JeDEM 14(2): 57-82, 2022
ISSN 2075-9517
http://www.jedem.org
Date of submission: 29.04.2022
Date of acceptance: 16.11.2022
Article DOI: 10.29379/jedem.v14i2.733

A Translation Service for Open Data Portals

Sebastian Urbanek[1*], Sonja Schimmler[2]

[1*] *ORCID Nr: 0000-0003-0925-1781*
Berliner Hochschule für Technik, 13353 Berlin, Germany, sebastian.urbanek@bht-berlin.de

[2] *ORCID Nr: 0000-0002-8786-7250*
Fraunhofer FOKUS, 10589 Berlin, Germany, sonja.schimmler@fokus.fraunhofer.de

Abstract: There exists a huge variety of Open Data portals, some of them providing just a handful, and others, tens of thousands of datasets. The datasets they provide are expected to be supplied with metadata describing them. However, this metadata is typically available in one or two languages only, and, if translations exist, they are usually added manually. To build an inclusive data infrastructure, metadata should be available in as many languages as possible. The paper presents an approach for automatic translation of metadata within Open Data portals, based on Semantic Web technologies and using the metadata standard DCAT-AP. Based on this approach, new functionalities are possible, such as enabling users to search for datasets in their native language. The approach was implemented for and tested within a practical application in a production environment.

Keywords: Translation service, open data, fair data, semantic web, dcat-ap, linked data

Acknowledgement: This work was supported by the Federal Ministry of Education and Research of Germany (BMBF) under grant no. 16DII138 ("Deutsches Internet-Institut").

1. Introduction

In recent years, the Open Data movement has become increasingly important; new Open Data portals are being established, and the amount of available Open Data is growing rapidly. The primary purpose for a user is to get access to Open Data from public administration and other domains.

According to the Open Knowledge Foundation (2016), data is entitled 'open' if it uses an open licence, has no restrictions on access, and supports an Open Data format and machine readability. Besides these 'must haves', there are other conditions that should hold, which are derived from the definition of open knowledge (Open Knowledge Foundation, 2016), according to which, knowledge is open 'when anyone can freely access, use, modify and share it'.

Together with the understanding of 'Openness', the FAIR principles are becoming more common. FAIR stands for findability, accessibility, interoperability and reusability. The goal is users 'could more easily discover, access, appropriately integrate and re-use, and adequately cite' public data (Wilkinson et al., 2016).

In Open Data portals, datasets are provided with a title and a description so that users can quickly understand what is inside. One advantage is that these descriptions make it easier to retrieve datasets. The additional information that comes with each dataset is entitled 'metadata[1]'. Metadata provides information about datasets, and consists of several properties, like *Title* or *Description*. For Open Data originating from the public sector, a metadata standard has been defined - the Data Catalogue Vocabulary (DCAT) (Albertoni et al., 2020). The standard regulates how terms are used and defines the contents. Combined with the Resource Description Framework (RDF) as part of the Semantic Web technology package, the standard enables a knowledge graph representation to provide context and promote interoperability.

Zuiderwijk et al. (2015) examined in their work, which important factors exist in the publication of Open Data. They concluded that 'multilingualism' plays a crucial role in terms of accessibility. Open Data is mainly published in the local language or English. This excludes users that speak other languages. For this reason, it is essential to make metadata of a dataset available in as many languages as possible. Petychakis et al. (2014) evaluated how many datasets are available in one or more languages. They looked at Open Government Data portals. Their result was that 77.8% of the available datasets are offered in the native language of the particular portal. Only 22.19% provide two or more languages, of which the second language is mostly English.

Open Government Data portals, such as Slovenia's[2], use Google Translate for translation. This also allows texts to be displayed in the user's national language, including titles and descriptions of datasets. They are translated and displayed 'on-the-fly' in the user's browser. There is no selection or preprocessing of metadata and also no long-term storage or further use of the translations. This means, for example, that no search can be executed in the user's language.

This paper presents an approach for the automatic translation of metadata designed for Open Data portals, based on Semantic Web technologies and the DCAT-AP metadata standard. Translations are performed on a knowledge graph representation that is incrementally extended and updated. Incorporating the translations into a knowledge graph, as suggested by our approach, enables additional search and exploration capabilities. Furthermore, utilising a SPARQL endpoint allows translations to be used in addition to the original metadata. This opens up the possibility of developing further applications.

A software component, named Translation Service, was developed and integrated into a data management ecosystem that uses automatic translation and knowledge graph embedding to take up the approach. This data management ecosystem forms the technical basis for the European Data Portal, which serves as a use case for the approach's feasibility. For the translation of text snippets, attention is drawn to the use of existing engines.

1 https://www.merriam-webster.com/dictionary/metadata
2 https://podatki.gov.si

Section 2 elaborates on related work, and Section 3 lists the recommendations a translation service should fulfil within a European Data Portal context. The system's architecture is described in Section 4, and Section 5 discusses the technical details during the translation process. The use case is detailed in Section 6, and Section 7 concludes the paper.

2. Related Work

Open Data portals have the central task of making data available to the general public, in the form of a linked registry (Colpaert, P. et al., 2013). And yet, some barriers to publishing Open Data exist among data providers. These issues can lead to metadata not being complete and valuable (Janssen et al., 2012). The approach presented in this paper reduces the barrier of multilingualism, as metadata has to be provided in one language only. When translated automatically, it is possible to get a proper understanding of the original metadata and its describing files (van der Waal et al., 2014).

Each dataset usually has one or more attached files representing the dataset's content. It comes with metadata, consisting of several properties. As soon as there is a title and a description provided, datasets can already be published. Users can read through title and description, and portals can offer technical functions based on the metadata. This value increases as more metadata is added, such as tags, categories or information about formats. The use of metadata and metadata standards enables additional functionalities. For instance, datasets can be searched for, according to different criteria (Lnenicka & Nikiforova, 2021). In order to enable the re-use of datasets, metadata also includes information on usage licenses. This way, users can quickly see how they may use the data and files.

2.1 FAIR and CARE Principles

The FAIR principles, although formulated primarily for scientific data, are playing an increasingly important role in the general provision of data. FAIR is an acronym that stands for *Findable, Accessible, Interoperable* and *Reusable* (Wilkinson et al., 2016). Especially at a time when more and more data is produced, it is essential to improve its provision. In this context, the work on the FAIR data principles is no longer focused exclusively on research data, but the scope has been extended to all types of data and other artefacts.

Open Data is not automatically FAIR. More and more data is being shared as Open Data. However, this does not mean it is shared in a FAIR manner. Its quality might be poor and/or metadata might be missing (Higman et al., 2019).

Translations can help improve the FAIRness of Open Data, especially findability. By providing metadata in multiple languages, indexing and search engines can make use of translations. As a result, users can search for data in their own language and get the same results as those searching in the original language. Accessibility is also improved, as users can read through the metadata in their own language. By working with metadata standards which include translations, interoperability is improved. Reusability is also improved by enabling new functions, based on translations.

In addition to the FAIR principles, the CARE principles are also playing an increasingly important role. Although explicitly addressing indigenous groups, the CARE principles can be applied to other groups as well. Its principles have been formulated universally. CARE is an acronym that stands for *Collective Benefit, Authority of Control, Responsibility* and *Ethics*. Collective Benefit targets the disadvantages that people face who may not have any access or understanding. This disadvantage can be mitigated through an adequate description in the user's language, and hence, automatic translations can improve this criterion (Carroll et al., 2020).

2.2 Text Translation

Nowadays, it is possible to translate texts without extensive knowledge of a target language. Machine translation engines do this without human assistance. Modern translation engines are based on statistical methods, i.e. statistical machine translation (SMT), which analyse the frequency and the similarity of words or phrases in two different languages. If the similarity is high enough, one phrase is considered to be the translation of the other (Koehn, 2010). Neural machine translation (NMT) is a special type of statistical translation. An artificial neural network analyses the texts, and the network is trained for the translation with some better results, in comparison to SMT (Bahdanau, Cho, Bengio, 2016).

This NMT approach was taken up by the European Commission, named eTranslation[3][4]. It emerged in 2017 from a research project called MT@EC (Foti, 2012) and can translate text snippets as well as full document files.

The service is steadily evolving. By using neural engines, parts of a text (phrases) can be better identified and translated. eTranslation uses different training samples for different domains (or subjects) (Nurminen and Koponen, 2020). For example, for legal texts, a different set of training samples is used for texts on water cartography. eTranslation allows a selection of a domain to get more accurate results.

In addition to the translation functions, eTranslation offers a high level of infrastructure security. Translations are encrypted during transmission, and the servers are located in the European Union[5]. According to eTranslation, the submitted metadata will not be analysed. This seems to be a paradox for an Open Data portal at first glance since metadata is usually made available in these portals. Nevertheless, the metadata is often subject to specific usage licences, and hence, infrastructure security is required.

The European Union has identified 24 languages[6] as official languages in its principles. These official languages include Bulgarian, Croatian, Czech, Danish, Dutch, English, Estonian, Finnish,

3 https://www.lr-coordination.eu/sites/default/files/Brussels_conference/Mai-K_ELRC-MT(at)EC%20in%20DGT_26_10_2016_K.%20Mai.pdf

4 https://ec.europa.eu/info/resources-partners/machine-translation-public-administrations-etranslation_en

5 https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/Machine+translation?preview=/82773442/214794760/SLA%20eTranslation%20MT_v1.8.pdf

6 https://european-union.europa.eu/principles-countries-history/languages_en

French, German, Greek, Hungarian, Irish, Italian, Latvian, Lithuanian, Maltese, Polish, Portuguese, Romanian, Slovak, Slovenian, Spanish and Swedish. The eTranslation service is primarily concerned with translating texts into these languages. Other languages, such as Norwegian (Bokmål) and Russian, are also supported.

2.3 Standards

Multiple standards exist to describe data. These standards can enhance each other and extend the individual attributes of a standard (Alasem, 2009).

As of (Nuffelen, 2019), each dataset is described by its metadata, consisting of several properties. Mandatory properties are *Title* and *Description*. A dataset as a collection of data available for access or download in one or more formats. An associated file is linked as distribution to a dataset. Each distribution is described by its metadata. Distributions have an access URL as a mandatory property. *Title* and *Description* are optional properties.

One of the earliest metadata standards is the Dublin Core Standard (DC)[7]. It defines a unique identification number (ID) for a dataset. It also includes a format specification, a title, a description and a type. In addition, there is further information on persons and rights (Weibel and Koch, 2000).

Dublin Core was picked up and extended by DCAT, which was developed by the World Wide Web Consortium (W3C). The acronym stands for Data Catalog Vocabulary and is explicitly designed for the Resource Description Framework (RDF). It defines metadata to describe datasets and data services that can be used and exchanged in a standardised way (Albertoni et al., 2020).

RDF is a Semantic Web technology that the W3C designed. RDF defines a graph-based data exchange format that uses 'triples'. Each triple consists of a subject, a predicate and an object. Subject and object usually describe instances or values and are connected by a predicate (Hayes and Patel-Schneider, 2014). Any information can be linked with each other this way.

The serialisation[8] of Linked Data and RDF is realised through three metadata interchange formats. RDF can be represented as JSON (called JSON-LD[9]), XML (called RDF/XML[10]), or Turtle[11].

An extension of DCAT is DCAT-AP. The 'Data Catalog Vocabulary - Application Profile' was created for datasets in the Open Government domain. It is now well established in Open Data portals that focus on Open Government Data. Nevertheless, the DCAT and DCAT-AP standards can be used or adapted well for other domains (Van Nuffelen, 2021). The DCAT-AP standard is regularly

7 https://www.dublincore.org/specifications/dublin-core
8 Representation as text
9 https://json-ld.org
10 https://www.w3.org/TR/rdf-syntax-grammar
11 https://www.w3.org/TR/turtle

updated by a working group and published by the EU Commission. The DCAT-AP standard is currently available in version 2.1[12]. To illustrate the structure, Figure 1 shows an example. Each *Catalogue* is a collection of *Datasets*. In the case of Open Data portals, this can, for instance, be a specific domain or a geographical region. Further, each *Dataset* is a collection of *Distributions*. This can, for instance, be data about one specific topic. DCAT-AP incorporates even more classes besides *Distributions*, *Datasets* and *Catalogues*, but these are not directly relevant for translations.

The DCAT-AP standard divides the individual metadata fields into three categories: mandatory, recommended and optional.

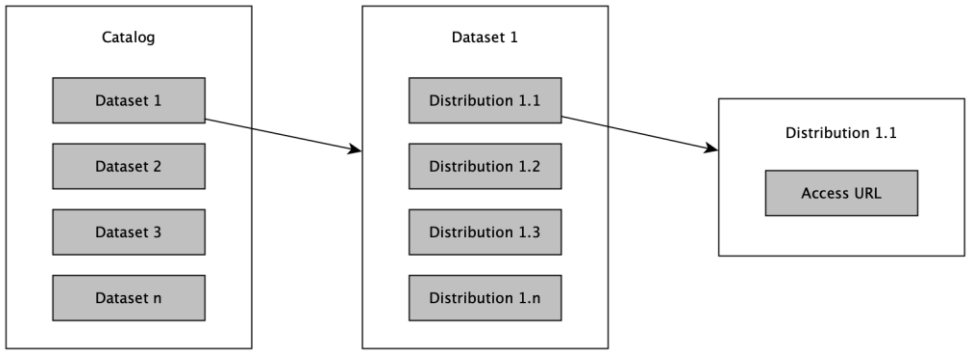


Figure 1: Parts of the knowledge graph (arrows symbolise links)

A dataset contains two mandatory metadata fields, *Title* and *Description*. Both are literals. A literal is a string of characters and/or numerical values that are saved in a human-readable format. Optionally, a language tag can be added to the end of each literal. For a definition, see (Phillips and Davis, 2009). For instance, '"This is a sample literal"@en' is a serialised representation. The text is in inverted commas, and the language tag is indicated with an @ symbol.

In addition to the mandatory metadata fields, *Title* and *Description*, there are seven recommended metadata fields. This includes *Publisher* and associated *Contact Point*, a *Category*, and a temporal and a spatial specification in relation to the data itself (*Temporal Coverage* and *Spatial Coverage*). A list of distributions (*Distributions*) is also a recommended metadata field, which is available if there are files associated with the dataset. Also a recommended metadata property is *Keywords*.

The optional metadata fields are a long list of further properties. This includes another *Language* property. It describes the language of each *Distribution*, i.e. associated file. The file language can be different from the dataset's metadata language. In addition, further optional properties are *access rights, creator, conforms to, documentation, frequency, has version, identifier, is referenced by, is version of, landing page, other identifier, provenance, qualified attribution, qualified relation, related resource, release date, sample, source, spatial resolution, temporal resolution, Type, update/modification date, version, version notes, was generated by*.

[12] https://joinup.ec.europa.eu/collection/semantic-interoperability-community-semic/solution/dcat-application-profile-data-portals-europe/release/211

Distributions have one mandatory metadata property, the *access URL*. The recommended metadata of distributions includes the *Availability* (information about the duration of availability), the *Format* (media type) and the *License* (regulates the rights for further use). Furthermore, it includes the property *Description*, which describe the content of the file or data.

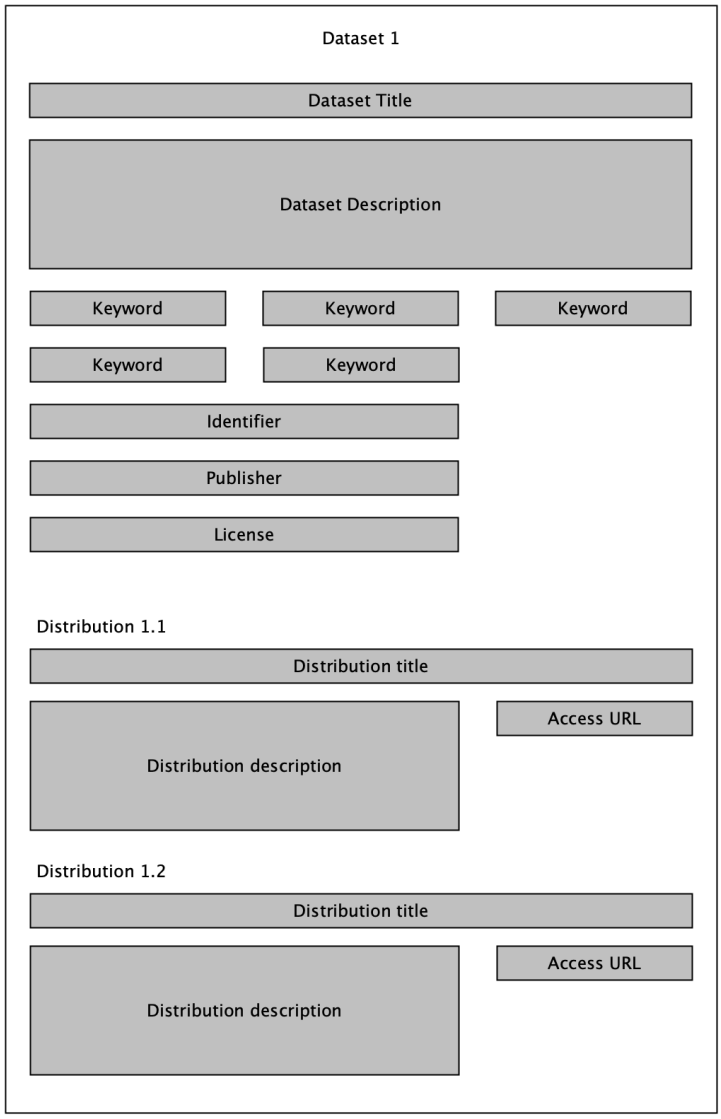
The optional metadata fields are a long list of further properties. It includes the *Title.* Likewise *access service, byte size, Checksum, compression format, Documentation, download URL, has policy, Language, linked schemas, media type, packaging format, release date, Rights, spatial resolution, status, temporal resolution, Title, update / modification date*.

Multilingualism is supported by the DCAT-AP standard. A text-based property can be saved in any number of languages. The *Title* of a dataset, for example, is a mandatory entry and is only added once. However, it is possible to add several translations of this property. For example, '"Hello World"@en, "Ciao mondo"@it, "Bonjour le monde"@fr, "Hola Mundo"@es, "Hallo Welt"@de'. It is not defined which specific language is mandatory.

Figure 2 shows an excerpt of text-based metadata properties that could potentially be translated. Each *dataset* has a *title* and a *description*. In addition, there is a set of *keywords*, an *identifier*, a *publisher* and a *license*. Each *dataset* also consists of at least one *distribution*. A *distribution* has a *title*, a *description* and an *access URL*.

Titles and *descriptions* of *datasets* and *distributions* are translatable. *Keywords* are also potential candidates for translation. Open Data portals also contain other text parts, which could be translated as well. For example, a *license*. However, translations on such text parts are too vulnerable to errors, and license descriptions, in particular, should be done by a professional translator. A *publisher* is a proper name and should not be translated, as well as identification numbers like a unique *identifier* and an *access URL*.

Figure 2: Partially overview about a DCAT-AP dataset and its distributions



Due to its Linked Data character and the possibility to formulate metadata in a standardised way, DCAT-AP is well suited to construct a knowledge graph (Ryen et al., 2022). In theory, it is possible to add one's own metadata properties. But this would be a disadvantage when it comes to interoperability. In its unrestricted form, a knowledge graph can be extended at will (Hogan et al., 2022).

Example 1 shows the structure of a dataset with its distributions in serialised form, according to the Turtle notation. There are some prefixes defined at the beginning to make the document more readable.

Example 1[13]: A dataset in Turtle notation.

```
1  @prefix adms:   <http://www.w3.org/ns/adms#> .
2  @prefix dcat:   <http://www.w3.org/ns/dcat#> .
3  @prefix dct:    <http://purl.org/dc/terms/> .
4  @prefix dqv:    <http://www.w3.org/ns/dqv#> .
5  @prefix edp:    <https://europeandataportal.eu/voc#> .
6  @prefix foaf:   <http://xmlns.com/foaf/0.1/> .
7  @prefix schema: <http://schema.org/> .
8  @prefix skos:   <http://www.w3.org/2004/02/skos/core#> .
9  @prefix spdx:   <https://spdx.org/rdf/terms/#> .
10 @prefix vcard:  <http://www.w3.org/2006/vcard/ns#> .
11
12 <http://data.europa.eu/88u/dataset/horizon-2020-environment-and-resources>
13         a                  dcat:Dataset ;
14         dct:title          "Horizon 2020 Environment and resources"@en ;
15         dct:description    "This data set covers projects funded under the Horizon 2020 Challenge Climate action,
16                             environment, resource efficiency and raw materials. \r\n\r\nActivities in this Challenge will help
17                             increase European competitiveness, raw materials security and improve wellbeing.
18                             At the same time, they will assure environmental integrity, resilience and
19                             sustainability with the aim of keeping average global warming below 2Â° C and enabling
20                             ecosystems and society to adapt to climate change and other environmental changes.\r\n\r\n"@en ;
21         dcat:keyword       "environment"@en ,
22                             "cultural heritage"@en ,
23                             "climate action"@en ,
24                             "raw material"@en ,
25                             "Earth observation"@en ,
26                             "Horizon 2020"@en ,
27                             "Research and innovation"@en ,
28                             "H2020"@en ,
29                             "circular economy"@en ;
30         dct:publisher      <http://publications.europa.eu/resource/authority/corporate-body/CINEA> ;
31         dcat:distribution  <http://data.europa.eu/88u/distribution/ea389e18-c6db-4ee0-bc06-55011db8681e> ,
32                            <http://data.europa.eu/88u/distribution/a5a10ebf-94c7-4d46-ac99-cf80753e93a1> ;
                               ...
```

Line 12/13 contains the triple that defines a dataset. The URI is a unique identifier (scheme: URI a *dcat:Dataset*). This dataset then includes additional metadata, such as the title and a description (line 14 and 15 in Example 1). The language tags are at the end of each literal, here @en (line 14, 20, and 21-29). They indicate that the texts are in English. The dataset has metadata about the associated distributions under dcat:distributions (line 31 and 32). Two distributions are linked to this dataset.

Example 2: A distribution in Turtle notation.

```
1 <http://data.europa.eu/88u/distribution/ea389e18-c6db-4ee0-bc06-55011db8681e>
2         a                  dcat:Distribution ;
3         dct:title          "Horizon 2020 Societal Challenge 'Climate action, environment, resource efficiency & raw materials' web"@en ;
4         dct:description    "Programme webpage"@en ;
5         dct:identifier     "http://data.europa.eu/88u/distribution/0904d417-d938-49db-8725-bf07323e207f" ;
6         dcat:accessURL     <https://ec.europa.eu/easme/en/horizon-2020-societal-challenge-climate-action-environment-resource-efficiency-raw-materials> ;
7
```

One distribution is shown in Example 2. The structure is similar to Example 1, but lines 1 and 2 say it is a distribution. A *Title* and a *Description* are also present here. Literals are put in quotation marks and have the optional language tag at the end (line 3 and 9).

As far as we know, very few Open Data portals have translations of their metadata into other languages, and if so, they are not managed automatically by a middleware. For instance, the EU

[13] https://data.europa.eu/data/datasets/horizon-2020-environment-and-resources

Open Data portal[14] had translated metadata, but these translations are done manually. Within research data portals, as provided by universities, metadata is usually provided in a native language and/or in English, and translations are done manually by data curators. In Wikidata (Wikidata, 2020), metadata and data are provided in multiple languages. Tens of thousands of volunteers contribute to these translations.

3. Recommendations for the Translation Service

From our long-time experience in developing Open Data portals and from setting up a translation service in a real-world scenario, we have gathered several recommendations when it comes down to setting up a Translation Service.

Translation in all European languages: To enable FAIR access, the Translation Service should be able to translate textual components in as many languages as possible.

Automatic translation of metadata fields containing text: To improve FAIRness, metadata should be translated. This is especially important for titles and descriptions of datasets and distributions, as they describe the file's content.

No (automatic) translation of other metadata fields: All other metadata can be partially translated as well. Some parts should be excluded, to avoid the introduction of errors. Information such as proper names or identification numbers should not be included in the translation process as they could unintentionally be changed. Information such as publication date or license details does not need to be translated, as rendering systems should already be able to provide adequate translations and formats in each language.

Preparation of texts for translation: Numerous exceptional cases should be dealt with in order to avoid the introduction of errors. Language specifications should not be translated. Also markup language should be excluded. These parts should be removed before translation, and should be added again afterwards.

Scaling-up of Translation Service: Handling a huge number of translations is necessary. On the one hand, portals nowadays provide tens of thousands of datasets. On the other hand, many translation engines have an artificial limit to protect the system from overload. For this reason, a Translation Service should have the capability to temporarily store the translation requests and, if necessary, to protect them from being lost in the event of a system failure.

Throughput of Translation Service: Translation requests should be processed at high speed. It is not necessary that everything is translated immediately, but the Translation Service should be able to handle all incoming requests. This requires asynchronous and reactive processing of incoming requests.

[14] The portal does not exist any longer. It is now part of data.europa.eu.

4. Global architecture of the Translation Service

The Translation Service has been implemented in the context of the data management ecosystem Piveau[15], but can be operated standalone as well. Piveau's central idea is to harvest metadata from several data portals and make them available in one place. The use of Piveau in a European context created the need to offer metadata in multiple languages.

The Translation Service is informed of the metadata to be translated via a REST interface. The Translation Service performs the translation process in several steps, as detailed in section 4.2, and makes use of an external translation engine for the actual translation, as described in section 4.1.

4.1. Translation engines

There are various solutions on the market for an automatic translation of texts. The results are better placed in the required context if a translation engine has been trained for a particular domain (Rivera-Trigueros, 2021). There exists a variety of commercial solutions. For example, Google Translate[16], Microsoft Translation[17], or DeepL[18].

These services provide application programming interfaces (APIs), which are helpful when it comes to the automated translation of datasets in Open Data portals. However, this is also where free use ends for these commercial systems, as requests are charged according to the number of words.

Another problem of commercial solutions is the location of the servers. For instantiating an Open Data portal within the EU, under certain conditions, it is not allowed to use a Translation Service that processes and stores its data on US servers.

In addition to commercial offerings, the open-source community has ambitions to develop free and open-source translation engines. One of these projects is LibreTranslate[19]. For the use case of translating a large number of text snippets very quickly, most of these systems reach their limits. The required performance would be associated with additional hardware and staff, generating costs that communities cannot easily afford.

In the context of the Connecting Europe Facility[20], the EU supports a project called eTranslation. This translation engine is not accessible to the public, but to a limited group of governmental institutions. Within the EU, many documents are constantly being translated. Some of these documents do not need to be manually translated into another language by a professional translator. They can also be translated automatically. For this purpose, eTranslation was launched as a project. The service can translate complete document files for users or individual text snippets via a Web application or an API.

15 https://www.piveau.de/en
16 https://translate.google.com/?sl=auto&tl=ar&op=translate
17 https://www.microsoft.com/en-us/translator
18 https://www.deepl.com/en/whydeepl
19 https://libretranslate.com
20 https://ec.europa.eu/inea/en/connecting-europe-facility

For reasons of data transparency, data sovereignty, performance, and costs, the decision was made to use eTranslation together with our Translation Service.

4.2. Data Management Platform

The approach presented in this article is part of a collection of micro services that together make up a data management platform for Open Data called Piveau (Kirstein et al., 2020). Developed specifically for the public sector, Piveau relies on Semantic Web technologies and the use of the DCAT-AP metadata standard.

The Translation Service focuses on the translation of metadata. The original files are excluded from the translation. In Open Data portals that are based on Piveau, the descriptive metadata is collected and presented. The original files remain in the harvested systems and are linked from within the Open Data portal.

As shown in Figure 3, for the Translation Service, Hub and Consus are the most relevant components of Piveau. The Open Data portals 1 to 3 are placeholders for data sources, portals like Open Data Bulgaria[21] or Open Data Island[22]. For an Open Data portal to be included, an API is required. It should follow common standards like REST or OAI-PMH. The Consus component forwards the harvested data from the sources to the Hub component for management and additional checks. The Hub component triggers the Translation Service. The translations themselves are performed by eTranslation. The Virtuoso database stores the metadata and its translations in a knowledge graph representation. The individual components are detailed in the following.

Piveau Consus (see Figure 3) is responsible for harvesting metadata. The harvesting mechanism is time-controlled and queries the metadata from a predefined set of Open Data portals. Broad compatibility for various API types is available, like REST, OAI-PMH[23], and SPARQL.

All newly retrieved metadata is compared with the previously harvested metadata. If a new dataset is found that is not yet in the Piveau system, a new entry is created. The Hub component checks whether a translation should take place.

Piveau Hub (see Figure 3) is the central component for the management of metadata. The Hub component communicates with a Virtuoso database, which stores the metadata in a knowledge graph representation. Virtuoso provides functions for writing, reading, updating and deleting datasets. The created knowledge graph follows the DCAT-AP metadata standard.

The Hub component comprises a check on the status of the translations of a dataset. If there is new or updated metadata describing a dataset, the translation process is triggered, and the Translation Service is informed. The Translation Service extracts text parts from the knowledge graph and pre-processes them, so that they can be sent to eTranslation. Metadata is updated with translations

[21] https://data.egov.bg
[22] https://opingogn.is
[23] https://www.openarchives.org/OAI/openarchivesprotocol.html

after each translation process. Old entries are removed and replaced by the new multilingual text snippets.

Translations do not only occur when a new dataset is harvested. Datasets and files can be updated over time, and the metadata changes. As often none or just a few properties change, translating everything from scratch would not be very efficient. For this reason, a text comparison is made to see which properties have changed. Only these properties are retranslated.

Regarding user-friendliness, a message is displayed to the user that the text has been updated and that a translation is in progress but has not yet been completed. This usability mechanism is based on the translation status of the Hub component. The Translation Service component messages status updates to the Hub component when a translation state is changed. Four states are defined: (1) translation in progress, (2) translation update in progress, (3) translation done, and (4) translation failed. In case of an updated dataset, the user can continue to read the old translation and is informed that the translation is in progress. In case of a new dataset, the user sees the text snippets in the original language, together with a status message, that a translation is in progress. In case of a completed translation, the user is shown a message that it is a machine-translated text. In addition, a reference to the original text is available.

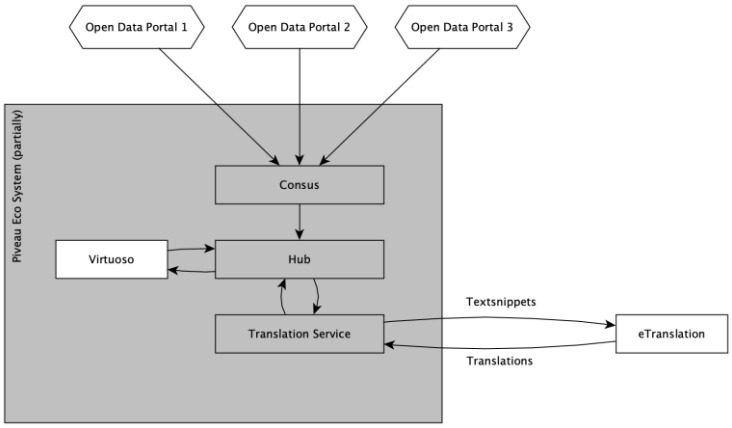


Figure 3: Partial overview of the Piveau components

5. Components of the Translation Service

A micro service architecture has the advantage that the Translation Service is stand-alone. Therefore, it can be used independently of the other Piveau components. For this purpose, the Translation Service offers a REST API for metadata exchange.

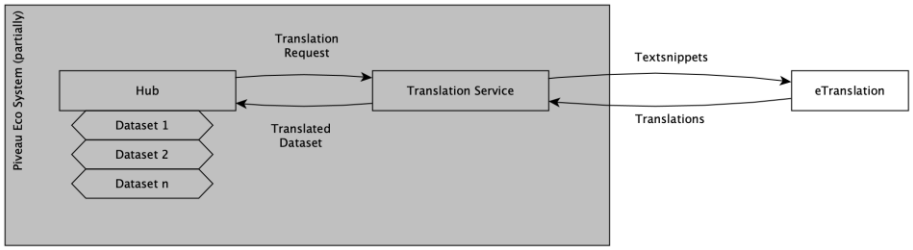


Figure 4: Overview of translation process

Figure 4 shows that there are, essentially, three components involved in the translation process. The Hub component manages the metadata catalogue. The translation process is started if it identifies metadata that needs translation. Therefore, a translation request is sent via the REST API to the Translation Service (see Figure 4).

The metadata to be translated are retrieved from the dataset and all its distributions. Example 3 shows an excerpt for demonstration (*Title* and *Description* from the dataset and its distribution from Examples 1 and 2). They are part of the request to the translation request handler (see Figure 5).

Example 3: Title and description of a dataset and distribution

```
<http://data.europa.eu/88u/dataset/horizon-2020-environment-and-resources>
   a           dcat:Dataset ;
   dct:title         "Horizon 2020 Environment and resources"@en ;
   dct:description   "This data set covers projects funded under the Horizon 2020 Challenge Climate action,
                      environment, resource efficiency and raw materials. \r\n\r\nActivities in this Challenge will help
                      increase European competitiveness, raw materials security and improve wellbeing.
                      At the same time, they will assure environmental integrity, resilience and sustainability with
                      the aim of keeping average global warming below 20° C and enabling ecosystems and society to
                      adapt to climate change and other environmental changes.\r\n\r\n"@en ;


<http://data.europa.eu/88u/distribution/ea389e18-c6db-4ee0-bc06-55011db8681e>
   a           dcat:Distribution ;
   dct:title         "Horizon 2020 Societal Challenge 'Climate action, environment, resource efficiency & raw materials' web"@en ;
   dct:description   "Programme webpage"@en ;

```

In general, the Translation Service pre-processes the dataset for use by eTranslation and sends the text snippets extracted from the metadata properties to the translation engine. Later, eTranslation sends the finished translations back to the Translation Service asynchronously. Depending on the volume and workload, the translation may take some time. The translated text passages are re-integrated into the metadata and into the knowledge graph.

As can be seen in Figure 5, the Translation Service consists of four components, each of which is explained in more detail in the following sections.

The four components have specific tasks. While the Translation Request Handler pre-processes the texts to be translated, the Translation Request Partitioner deals with the requests to eTranslation. All intermediate results are stored in a database to prevent the loss of metadata. The Translation Receiver post-processes the finished translations and reintegrates them into the datasets, and the Translation Allocator reintegrates all texts into the knowledge graph.

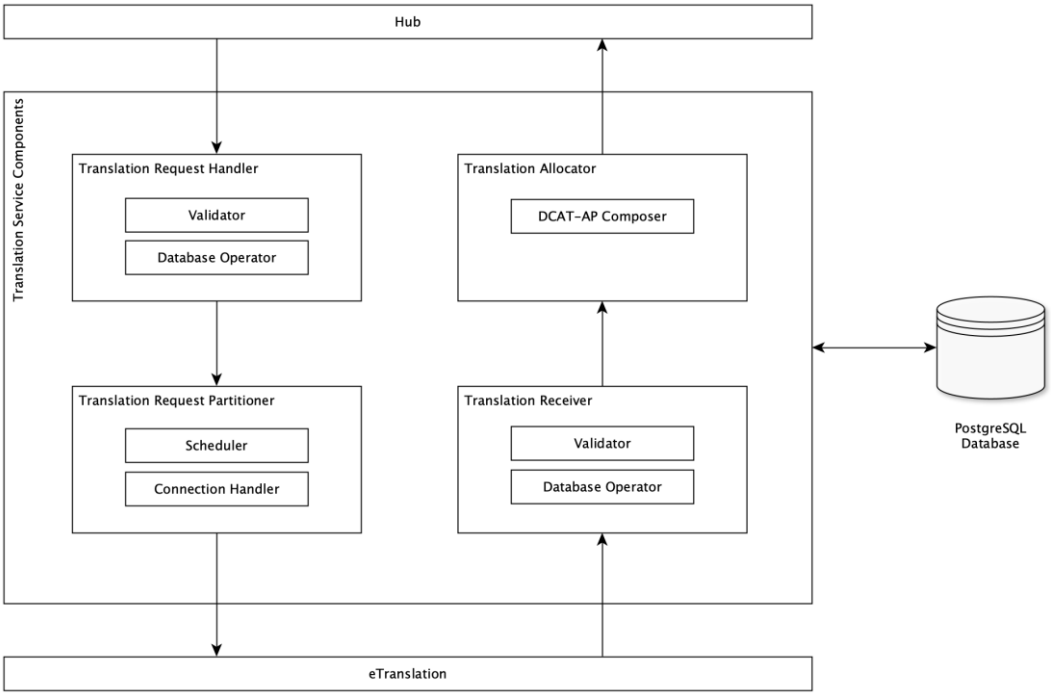


Figure 5: Internal structure of the Translation Service

5.1. Translation Request Handler

As Figure 3 suggests, metadata can be collected from a huge number of sources, with each source often providing hundreds to several thousands of datasets. Each dataset, in turn, consists of several text snippets.

The translation request handler has the central task of accepting translation requests and checking the contents in detail. This first check verifies whether the texts can be translated with a machine translation engine or not. In case a text cannot be translated, the component must reject the request with an adequate error code.

Datasets collected from national Open Data portals are not error-free (Umbrich, J. et al., 2015). In an ideal system, all texts are well formulated in a comprehensible way, do not have any unspecific special characters, are complete, and marked with an accurate language reference.

From a practical point of view, it is known that texts may be missing or written in an incorrectly specified language. The specification of the language is essential for a translation engine, although

solutions exist that can recognise a language based on the text. The shorter a text snippet is, the less likely the language recognition will work correctly[24]. The title of a dataset can be a very short text snippet, consisting of only one word. Automatic identification of the language is not performed for the reason of unreliability, even more so if the titles contain proper names or special terms that only make sense in the context of the data provider.

Instead, a hierarchy of occurring cases has been implemented. The language tag has the highest priority. If specified, the tag can be used as the source language. This does not exclude the possibility that a data provider has made a mistake here. Nevertheless, experience shows that this is correct in most cases. If a language tag is not present, the Translation Service checks whether the property *Language* in the dataset is present. It is an optional property. If it exists, this specified language will be used. If this information is also missing, the property *Language* from the catalogue is used. This information is also not mandatory. English is assumed as the default value if this property is missing from the catalogue.

The ISO 639-1 standard defines two characters that stand for a specific language. It is used in the DCAT-AP standard. Nevertheless, not all data providers follow the ISO 639-1 standard and choose a different one or go beyond the two characters. This requires an additional comparison with a second table that internally includes many other variants of the abbreviations. If a mismatch occurs, a search is done within the table, and, if necessary, the language tag is replaced. If the language tag is unknown, the procedure continues as if the tag did not exist, and exception handling is performed as described above.

There may be text components that are not suitable for being translated, due to markup languages or URLs. While URLs are not a problem unless they are wrapped in the body text and not recognised by the translation engine. Often, text snippets in markup language are provided or errors occur during the harvesting process so that HTML (or XML) markup language is present in the description. This is where a translation engine has more significant problems (Tezcan and Vandeghinste, 2011). However, markups are not necessary to display descriptions and can be deleted. In any case, this must be done in a preprocessing step.

In addition to the text's semantic challenges, the incoming requests must be processed with high performance. The harvesting process of the Piveau Consus component runs very fast due to parallel processing so that several hundreds to thousands of requests reach the Translation Service in a few seconds. The translations cannot simply be passed through due to a technical limitation of the translation engine. They are handled and stored in a database, waiting for being translated. In the context of Piveau, the aim is to create a responsive, resilient and elastic application (Kirstein et al., 2020). In particular, the Vert.x framework[25] meets these requirements and offers a non-blocking and event-based runtime. Moreover, Vert.x is suitable for exchanging a high number of requests between several APIs.

[24] https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/all+etranslation+services?preview=/82772053/378242303/SLA%20eTranslation%20MT_v1.9.pdf

[25] https://vertx.io/

The challenges listed here are addressed in the Translation Request Handler. Depending on the specific case, it applies exception handling or rejects the translation request. This depends on the translation engine used and whether it can handle the corresponding case and deliver a proper translation. The Translation Request Partitioner receives a message for further processing when a translation request is accepted.

5.2. Translation Request Partitioner

The Translation Request Partitioner component manages the requests to eTranslation. Due to the high number of texts that need to be translated every day, it is necessary to send the maximum number of allowed requests without exceeding the limits of the translation engine. As no requests should be rejected, the Partitioner independently regulates when it sends new requests to the translation engine. The process is thus, not immediately triggered by the arrival of new translation requests.

A dataset is broken down into its text components. While the Translation Request Handler has already formally checked and prepared all requests about their translatability, the Translation Request Partitioner must check whether texts are too long. The maximum number of characters is 5000 within eTranslation. The DCAT-AP standard does not provide any limit, so it is possible that there are texts with more than 5000 characters. In this case, two or more separate requests to eTranslation must be prepared. The texts would be divided into several queries, based on complete sentences. A simple extraction, after, for example, 5000 characters would most likely not cover a sentence completely, which would have an impact on the quality of the translation.

The API of eTranslation offers the transmission of an ID. After translation, this ID is sent back unchanged to the Translation Service, including text and language. It can be freely selected and is used by the Translation Receiver and Translation Allocator to assign the translations to the matching dataset. The ID is the same as the identifier of the dataset or the distribution according to the DCAT-AP standard. This makes it possible to assign the translations to the correct datasets without further mapping. In the case described above, where the character limit is exceeded, the ID is provided with an addition (…part1/3, …part2/3, etc.). In this way, it is recognisable that it is a fragmented text and the Translation Service has to wait until the other translations arrive, before they are written back into the knowledge graph.

The translation engine, i.e. eTranslation, expects the requests in a predefined JSON format. A simple forwarding of the RDF triples is impossible, as the engine will probably change the RDF syntax and not produce a syntactically correct RDF document. The Translation Request partitioner prepares the request in JSON format.

As our experience shows from the European Data Portal (Kirstein et al., 2019), title and description of a dataset are often not correctly entered by the data provider. The actual plain text is sometimes an XML or HTML document, sometimes a string of characters resulting from a binary coded file. In the case of XML or HTML documents, a pre-procession step is necessary (described in 5.1). In the case of a binary file, no translation is performed.

The descriptive texts may contain URLs and hashtags. These are copied before translation, saved separately and then added again afterwards. URLs and hashtags can be efficiently identified in a floating text using a regular expression. After translation, they are placed in the correct position again. The new positions can be found in the text through the used regular expressions. Finally, the URLs and hashtags are overwritten with the original snippets.

5.3. Translation Receiver

The Translation Service also provides an interface for the finished translations, as eTranslation works asynchronously. For each language, the Translation Service must receive a separate GET request (as an HTTP method in the form of a parameter within the URL) from eTranslation. On average, each text is translated into 26 languages. Additionally, each dataset consists of at least four texts - *Title* and *Description* of a dataset and *Title* and D*escription* of each distribution. Datasets often have multiple distributions, on average 8 in our usage scenario, and sometimes hundreds, thus creating a vast number of requests (e.g. 416 requests for eight distributions).

The Translation Receiver manages the finished translations. Because the result is sent back by eTranslation only after a specific time (asynchronously), each translated snippet is stored in a database. This also prevents duplicate translations. eTranslation resends the translation in case of any error, even if the previous request has been completed. Depending on the infrastructure, this can be caused by an over engineered firewall, for example.

Parallel processing is essential here, as numerous translations arrive every second. Once a text has been validated and saved, a check is made to see whether the translation request has now been completely processed. If all translated snippets are complete and error-free, the translation allocator is notified.

If, for unknown reasons, not all snippets are available, the translation process is restarted after a specified time.

5.4. Translation Allocator

The Translation Allocator prepares the translated texts for integration into the internal knowledge graph. All text extracts are assigned to the related dataset or distribution and are provided with semantic features, like a new language tag and an allocation to the specific RDF triple. The ID used in the translation request partitioner is used for the assignment. It contains the original ID of the dataset or distribution.

The translations are added to the *Title* or *Description* at the appropriate position in the knowledge graph. The original text is already there, and different texts in other languages can be added using the RDF standard. The triple and the knowledge graph structure are not changed.

Information about the languages and whether it is a manual or machine translation is also added. According to the DCAT-AP standard, a text to be translated is literal and has an optional language

tag. However, it makes sense to append this, so that the language tag can be used to determine the language. This simplifies further use of the texts for a wide variety of applications.

The XML data type "lang" specifies the language tag. "This is a sample text." is expanded by the language tag "@en". (i.e. Example 3 line 3, 9, 14 or 15). When a text is machine translated, the tag is extended by: 'en-t-de-t0-etranslation'. The first language code stands for the original language, followed by a 't' for the translation from the original language (in this example: de). The 't0' represents a machine translation, and the 'etranslation' stands for the engine used (Davis et al., 2012).

Example 4 shows the reintegrated translations as an RDF triple in the Turtle notation. Compared to Example 3, only one metadata property (*Title* or *Description*) is shown in each case because otherwise, it would become very lengthy. However, it is clear that the translations are attached in the same triple and can be distinguished from each other by the new language tag. The lines 3 to 26 of Example 4 show translations, including an extended language tag, except for line 7, which is the original *Title* of the dataset. The same applies for the distribution's description in the lines 30 to 53. Line 32 contains the original *Description* of this distribution.

Example 4: Reintegrated translation to triple

```
1  <http://data.europa.eu/88u/dataset/horizon-2020-environment-and-resources>
2          a                   dcat:Dataset ;
3          dct:title           "Horizont 2020 Životné prostredie a zdroje"@sk-t-en-t0-mtec ,
4                              "Programa „Horizontas 2020“ Aplinka ir ištekliai"@lt-t-en-t0-mtec ,
5                              "Orizzont 2020 Ambjent u riżorsi"@mt-t-en-t0-mtec ,
6                              "Horizont 2020 Umwelt und Ressourcen"@de-t-en-t0-mtec ,
7                              "Horizon 2020 Environment and resources"@en ,
8                              "Fís 2020 Comhshaol agus acmhainní"@ga-t-en-t0-mtec ,
9                              "“Apvārsnis 2020” vide un resursi"@lv-t-en-t0-mtec ,
10                             "Horisont 2020 Miljø og ressourcer"@da-t-en-t0-mtec ,
11                             "Obzorje 2020 Okolje in viri"@sl-t-en-t0-mtec ,
12                             "Obzor 2020. Okoliš i resursi"@hr-t-en-t0-mtec ,
13                             "Horizon 2020 Environnement et ressources"@fr-t-en-t0-mtec ,
14                             "Orizont 2020 Mediu şi resurse"@ro-t-en-t0-mtec ,
15                             "Horizonte 2020 Ambiente e recursos"@pt-t-en-t0-mtec ,
16                             "Horizont 2020 Környezetvédelem és erőforrások"@hu-t-en-t0-mtec ,
17                             "Horizont 2020 Životní prostředí a zdroje"@cs-t-en-t0-mtec ,
18                             "Horisontti 2020 Ympäristö ja resurssit"@fi-t-en-t0-mtec ,
19                             "Programm „Horisont 2020“Keskkond ja ressursid"@et-t-en-t0-mtec ,
20                             "Horisont 2020 Miljö och resurser"@sv-t-en-t0-mtec ,
21                             "Orizzonte 2020 Ambiente e risorse"@it-t-en-t0-mtec ,
22                             "„Хоризонт 2020“ Околна среда и ресурси"@bg-t-en-t0-mtec ,
23                             "Horizon 2020 Milieu en middelen"@nl-t-en-t0-mtec ,
24                             "Horizonte 2020 Medio ambiente y recursos"@es-t-en-t0-mtec ,
25                             "„Horyzont 2020” Środowisko i zasoby"@pl-t-en-t0-mtec ,
26                             "«Ορίζων 2020» Περιβάλλον και πόροι"@el-t-en-t0-mtec ;
27
28 <http://data.europa.eu/88u/distribution/ea389e18-c6db-4ee0-bc06-55011db8681e>
29         a                   dcat:Distribution ;
30         dct:description     "Página Web do programa"@pt-t-en-t0-mtec ,
31                             "Programos tinklalapis"@lt-t-en-t0-mtec ,
32                             "Programme webpage"@en ,
33                             "Página web del programa"@es-t-en-t0-mtec ,
34                             "Pagina web del programma"@it-t-en-t0-mtec ,
35                             "Programmas tīmekļa vietne"@lv-t-en-t0-mtec ,
36                             "Programmi veebileht"@et-t-en-t0-mtec ,
37                             "Pagina web a programului"@ro-t-en-t0-mtec ,
38                             "Leathanach gréasáin an Chláir"@ga-t-en-t0-mtec ,
39                             "Website des Programms"@de-t-en-t0-mtec ,
40                             "Webová stránka programu"@sk-t-en-t0-mtec ,
41                             "Ohjelman verkkosivut"@fi-t-en-t0-mtec ,
42                             "Page web du programme"@fr-t-en-t0-mtec ,
43                             "Уебстраница на програмата"@bg-t-en-t0-mtec ,
44                             "Ιστοσελίδα του προγράμματος"@el-t-en-t0-mtec ,
45                             "Webpagina van het programma"@nl-t-en-t0-mtec ,
46                             "Strona internetowa programu"@pl-t-en-t0-mtec ,
47                             "Paġna web tal-programm"@mt-t-en-t0-mtec ,
48                             "Programmets hjemmeside"@da-t-en-t0-mtec ,
49                             "Web-stranica programa"@hr-t-en-t0-mtec ,
50                             "Spletna stran programa"@sl-t-en-t0-mtec ,
51                             "Internetové stránky programu"@cs-t-en-t0-mtec ,
52                             "A program weboldala"@hu-t-en-t0-mtec ,
53                             "Programmets webbplats"@sv-t-en-t0-mtec ;
54
```


6. Use Case: European Data Portal (EDP) & Data Europa EU

Each EU member has one or more national Open Data portals, often separated by topic, such as geographic or administrative data. Due to the historical development of Europe, different languages are spoken in these countries. Nevertheless, the EU is trying to foster trans-European exchange and minimise language barriers.

This requirement also exists for the European Data Portal (EDP)[26] (EDP, 2019). Its central task is to make public sector information available as Open Data. It currently provides approximately 1.4 million datasets from 36 European states, harvested from 81 national Open Data portals. These national portals mostly contain datasets in English or the national language. For the goal of being an inclusive portal, all datasets need to be translated, possibly into all other European languages - together with the use of semantic web technologies (Ibáñez et al., 2019).

In August 2019, we successfully integrated our Translation Service into the EDP. It meets the formulated recommendations, and apart from some minor challenges, it has been fully functioning since then.

Due to the daily harvesting of national Open Data portals, there are a lot of datasets that need to be translated every day within the EDP. This is because titles or descriptions change or because new distributions or even entire datasets are added. Usually, every day, between 1,000 and 130,000 new datasets, and respectively, translation requests are sent to the Translation Service. So, the Translation Service must produce the translations in a timely manner as a requirement.

At some point, the metadata has been integrated into a national Open Data portal by either humans or machines. Thus, the input is very heterogeneous and does not always follow the rules defined by DCAT-AP. This is a significant challenge and requires metadata pre-processing.

Table 1: Statics about provided metadata

Datasets	**Distributions**	**Provided dataset titles**	**Provided dataset descriptions**	**Provided distribution titles**	**Provided distribution descriptions**
1,441,598	2,591,933	1,510,096	N/A	1,501,291	597,291

As of today[27], the EDP manages 1,441,598 datasets in its internal knowledge graph. These datasets contain a total of 2,591,933 distributions. This corresponds to 1.8 distributions per dataset. As a rule, each dataset and distribution should contain a title and description. As shown in Table 1, there are 1,510,096 titles supplied by the providers. Sometimes translations are also provided, so the number is slightly higher than the number of datasets. A similar number is available for the titles of the

[26] data.europa.eu
[27] 14th April 2022

distributions. However, about one million distributions were provided without a title. For the descriptions, the number is even lower at 597,291; about two million distribution descriptions are not available.

Table 2: Missing language tags and empty texts

	Empty string	**Missing language tag**
Dataset title	304	203,269
Dataset description	6,711	N/A
Distribution title	1,432	810,634
Distribution description	189,282	600,143

In addition to the metadata that is not provided at all, one must also consider empty character strings. A description would be available as metadata but not usable with its empty string. There are 304 titles and 6,711 descriptions in datasets that contain an empty string (see Table 2). For distributions, the quality is even worse. There are 1,432 empty titles and 189,282 empty descriptions. Another problem is the lack of language tags. 203,269 datasets require additional effort to translate because the language tag was not provided. The distributions are also missing many language tags. 810,634 titles have no language tag, and 600,142 descriptions are present without a language tag. It can be seen that the quality of the distributions is significantly worse and that the language must be determined in other ways.

Table 3: Average character length of title and description including maximum value and the number of metadata which exceeds eTranslations character limitation

	Average character length	**Maximum character length**	**Number of texts with length > 5000**
Dataset title	67	8,666	5
Dataset description	574	2,670,299	63,010
Distribution title	36	12,484	6
Distribution description	152	49,619	11,685

The average length of existing dataset titles is 67 characters per title, as shown in Table 3. However, the longest title is 8,666 characters long, and 5 dataset titles exceed 5,000 characters. The descriptions are, on average, 574 characters long, with the most extended description being 2,670,299

characters long. 63,010 dataset descriptions exceed 5,000 characters. The situation is similar on the distribution side. A title is, on average, 36 characters long and an associated description 152 characters. The longest title is 12,484 characters, and the longest description is 49,619 characters. There are 6 titles from distributions that exceed 5,000 characters, and 11,685 descriptions that are longer than 5,000 characters. When computing the average character length, texts with empty strings have been excluded.

There are 46,704,562 translated titles in the EDP. The number is high because each title is usually translated into 26 languages. The data provider can supply several translations. These are not re-translated and will remain in the data portal.

Table 4: Number of text snippets which contain different special character sequences

	Dataset title	**Dataset description**	**Distribution title**	**Distribution description**
HTML / XML	136	30,444	91	51,148
Markdown	255	14,425	87	6,151
Hashtags	4,337	91,981	10,057	21,153
URLs	627	733,048	86,349	169,987

Table 4 shows an overview of all text snippets that contain special characters or other markup elements. No difference was made between HTML and XML, but both markup languages have the unique feature of using tags for structuring. The DCAT-AP standard does not prohibit it, but it is not suitable for every representation and every translation engine. Markdown symbols does not cause translation problems because special characters are usually translated exactly as they are in the original. The situation is similar with hashtags; overall, 2 to 5 percent of all datasets are affected. However, it is noticeable that almost exactly half of all datasets include one or more URLs in their description texts.

7. Discussion & Conclusion

The Translation Service presented in this paper has been developed to translate large amounts of metadata automatically. It is designed for Open Data portals based on Semantic Web technologies and the metadata standard DCAT-AP. Thanks to the integration of the translations into a knowledge graph representation, new functionalities are enabled.

The Translation Service can be used out-of-the-box. The component works independently of specific data portals and offers APIs for metadata transfer, as well as a configuration file for the use of a translation engine. The service can also be used in a container environment, such as Docker, which

minimises the administration effort and improves scalability. The source code has been published as part of Piveau[28].

There is no dependency on particular languages, so that the Translation Service can be used neutrally for all languages. However, the encoding of all language symbols is limited to the UTF-8 standard.

It is generally challenging to balance translation speed with a high number of translation requests. For this purpose, the Translation Service uses a database and a queue mechanism. In this way, any number of translation requests can be sent to the Translation Service without exceeding the limitations of translation engines.

The permanent storing of translations and their incorporation in a knowledge graph representation enables new features and functions in a portal. Apart from a multilingual search, the development of new services is enabled.

Recognition of the existing language is currently only rudimentary. This requires trust in the use of a correct language tag by data providers. The availability of more reliable methods is up to future work.

The translation of data itself is up to future work. Here one could start with plain texts and proceed with more complex data later on.

References

Alasem, A. (2009) 'An Overview of e-Government Metadata Standards and Initiatives based on Dublin Core', Electronic Journal of eGovernment, 7(1), pp. 1–10.

Albertoni, R. et al. (2020) 'Data Catalog Vocabulary (DCAT) - Version 2', 4 February. Available at: https://www.w3.org/TR/2020/REC-vocab-dcat-2-20200204/ (Accessed: 21 March 2021).

Bahdanau, D., Cho, K. and Bengio, Y. (2016) 'Neural Machine Translation by Jointly Learning to Align and Translate', arXiv:1409.0473 [cs, stat] [Preprint]. Available at: http://arxiv.org/abs/1409.0473 (Accessed: 21 March 2022).

Carroll, S.R. et al. (2020) 'The CARE Principles for Indigenous Data Governance', Data Science Journal, 19, p. 43. Available at: https://doi:10.5334/dsj-2020-043.

Colpaert, P. et al. (2013) 'The 5 Stars of Open Data Portals', in Proceedings of MeTTeG(2013(7)), pp. 61–67.

Davis, M. et al. (2012) 'BCP 47 Extension T - Transformed Content (RFC 6497)', February. Available at: https://datatracker.ietf.org/doc/html/rfc6497 (Accessed: 19 March 2021).

EDP (2019) 'European Data Portal - Portal Version 4.3'. Available at: https://data.europa.eu/sites/default/files/edp_s1_man_portal-version_4.3-user-manual_v1.0.pdf (Accessed: 22 March 2022).

[28] https://github.com/orgs/piveau-data/repositories

Foti, M. (2012) 'MT@EC: working with translators', in Proceedings of Translating and the Computer 34. London, UK: Aslib. Available at: https://aclanthology.org/2012.tc-1.13.

Hayes, P. and Patel-Schneider, P. (2014) 'RDF 1.1 Semantics'. W3C Recommendation. Available at: https://www.w3.org/TR/rdf11-mt/ (Accessed: 17 March 2022).

Higman, R., Bangert, D. and Jones, S. (2019) 'Three camps, one destination: the intersections of research data management, FAIR and Open', Insights the UKSG journal, 32, p. 18. Available at: https://doi.org/10.1629/uksg.468.

Hogan, A. et al. (2022) 'Knowledge Graphs', ACM Computing Surveys, 54(4), pp. 1–37. Available at: https://doi.org/10.1145/3447772.

Ibáñez, L.-D. et al. (2019) 'An Assessment of Adoption and Quality of Linked Data in European Open Government Data', in Ghidini, C. et al. (eds) in Proceedings of The Semantic Web (ISWC 2019). Cham: Springer International Publishing (Lecture Notes in Computer Science), pp. 436–453. Available at: https://doi:10.1007/978-3-030-30796-7_27.

Janssen, M., Charalabidis, Y. and Zuiderwijk, A. (2012) 'Benefits, Adoption Barriers and Myths of Open Data and Open Government', Information Systems Management, 29(4), pp. 258–268. Available at: https://doi.org/10.1080/10580530.2012.716740.

Kučera, J., Chlapek, D. and Nečaský, M. (2013) 'Open Government Data Catalogs: Current Approaches and Quality Perspective', in A. Kő et al. (eds) Technology-Enabled Innovation for Democracy, Government and Governance. Berlin, Heidelberg: Springer Berlin Heidelberg (Lecture Notes in Computer Science), pp. 152–166. Available at: https://doi.org/10.1007/978-3-642-40160-2_13.

Kirstein, F. et al. (2019) 'Linked Data in the European Data Portal: A Comprehensive Platform for Applying DCAT-AP', in I. Lindgren et al. (eds) Electronic Government. Cham: Springer International Publishing (Lecture Notes in Computer Science), pp. 192–204. Available at: https://doi.org/10.1007/978-3-030-27325-5_15.

Kirstein, F. et al. (2020) 'Piveau: A Large-Scale Open Data Management Platform Based on Semantic Web Technologies', in Harth, A. et al. (eds) The Semantic Web. Cham: Springer International Publishing (Lecture Notes in Computer Science), pp. 648–664. Available at: https://doi:10.1007/978-3-030-49461-2_38.

Lnenicka, M. and Nikiforova, A. (2021) 'Transparency-by-design: What is the role of open data portals?', Telematics and Informatics, 61, p. 101605. Available at: https://doi:10.1016/j.tele.2021.101605.

Mons, B. et al. (2017) 'Cloudy, increasingly FAIR; revisiting the FAIR Data guiding principles for the European Open Science Cloud', Information Services & Use, 37(1), pp. 49–56. Available at: https://doi.org/10.3233/ISU-170824.

Nurminen, M. and Koponen, M. (2020) 'Machine translation and fair access to information', Translation Spaces, 9(1), pp. 150–169. Available at: https://doi:10.1075/ts.00025.nur.

Open Knowledge Foundation (2016) Open Definition 2.1 - Open Definition - Defining Open in Open Data, Open Content and Open Knowledge (no date). Available at: http://opendefinition.org/od/2.1/en/ (Accessed: 28 October 2022).

Petychakis, M. et al. (2014) 'A State-of-the-Art Analysis of the Current Public Data Landscape from a Functional, Semantic and Technical Perspective', Journal of theoretical and applied electronic commerce research, 9(2), pp. 7–8. Available at: https://doi.org/10.4067/S0718-18762014000200004.

Phillips, A. and Davis, M. (2009) 'Tags for Identifying Languages (RFC 5646)', September. Available at: https://www.rfc-editor.org/rfc/bcp/bcp47.txt (Accessed: 19 March 2021).

Rivera-Trigueros, I. (2021) 'Machine translation systems and quality assessment: a systematic review', Language Resources and Evaluation, 56(2), pp. 593–619. Available at: https://doi.org/10.1007/s10579-021-09537-5.

Ryen, V., Soylu, A. and Roman, D. (2022) 'Building Semantic Knowledge Graphs from (Semi-)Structured Data: A Review', Future Internet, 14(5), p. 129. Available at: https://doi.org/10.3390/fi14050129.

Schmalz, A. (2019) 'Maschinelle Übersetzung', in Wittpahl, V. (ed.) Künstliche Intelligenz. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 194–211. Available at: https://doi:10.1007/978-3-662-58042-4_12.

Tezcan, A. and Vandeghinste, V. (2011) 'SMT-CAT integration in a technical domain: handling XML markup using pre & post-processing methods', in Proceedings of the 15th Conference of the European Association for Machine Translation, pp. 55–62.

Umbrich, J., Neumaier, S. and Polleres, A. (2015) 'Quality Assessment and Evolution of Open Data Portals', in Proceedings of 2015 3rd International Conference on Future Internet of Things and Cloud (FiCloud), Rome, Italy: IEEE, pp. 404–411. Available at: https://doi.org/10.1109/FiCloud.2015.82.

van der Waal, S. et al. (2014) 'Lifting Open Data Portals to the Data Web', in Auer, S., Bryl, V., and Tramp, S. (eds) Linked Open Data -- Creating Knowledge Out of Interlinked Data. Cham: Springer International Publishing (Lecture Notes in Computer Science), pp. 175–195. Available at: https://doi:10.1007/978-3-319-09846-3_9.

Van Nuffelen, B. (2021) 'DCAT Application Profile for data portals in Europe Version 2.1.0'. Available at: https://joinup.ec.europa.eu/collection/semantic-interoperability-community-semic/solution/dcat-application-profile-data-portals-europe/releases (Accessed: 17 March 2022).

Weibel, S.L. and Koch, T. (2000) 'The Dublin Core Metadata Initiative - Mission, Current Activities, and Future Directions', D-Lib Magazine, 6(12). Available at: http://mirror.dlib.org/dlib/december00/weibel/12weibel.html (Accessed: 16 March 2020).

Wikidata (2020) 'Help:Multilingual', Wikidata, 10 August. Available at: https://www.wikidata.org/wiki/Help:Multilingual (Accessed: 19 March 2021).

Wilkinson, M.D. et al. (2016) 'The FAIR Guiding Principles for scientific data management and stewardship', Scientific Data, 3(1), p. 160018. doi:10.1038/sdata.2016.18.

Zuiderwijk, A. et al. (2015) 'Open Data Disclosure and Use: Critical Factors From A Case Study', p. 12.

About the Authors

Sebastian Urbanek

Sebastian Urbanek is a researcher at the Berliner Hochschule für Technik in the Department of Construction Engineering and Geoinformation. As a PhD candidate of the Institute for Geography at the Otto-Friedrich-

University Bamberg, he is developing a visualization of lost places for use by immersive technologies. In his scientific work, he focuses on data management, data standards, and data visualization, and also on computer graphics and computer vision. He worked at the Fraunhofer Institute for Open Communication Systems in the scope of Open Data before and supported the development of the European Data Portal. He studied computer science with a focus on media at the Berliner Hochschule für Technik.

Sonja Schimmler

Sonja Schimmler is research group lead at Fraunhofer FOKUS. She is also an associated researcher at Technical University of Berlin. In her research, she focuses on the digitalisation and opening up of science and puts a special emphasis on research data infrastructures. Her research interests range from semantic web and linked data over data science and artificial intelligence to software engineering and human-centered computing. She holds a Ph.D. in Computer Science from the University of the Federal Armed Forces Munich. She studied Computer Science at the Technical University of Munich and at the Georgia Institute of Technology (USA).

