

# ELI ‘Pillar IV’ specification

## Protocol to synchronise ELI metadata

- Version: 1.0
- Date: 1 December 2022
- This document supersedes the ‘Specifications to describe ELI datasets’ document <sup>(1)</sup>.

### Contents

1. Abstract .....	2
2. Introduction .....	2
2.1. ELI Pillars I, II, III and IV .....	2
2.2. Conformance .....	3
3. ELI ‘Pillar IV’ protocol .....	3
3.1. Pillar IV commitment .....	3
3.2. ELI Sitemap .....	4
3.2.1. Example of an ELI Sitemap .....	4
3.2.2. ELI Sitemap conformance .....	4
3.2.3. Relation with web search engines .....	7
3.3. ELI update Atom feed .....	8
3.3.1. Example of an ELI update Atom feed .....	8
3.3.2. ELI update Atom feed conformance .....	8
3.3.3. Cases of massive updates to the complete dataset .....	10
3.3.4. Relation with traditional feed readers .....	10
3.4. Provide access to full-text content of the legislation .....	10
4. Annex: processing model .....	10
4.1. Processing model to retrieve ELI metadata .....	11
4.2. Processing model to retrieve full text of the legislation .....	12
5. Annex: abstract legal resources and consolidated versions .....	12

---

(1) [https://eur-lex.europa.eu/content/eli-register/ELI\\_dataset\\_description-EN.pdf](https://eur-lex.europa.eu/content/eli-register/ELI_dataset_description-EN.pdf).

## 1. Abstract

---

European legislation identifier (ELI) metadata are disseminated in structured metadata embedded in the web pages of Official Journal websites. ELI has provided an easy path for legislation publishers to share their data in an interoperable way.

The ELI ‘Pillar IV’ specifications describe a protocol by which a data consumer can retrieve the complete set of ELI metadata from a given ELI provider and can get daily updates. This fourth pillar is necessary to build systems that need to aggregate complete sets of ELI metadata.

The ELI ‘Pillar IV’ protocol is a combination of two protocols: Sitemaps and Atom feeds.

## 2. Introduction

---

In this document we use the terms ‘ELI providers’ (or ‘data provider’) and ‘ELI consumers’ (or ‘data consumers’) to refer respectively to organisations that publish ELI-compatible metadata and to systems/clients willing to retrieve and use those metadata.

### 2.1. ELI Pillars I, II, III and IV

---

As ELI progresses toward making legal metadata more reusable, it strives to lower the barrier for data consumers to retrieve the ELI metadata from one or more ELI providers. The role of the ELI ‘Pillar IV’ specification is to lower this barrier.

ELI is currently based on three pillars.

- Pillar I is the specification of ELI URI (uniform resource identifier) to identify legal resources, their linguistic variants and files.
- Pillar II is the ELI ontology that specifies how to describe these legal resources, linguistic variants and files with a common metadata structure.
- Pillar III is the specification of how to disseminate these metadata in web pages of legal portals, using RDFa (Resource Description Framework in Attributes) or JSON-LD (JavaScript Object Notation for Linked Data).

With these three pillars in place, a data consumer willing to use the metadata from an ELI provider must first crawl all the pages of the website of this provider and extract the metadata embedded inside these pages. This is a fairly complicated task because of the following points.

1. **Coverage.** A data consumer has no guarantee that it has crawled **all** the available ELI metadata from a given publisher.
2. **Freshness.** A data consumer has no guarantee that it has up-to-date data, unless it crawls the pages in a continuous way. Even by doing so, the delay might be too long to have an up-to-date dataset and would cause unnecessary traffic to the website being crawled.

To address these two points of ‘coverage’ and ‘freshness’, this document specifies a **protocol** that enables ELI consumers to retrieve:

1. the **exhaustive** list of all ELI legal resources from a given ELI provider, using a **Sitemap file**;
2. the list of the **last updated** ELI legal resources from an ELI provider, using an **Atom feed**.

## 2.2. Conformance

---

Key words MAY, MUST and SHOULD in this document are to be interpreted as described in [BCP 14](#).

- The key word MUST, when appearing in capital letters, means that the definition is an absolute requirement.
- The key word SHOULD or the adjective RECOMMENDED, when appearing in capital letters, mean that there may be valid reasons in certain circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
- The key word MAY, or the adjective OPTIONAL, mean that an item is truly optional.

## 3. ELI ‘Pillar IV’ protocol

---

To be compatible with the ELI ‘Pillar IV’ protocol, an ELI provider MUST provide the two channels described below: the ELI Sitemap and the ELI Atom feed. Implementing only one of them is not sufficient.

To implement Pillar IV, it is necessary to have ELI Pillar III implemented (inclusion of structured metadata in web pages).

### 3.1. Pillar IV commitment

---

The implementation of Pillar IV implies a long-term commitment to making sure the Sitemap and RSS (Really Simple Syndication) feeds remain functional and accessible. Any change to the implementation should be communicated well in advance to avoid disruption of services relying on Pillar IV harvesting. In particular, publishers MUST notify the ELI taskforce and ELI registry at [OP-ELI-MAILBOX@publications.europa.eu](mailto:OP-ELI-MAILBOX@publications.europa.eu) upon the following changes.

- Any change that may affect the Sitemap or Atom feed clients in a non-backward compatible way (e.g. changes in the way some XML (Extensible Markup Language) elements are disseminated) (see below).
- Change to the URL (uniform resource locator) where the Sitemap(s) or Atom feed(s) are accessible.
- Change to the update frequency of the Sitemap(s) or Atom feed(s).
- If a new Sitemap and its corresponding Atom feed is introduced, or if an existing Sitemap and its corresponding Atom feed are removed <sup>(2)</sup>.

---

<sup>(2)</sup> This does **not** include situations where a new file is created within the same sitemap if the 50 000 entries limit is reached (see below). This is to deal with situations concerning an ELI publisher publishing data on multiple websites with multiple domain names.

## 3.2. ELI Sitemap

---

An ELI provider willing to implement ELI Pillar IV MUST provide a Sitemap file (or set of Sitemap files) that provides the **complete** list of URIs for its ELI legal resources.

### 3.2.1. Example of an ELI Sitemap

---

An ELI Sitemap file looks like the following example.

```
<urlset
  xmlns="http://www.sitemaps.org/schemas/sitemap/0.9"
  xmlns:dct="http://purl.org/dc/terms/"
  dct:relation="http://country.xy/eli/eli-update-feed.atom"
>
  <url>
    <loc>http://country.xy/eli/law/2016/501/003/jo</loc>
    <lastmod>2016-03-08</lastmod>
  </url>
  <url>
    <loc>http://country.xy/eli/law/2016/501/002/jo</loc>
    <lastmod>2016-03-06</lastmod>
  </url>
  <url>
    <loc>http://country.xy/eli/law/2016/501/001/jo</loc>
    <lastmod>2016-03-06</lastmod>
  </url>
  <url>
    <loc>http://country.xy/eli/decree/2005/199/999/jo</loc>
    <lastmod>2016-09-05</lastmod>
  </url>
</urlset>
```

### 3.2.2. ELI Sitemap conformance

---

To be conformant, an ELI Sitemap file MUST adhere to the specifications described in this section.

#### *Conform to the Sitemap protocol*

An ELI Sitemap file MUST conform to the Sitemap protocol as specified at <https://www.sitemaps.org/protocol.html>.

#### *Publish under /eli/sitemap.xml*

The ELI Sitemap SHOULD be provided at the URI ‘.../eli/sitemap.xml’.

Following the Sitemap protocol, an ELI Sitemap at <http://example.com/eli> MUST only list an ELI that starts with <http://example.com/eli>. Hence, if an ELI provider publishes ELIs on multiple websites with multiple domain names, it MUST publish multiple ELI Sitemap files (e.g. <http://example.com/oj/eli/sitemap.xml> v <http://example.com/consolidations/eli/sitemap.xml>).

Also note that ‘all URLs listed in the Sitemap must use the same protocol (http, in this example) and reside on the same host as the Sitemap. For instance, if the Sitemap is located at <http://www.example.com/sitemap.xml>, it can’t include URLs from <http://subdomain.example.com>.’

This constraint from the Sitemap protocol means that if the ELI URI of an ELI provider starts with ‘[http://data.country.xy/eli/...](http://data.country.xy/eli/)’, the corresponding Sitemap MUST be published under ‘[http://data.country.xy/eli/...](http://data.country.xy/eli/)’.

However, the Sitemap protocol describes a possibility of ‘cross-submit’ <sup>(3)</sup>, where the Sitemap can reside on a different domain than the URLs advertised in the Sitemap. To publish a Sitemap for host [www.host1.com](http://www.host1.com) from [www.sitemaphost.com](http://www.sitemaphost.com):

‘[you need] to prove that you own (i.e. have the authority to modify files) [www.host1.com](http://www.host1.com). You can do this by modifying the robots.txt file on [www.host1.com](http://www.host1.com) to point to the Sitemap on [www.sitemaphost.com](http://www.sitemaphost.com).

In this example, the robots.txt file at <http://www.host1.com/robots.txt> would contain the line “Sitemap: <http://www.sitemaphost.com/sitemap-host1.xml>”. By modifying the robots.txt file on [www.host1.com](http://www.host1.com) and having it point to the Sitemap on [www.sitemaphost.com](http://www.sitemaphost.com), you have implicitly proven that you own [www.host1.com](http://www.host1.com). In other words, whoever controls the robots.txt file on [www.host1.com](http://www.host1.com) trusts the Sitemap at <http://www.sitemaphost.com/sitemap-host1.xml> to contain URLs for [www.host1.com](http://www.host1.com).’

In the case of EUR-Lex, what precedes means that even if the ELIs start with <http://data.europa.eu>, the Sitemap can be published under <http://eur-lex.europa.eu/eli/sitemap.xml>, provided that the robots.txt file on [data.europa.eu](http://data.europa.eu) indicates: ‘Sitemap: <http://eur-lex.europa.eu/eli/sitemap.xml>’. Note that this will probably trigger the recognition of the ELI Sitemap by major search engines, and the impact of this in terms of search engine optimisation has not been assessed.

#### *Include only legal resources URI, not legal expressions or formats*

The ELI Sitemap MUST only contain URIs of legal resources, not legal expressions or formats. The ELI Sitemap MUST NOT mix these URIs with other (non-ELI-related) URIs/URLs. See the annex for recommendations on how to handle abstract legal resources and consolidated texts.

In other words, the list of URIs provided MUST correspond to the set of web pages that a client needs to visit to fetch a complete ELI dataset for this ELI provider.

An ELI provider MAY provide other (non-ELI) Sitemaps for the purpose of general search engine indexing.

#### *Include only canonical ELIs, not ‘synonyms’ ELIs*

If the ELI implementation uses ‘synonyms’ ELI, these synonyms MUST NOT be included in the Sitemap.

Again, the list of URLs in the Sitemap file MUST correspond to the set of web pages that a client needs to visit to fetch a complete ELI dataset for this ELI provider. So, if canonicals ELIs are included, there is no need to include ‘synonyms’ ELIs.

#### *Provide ‘loc’ and ‘lastmod’*

An ELI Sitemap file MUST provide at least the following two XML tags on each URL elements.

- Loc: contains the ELI URI of a legal resource.
- Lastmod: indicates the last modification date of the legal resource. This date should be in World Wide Web Consortium (W3C) datetime format. This format allows you to omit the time portion, if desired, and use YYYY-MM-DD.

---

<sup>(3)</sup> Sitemap cross submits: <https://www.sitemaps.org/protocol.html#location>.

An ELI Sitemap file MAY provide more Sitemap elements on each entry, but with no guarantee that an ELI client will use them.

An ELI Sitemap file MAY extend the content of each entry in the Sitemap with additional metadata (see the Sitemap protocol). In particular, the ELI/XML schema enables the expression of ELI metadata in an XML document. An example of the inclusion of an act title MAY look like the following using ELI/XML.

```
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9" xmlns:eli="http://data.europa.eu/eli/ontology#">
  <url>
    <loc>http://country.xy/eli/law/2016/501/003/jo</loc>
    <lastmod>2016-03-08</lastmod>
    <eli:LegalResource eli:URI="http://country.xy/eli/law/2016/501/003/jo">
      <eli:LegalExpression eli:URI="http://country.xy/eli/law/2016/501/003/jo/spa">
        <eli:title>Here is the title</eli:title>
      </eli:LegalExpression>
    </eli:LegalResource>
  </url>
</urlset>
```

It MAY look like the following using the Dublin Core terms metadata vocabulary.

```
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9" xmlns:dct="http://purl.org/dc/terms/">
  <url>
    <loc>http://country.xy/eli/law/2016/501/003/jo</loc>
    <lastmod>2016-03-08</lastmod>
    <dct:title xml:lang="es">Here is the title</dct:title>
  </url>
```

### *Provide link to corresponding Atom feed*

The Sitemap MAY provide the link to the URL pointing to the Atom feed that provides the updates to the URIs listed in the Sitemap. This is declared using a ‘dct:relation’ attribute in the root XML element.

```
<urlset
  xmlns="http://www.sitemaps.org/schemas/sitemap/0.9"
  xmlns:dct="http://purl.org/dc/terms/"
  dct:relation="http://country.xy/eli/eli-update-feed.atom"
>
...
</urlset>
```

This information is indicative only and will not be used by clients.

### *Split if more than 50 000 entries*

As described in the Sitemap protocol, if there are more than 50 000 entries, the Sitemap MUST be split into multiple files and a Sitemap index file MUST be provided. In that case, it is the Sitemap index that needs to be published under /eli/sitemap.xml.

The exact naming of individual Sitemap files referenced from the Sitemap index is left open and can be chosen by each ELI provider. As an indication, each Sitemap file MAY be suffixed with a number: '/eli/sitemap1.xml', '/eli/sitemap2.xml', etc.

#### ***Provide dereferenceable ELI URIs***

The ELI URIs provided in the loc XML elements, when accessed with the HTTP (Hypertext Transport Protocol) header 'Accept: text/html', MUST return to the client the HTML page that contains the metadata description of the corresponding legal resource URI, possibly using an HTML redirect.

In other words, a client MUST be able to issue the following call and get the HTML containing RDFa in return.

```
curl -L -H 'Accept:text/html' http://country.xy/eli/law/2016/501/003/jo
```

The RDFa or JSON-LD contained in the retrieved HTML page MUST describe the legal resource identified by the URI that was requested to the server.

#### ***Provide relevant multilingual legal expressions and formats in the RDFa metadata (at least one expression and one format)***

The RDFa or JSON-LD contained in the retrieved HTML MUST describe at least one legal expression and one format for this legal expression. The metadata MAY describe all legal expressions and all formats of these legal expressions, in order to provide clients with a complete view of the FRBR (Functional Requirements for Bibliographic Records) tree for the legal resource, whatever linguistic variant it retrieves.

An ELI publisher is thus free to exclude, for example, unofficial translations of a legislation from the legal expressions described by the metadata or exclude unofficial formats from the metadata.

Note that, in the case where the HTML page returned contains structured metadata for only the legal resource and one of its legal expressions, while the description of other legal expressions would be contained in other pages, the legal resource MAY refer to these other legal expressions using the 'eli:is\_realized\_by' link, even if they are not described in the page. This can help a client to follow these links and retrieve the metadata of these expressions from other pages.

#### ***Update monthly***

The ELI Sitemap must be updated at least every month.

#### ***Advertise the ELI Sitemap(s) location***

The ELI Sitemap(s) (or Sitemap index(-es)) location MUST be advertised by the ELI provider and MUST be notified to the ELI registry in EUR-Lex.

### **3.2.3. Relation with web search engines**

---

An ELI provider that publishes an ELI Sitemap as described in this specification is not required to submit the ELI Sitemap to a general web search engine.

### 3.3. ELI update Atom feed

---

#### 3.3.1. Example of an ELI update Atom feed

---

An ELI update Atom feed looks like the following example.

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom">

  <title>Fictitious Country.xy ELI update feed</title>
  <link rel="self" href="http://country.xy/eli/eli-update-feed.atom" type="application/atom+xml"/>
  <updated>2016-03-08T16:20:00Z</updated>
  <author>
    <name>Country Legislation Service</name>
  </author>
  <id>urn:country-xy:eli:eli-update-feed</id>

  <entry>
    <title>http://country.xy/eli/law/2016/501/003/jo</title>
    <link href="http://country.xy/eli/law/2016/501/003/jo"/>
    <id>http://country.xy/eli/law/2016/501/003/jo</id>
    <updated>2016-03-08T16:20:00Z</updated>
  </entry>

  <entry>
    <title>http://country.xy/eli/law/2016/501/002/jo</title>
    <link href="http://country.xy/eli/law/2016/501/002/jo"/>
    <id>http://country.xy/eli/law/2016/501/002/jo</id>
    <updated>2016-03-07T09:20:00Z</updated>
  </entry>

  <entry>
    <title>http://country.xy/eli/law/2016/501/001/jo</title>
    <link href="http://country.xy/eli/law/2016/501/001/jo"/>
    <id>http://country.xy/eli/law/2016/501/001/jo</id>
    <updated>2016-03-06T16:20:00Z</updated>
  </entry>

  <entry>
    <title>http://country.xy/eli/decree/2005/199/999/jo</title>
    <link href="http://country.xy/eli/decree/2005/199/999/jo"/>
    <id>http://country.xy/eli/decree/2005/199/999/jo</id>
    <updated>2005-09-06T16:20:00Z</updated>
  </entry>

</feed>
```

#### 3.3.2. ELI update Atom feed conformance

---

To be conformant, an ELI update Atom feed MUST adhere to the specifications described in this section.

*Conform to the Atom format*



An ELI update Atom feed **MUST** conform to the Atom specification for which a validator is provided on the W3C website <sup>(4)</sup>.

### *Provide minimal header information*

The feed root element **MUST** contain the mandatory Atom elements below.

- Title: the title of the feed, in any language.
- Link rel="self": the URL at which the ELI update Atom feed is published (see below).
- Updated: the date of the most recent entry in the feed.
- Author/name: the name of the service in charge of publishing the ELI update feed.
- ID: a unique ID for the feed. This is not the same as the publication URL of the feed, as the ID should remain the same if the feed changes URL or is copied or republished. The ID **SHOULD** be generated in the following way:
  - use scheme 'urn:';
  - append the domain name of the website where the feed is published, with dots replaced by hyphens (e.g. 'country-xy' and not 'country.xy');
  - append 'eli:eli-update-feed'.

Other Atom attributes **MAY** be used in the feed header.

Other XML elements (not in the Atom namespace) **MAY** be used in the feed.

### *Provide minimal entry information*

Each entry element in the feed **MUST** contain the following mandatory XML elements:

- link href="...": **MUST** be the ELI of the legal resource;
- ID: **MUST** be the ELI of the legal resource;
- title: **SHOULD** be the title of the legal resource, but **MAY** be the ELI of the legal resource if the title is not easily available;
- updated: **MUST** be the date of publication or last update of the legal resource.

Other Atom attributes **MAY** be used in each entry.

Other XML elements (not in the Atom namespace) **MAY** be used in each entry.

### *Publish under /eli/eli-update-feed.atom*

The feed **SHOULD** be provided at the URI '/eli/eli-update-feed.atom' (that is, under the same domain name as the ELI Sitemap).

This is not a mandatory requirement. The ELI update feed **MAY** be provided under another domain name, as long as it is advertised properly in the ELI registry (see below).

### *Update daily*

The ELI update Atom feed **MUST** be updated on a daily basis.

### *Keep 60 days of history in the feed*

The feed **MUST** contain at least 60 days of history of legal resources updates. This corresponds to one full cycle of the Sitemap update, plus one cycle of additional buffer.

---

<sup>(4)</sup> <https://validator.w3.org/feed/>.

This time frame allows a data consumer that has retrieved an ELI Sitemap, possibly 1-month old, and possibly involving a few days of processing time, to be sure to not miss any update to a legal resource since the ELI Sitemap was last updated.

### *Provide one ELI update feed per ELI Sitemap*

If multiple ELI Sitemaps are provided, because ELIs are published in multiple domains or multiple websites, then one corresponding ELI update feed **MUST** be published for each ELI Sitemap.

### *Advertise the ELI update feed(s) location*

The ELI update feed(s) location **MUST** be advertised by the ELI provider and **MUST** be notified to the ELI registry in EUR-Lex, in relation to the corresponding ELI Sitemap file (that is, the notification **MUST** advertise a pair of URL: the Sitemap URL and the corresponding Atom feed URL).

### 3.3.3. Cases of massive updates to the complete dataset

---

The ELI update Atom feed is intended to notify clients about new legal resources or updates to (the metadata of) legal resources (e.g. a keyword was added). It is not meant to notify clients if there is a massive update of the entire dataset for technical reasons, for example when a new metadata is added on all resources, or when the data model undergoes a significant evolution and the backlog is reprocessed.

In this case, the client should do a complete update of the dataset by starting again from the Sitemap to go through the complete list of URIs.

Put differently, it means that the update date indicated in the Sitemap and the Atom feed is the date of update of the resource in the system (regardless of legal considerations).

### 3.3.4. Relation with traditional feed readers

---

The ELI update Atom feed can be read by traditional feed readers and is thus a way to advertise newly published ELI.

While the ELI update Atom feed is designed for technical synchronisation, if human consumption is an important use case, ELI providers **SHOULD**:

- provide a human-readable title in the title element instead of the ELI URI;
- provide a content element with the abstract or description of the legal resource.

## 3.4. Provide access to full-text content of the legislation

---

ELI URIs provided in the Sitemap and Atom feed **SHOULD** give access to a file representation containing the full text of the legislation to allow for its full-text indexing by a search engine. A processing model is proposed in the annex.

## 4. Annex: processing model

---

This section is non-normative.

## 4.1. Processing model to retrieve ELI metadata

---

A client of the ELI ‘Pillar IV’ protocol can follow this algorithm to build the complete set of ELI metadata from a given ELI provider.

1. Input:
  - (a) the URL of an ELI Sitemap;
  - (b) the URL of an ELI Atom feed.
2. Get the initial state of the ELI dataset.
  - (a) At time  $tI$ , the client retrieves the ELI Sitemap of that publisher. This may involve going through a Sitemap index and processing multiple Sitemap files if there are more than 50 000 entries.
  - (b) The client stores the provided date of modification of each legal resource in the Sitemap.
  - (c) The client iterates on each ELI URI given in the Sitemap, and for each URI:
    - issues an HTTP request to this URI with the header ‘Accept: text/html’, and retrieves the corresponding HTML with embedded RDFa or JSON-LD;
    - parses the content of the page to extract the triples;
    - stores the triples (typically in a triplestore, in a named graph identified by the URI of the legal resource);
    - stores the provided date of modification in the Sitemap, associated to the URI (typically as a metadata on the corresponding named graph in the triplestore);
    - retrieves the full text of the act, using the algorithm described separately;
    - waits for 5 seconds (this waiting time is important to not put too much pressure on the server).

At this stage, the client has got a snapshot of the ELI dataset as it was at the time the ELI Sitemap was last published.

3. Get the delta between the initial state and now.
  - (a) At time ( $tI + \text{Sitemap iteration time}$ ), the client retrieves the ELI update Atom feed of that publisher (the one corresponding to the Sitemap initially iterated).
  - (b) The client iterates on each entry in the feed, and for each entry.
    - If the entry is not present in its database, the client must:
      - retrieve its metadata by following the same steps as above;
      - retrieve the full text of the act, using the algorithm described separately.
    - If the entry is present but the update date in the feed is more recent than the stored update date of that entry, the client must:
      - retrieve its metadata and update the corresponding record in its database (typically, by removing the corresponding named graph in the triplestore and replacing it with the new metadata extracted, and with the update date indicated in the feed) <sup>(5)</sup>;
      - retrieve and replace the full text of the act, using the algorithm described separately.

At this stage, the client has got a snapshot of the ELI dataset as it is now.

4. Put in place a daily synchronisation.
  - (a) Every day, the client pulls the ELI update Atom feed and applies the same algorithm that is iterated on each entry, and for each entry.
    - If the entry is not present in its database, the client must retrieve its metadata by following the same steps as above.

---

<sup>(5)</sup> It means in particular that, for an ELI that corresponds to the latest consolidated version of an act, whenever a new consolidated version of that act is published, the entry in the feed needs to be updated accordingly, and the Atom client will update it.

- If the entry is present but the update date in the feed is more recent than the stored update date of that entry, the client must retrieve its metadata and update the corresponding record in its database.

At this stage, the client can maintain the synchronisation of its database with the ELI metadata published by the ELI provider, on a daily basis.

5. Resynchronise the complete dataset. Every time it is necessary, or periodically (for instance every year or every 2 years), a client may decide to go through the complete dataset again by starting from the Sitemap, in order to retrieve an updated image of the complete dataset. This is to deal with cases where an important update is made on the dissemination of ELI metadata, while the legal resources themselves have not been updated (from a legal point of view).

## 4.2. Processing model to retrieve full text of the legislation

---

The algorithm to retrieve the full text of an act is the following.

1. Given the ELI metadata contained in the web page (with at least one legal expression and one format), a client of the ELI ‘Pillar IV’ protocol will, for each legal expression found, look for:
  - (a) an HTML format that is an ‘eli:Format’ with ‘eli:media\_type’ property set to <https://www.iana.org/assignments/media-types/text/html>;
  - (b) if not found, an XHTML format that is an ‘eli:Format’ with ‘eli:media\_type’ property set to <https://www.iana.org/assignments/media-types/application/xhtml+xml>;
  - (c) if not found, an XML format that is an ‘eli:Format’ with ‘eli:media\_type’ property set to <https://www.iana.org/assignments/media-types/application/xml>;
  - (d) if not found, a PDF format that is an ‘eli:Format’ with ‘eli:media\_type’ property set to <https://www.iana.org/assignments/media-types/application/pdf>;
  - (e) if not found, repeat the same sequence by looking at the ‘eli:Format’ property instead of ‘eli:media\_type’, looking at the same values.
2. Once an eli:Format is found, the client will:
  - (a) test if an ‘eli:is\_exemplified\_by’ property is present on the ‘eli:Format’, and, if present, use the URL value of the property as the address of the full-text content;
  - (b) if no ‘eli:is\_exemplified\_by’ is present, use the URI of the ‘eli:Format’ itself as the URL of the full-text content, in this case the URI of the ‘eli:Format’ MUST return the document containing the full text of the legislation, for example, if the ‘eli:Format’ URI is <http://country.xy/eli/law/2016/501/003/jo/fr/pdf> (indicating a PDF format of a French expression of an act), then that URI must return the PDF file of the French linguistic variant of that act.
3. Once a URL is obtained via the above algorithm, the client will dereference the URL, following redirects (‘curl -L http://...’).
4. The client MAY access more than one format (HTML, XHTML, XML or PDF) per language.
5. The client MAY repeat the process for each legal expression to retrieve the multilingual content of all languages.

## 5. Annex: abstract legal resources and consolidated versions

---

This section is non-normative.

This specification does not mandate how to deal with abstract legal resources or consolidated versions of the legislation. The rule to follow is that the ELI Sitemap and the corresponding Atom feed MUST correspond to the set of web pages that a client needs to visit to retrieve an exhaustive ELI dataset for that publisher. Whether that set of web pages should contain the abstract legal resources and/or all consolidations is a choice of the ELI publisher.

If the URI of an abstract legal resource does not correspond to an actual web page of the legislation and redirects to the page of the current consolidated version of this act, then the URI of abstract legal resources should probably not be included in the Sitemap. If, on the contrary, the URI of abstract legal resources return a web page that contains all the necessary metadata to describe this act, then the URI of abstract legal resources should probably be included in the Sitemap.

If all the history of consolidation is kept, and each new consolidated version has an ELI URI (usually including a timestamp), each new consolidated version URI should probably be included in the Sitemap to enable clients to get the complete view of the consolidation history of an act.