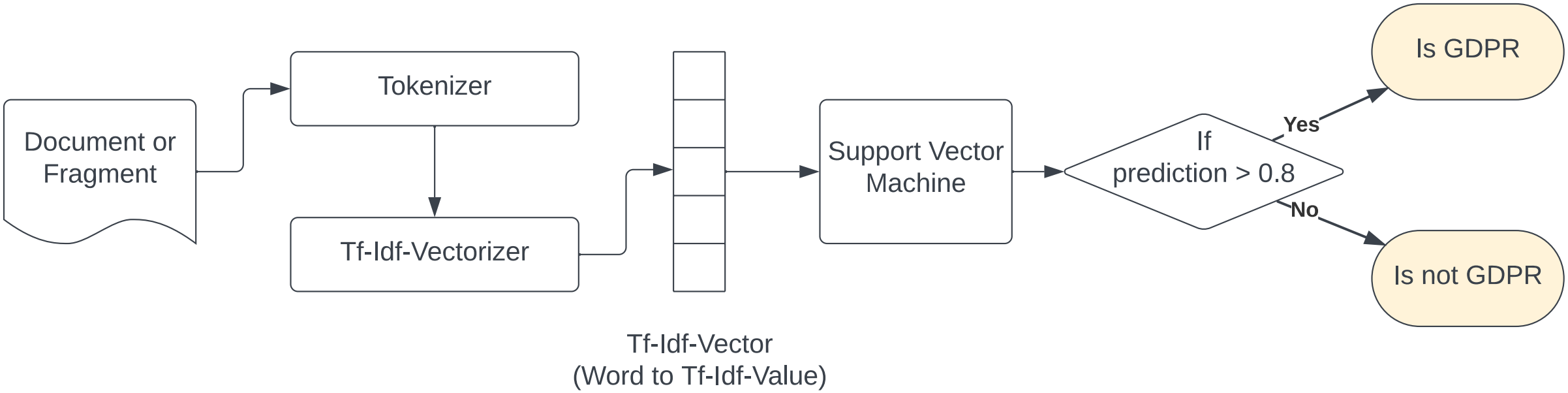
Encoding support

Atra uses the same decoding mechanism as Gecko Version 54+

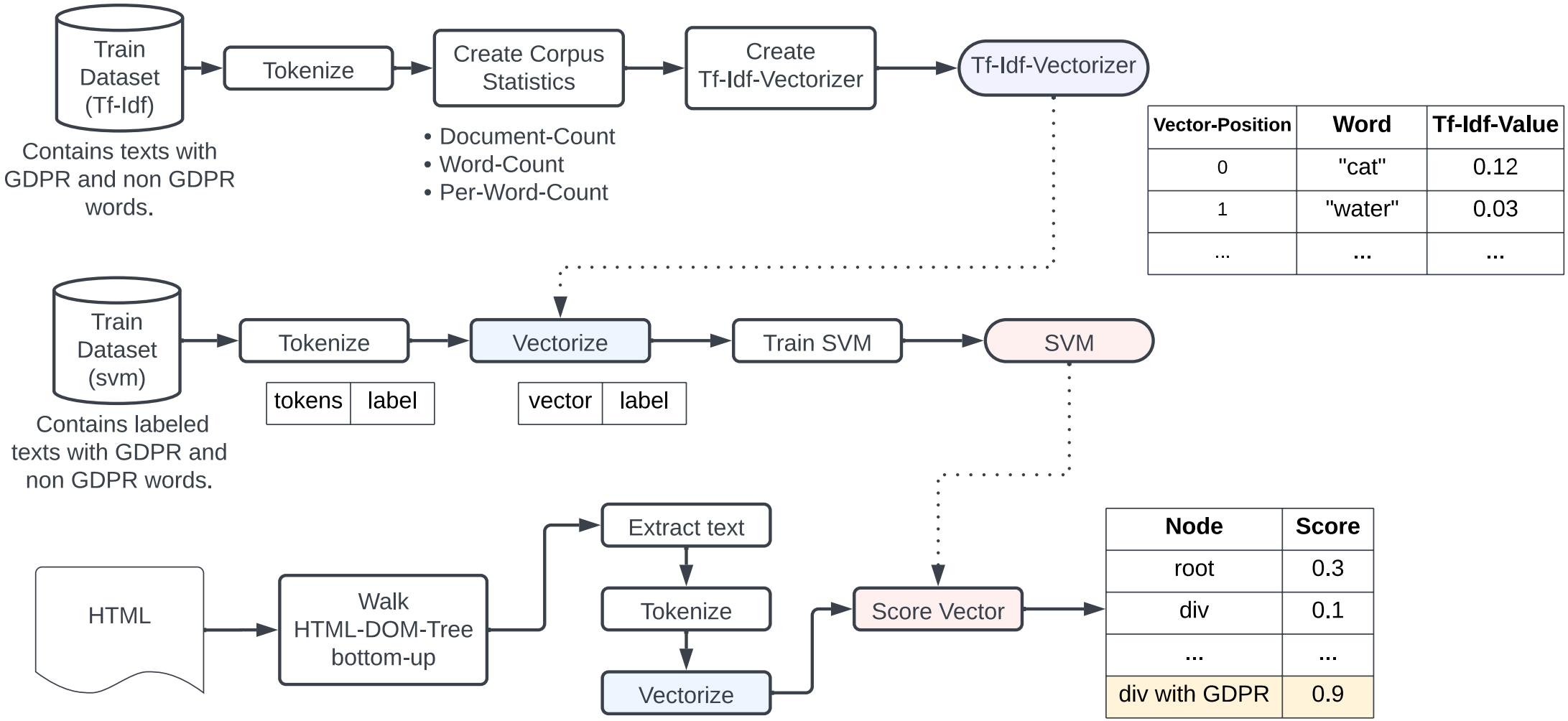
GDPR Filter


Sample Website
Some text content on this website. Related to the website topic. Goes on for thousands of words...
Text about GDPR. Not related to the website topic.
Accept
body
div with content
div with GDPR
text
text
text
button
Sample Website
Some text content on this website. Related to the website topic. Goes on for thousands of words...
Text about GDPR. Not related to the website topic.
Accept

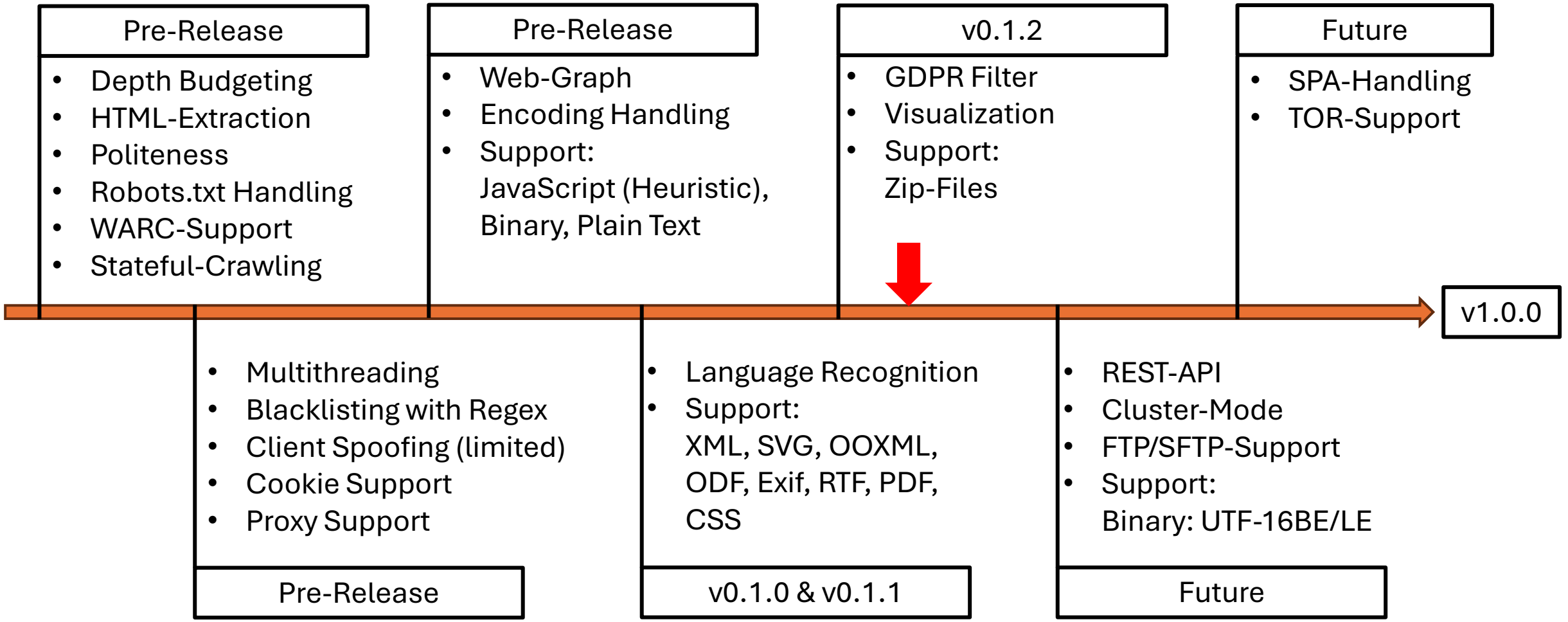
GDPR Filter


Document or Fragment
Tokenizer
Tf-Idf-Vectorizer
Tf-Idf-Vector
(Word to Tf-Idf-Value)
Support Vector Machine
If prediction > 0.8
Yes
No
Is GDPR
Is not GDPR

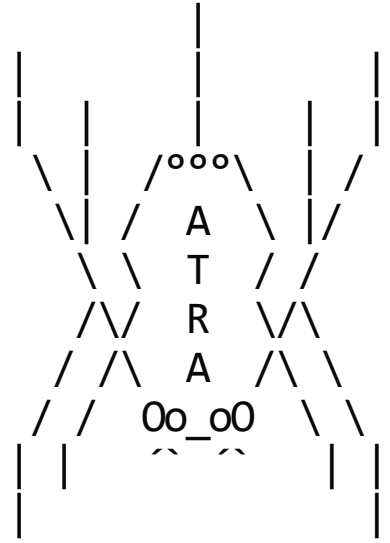
GDPR Filter


Train Dataset (Tf-Idf)
Contains texts with GDPR and non GDPR words.
Tokenize
Create Corpus Statistics
• Document-Count
• Word-Count
• Per-Word-Count
Create Tf-Idf-Vectorizer
Tf-Idf-Vectorizer
Vector-Position | Word | Tf-Idf-Value
0 | "cat" | 0.12
1 | "water" | 0.03
... | ... | ...
Train Dataset (svm)
Contains labeled texts with GDPR and non GDPR words.
Tokenize
tokens | label
Vectorize
vector | label
Train SVM
SVM
HTML
Walk HTML-DOM-Tree bottom-up
Extract text
Tokenize
Vectorize
Score Vector
Node | Score
root | 0.3
div | 0.1
... | ...
div with GDPR | 0.9

Roadmap


Pre-Release
Depth Budgeting
HTML-Extraction
Politeness
Robots.txt Handling
WARC-Support
Stateful-Crawling
Pre-Release
Web-Graph
Encoding Handling
Support:
JavaScript (Heuristic),
Binary, Plain Text
v0.1.2
GDPR Filter
Visualization
Support:
Zip-Files
Future
SPA-Handling
TOR-Support
v1.0.0
Multithreading
Blacklisting with Regex
Client Spoofing (limited)
Cookie Support
Proxy Support
Pre-Release
Language Recognition
Support:
XML, SVG, OOXML,
ODF, Exif, RTF, PDF,
CSS
v0.1.0 & v0.1.1
REST-API
Cluster-Mode
FTP/SFTP-Support
Support:
Binary: UTF-16BE/LE
Future

Demo



https://github.com/FelixEngl/atra

Thank you for your attention!

Questions?