DELIVERABLE D3.2

Summary report on user requirements and techniques for data transformation, quality assessment, cleansing, data integration and intended data consumption of the selected datasets

<table>
<thead>
<tr>
<th>Project</th>
<th>Components Supporting the Open Data Exploitation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acronym</td>
<td>COMSODE</td>
</tr>
<tr>
<td>Contract Number</td>
<td>FP7-ICT-611358</td>
</tr>
<tr>
<td>Start date of the project</td>
<td>1st October 2013</td>
</tr>
<tr>
<td>Duration</td>
<td>24 months, until 31st September 2015</td>
</tr>
</tbody>
</table>

<p>| Date of preparation      | 29.7.2014                                        |
| Author(s)                | Martin Nečaský, Tomáš Knap, Jakub Klímek, Peter Hanečák, Anisa Rula |
| Deliverable responsible person | Martin Nečaský                                   |
| Email                    | <a href="mailto:necasky@ksi.mff.cuni.cz">necasky@ksi.mff.cuni.cz</a>                          |
| Reviewed by              | Andrea Maurino and Miroslav Konečný              |
| Status of the Document   | Final                                            |
| Version                  | 1.0                                              |
| Dissemination level      | PU (Public)                                      |</p>
<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Description</th>
<th>Revised by</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>2014-04-01</td>
<td>Basic structure of the spreadsheet for collecting requirements established.</td>
<td>Martin Nečaský, Jakub Klímek</td>
</tr>
<tr>
<td>0.2</td>
<td>2014-05-06</td>
<td>Deliverable document created and basic outline established</td>
<td>Jakub Klímek</td>
</tr>
<tr>
<td>0.3</td>
<td>2014-06-16</td>
<td>First version of requirements for data transformation and linking prepared.</td>
<td>Martin Nečaský, Jakub Klímek</td>
</tr>
<tr>
<td>0.4</td>
<td>2014-06-20</td>
<td>First version of requirements for data quality assessment prepared.</td>
<td>Anisa Rula, Martin Nečaský, Tomáš Knap</td>
</tr>
<tr>
<td>0.5</td>
<td>2014-06-27</td>
<td>First version of requirements for data consumption prepared.</td>
<td>Tomáš Knap, Jakub Klímek, Peter Hanečák</td>
</tr>
<tr>
<td>0.6</td>
<td>2014-07-09</td>
<td>Requirements revised and techniques identified.</td>
<td>All</td>
</tr>
<tr>
<td>0.7</td>
<td>2014-07-18</td>
<td>First version of the deliverable text ready</td>
<td>Anisa Rula, Jakub Klímek, Tomáš Knap</td>
</tr>
<tr>
<td>0.8</td>
<td>2014-07-22</td>
<td>Requirement statistics added</td>
<td>Anisa Rula, Jakub Klímek, Tomáš Knap</td>
</tr>
<tr>
<td>0.9</td>
<td>2014-07-24</td>
<td>Final version of deliverable - first draft</td>
<td>All</td>
</tr>
<tr>
<td>0.10</td>
<td>2014-07-28</td>
<td>Final version of deliverable - second draft</td>
<td>Martin Nečaský</td>
</tr>
<tr>
<td>1.0</td>
<td>2014-07-29</td>
<td>Deliverable finished</td>
<td>Miroslav Konečný, Andrea Maurino</td>
</tr>
</tbody>
</table>
Table of Contents

1. Executive summary ........................................................................................................4
2. Deliverable context ........................................................................................................5
   2.1. Purpose of deliverable ..........................................................................................5
   2.2. Related Documents .............................................................................................5
   2.3. Methodology used ...............................................................................................6
   2.3.1. Methodology ..................................................................................................6
   2.3.2. Datasets .........................................................................................................7
   2.3.3. Partner contributions ......................................................................................8
3. Requirements and techniques for data transformation of the dataset and its integration with other datasets ........................................................................................................9
   3.1. Requirements on techniques independent of a specific dataset ................................10
       3.1.1. DPU logging ..............................................................................................10
       3.1.2. Basic RDF representation .........................................................................10
       3.1.3. Handle large datasets ..............................................................................10
   3.2. Requirements on highly reused techniques used in processing of multiple datasets .........................................................................................................................10
       3.2.1. Extract .......................................................................................................10
       3.2.2. Transform ..................................................................................................11
       3.2.3. Load ..........................................................................................................11
   3.3. Requirements on techniques specific to a single dataset or a small number of related datasets ..................................................................................................................14
       3.3.1. Extract .......................................................................................................14
       3.3.2. Transform ..................................................................................................14
   3.4. Overview of statistics of requirements and techniques for data transformation ....15
4. Requirements and techniques for the quality assessment and cleansing of the dataset .................................................................................................................................20
   4.1. Requirements and techniques for quality assessment ..........................................20
       4.1.1. Normalized scores for quality metrics .........................................................20
       4.1.2. Metadata Quality Requirements and Techniques .......................................21
       4.1.3. Data Quality Requirements and Techniques ...............................................21
   4.2. Requirements and techniques for data cleaning ....................................................26
   4.3. Overview of statistics of requirements and techniques for data quality assessment ..............................................................................................................................27
5. Requirements for consuming curated datasets .............................................................29
   5.1. Overview of statistics ..........................................................................................29
6. Conclusions ....................................................................................................................30
List of figures and tables ....................................................................................................31
References ..........................................................................................................................32

ANNEX 1: Requirements and Techniques (xlsx)

ANNEX 2: Metadata (xlsx)
1. Executive summary

The purpose of this deliverable is to summarize user requirements for data transformation, quality assessment, cleansing, data integration and intended data consumption of the datasets selected for publication in the COMSODE project previously selected in D3.1.

Another goal is to identify necessary techniques for fulfillment of requirements allowing datasets to be correctly published as (linked) open data with the emphasis on data quality and comfortable ways of data consumption by users (application developers). The document also presents how the techniques are going to be implemented by Open Data Node (ODN) developers by data processing units (DPUs) in ODN.

The result of this deliverable is a set of requirements related to:

(i) data transformation and linking

(ii) data quality and cleansing

(iii) data consumption

A detailed list of all requirements is provided in Attachment A of this deliverable. In total, we identified 1794 requirements on how the datasets should be transformed, aligned with existing ontologies and linked on each other and with some external datasets. Moreover, we identified 1345 requirements related to data quality and cleansing. Data quality requirements identify necessary quality metrics which will be measured for particular datasets. Finally, we proposed for each dataset how it should be made accessible to the users of the datasets (mainly application developers) - if it will be published as 5* data (i.e. Linked Open Data available for download or through API - SPARQL endpoints) or only as 3* data (i.e. CSV files available for download or through a tabular API). As the result, we identified 161 (out of 178) datasets which will be published as 5* data by the COMSODE project.

Each set of requirements is summarized in a respective chapter. Chapter 3 is related to transformation and linking requirements, Chapter 4 is related to data quality and cleansing requirements and Chapter 5 is related to data consumption requirements. Chapters summarize the work done in respective tasks (T3.2, T3.3 and T3.4). Besides the summarization of the requirements, each chapter also describes data processing techniques identified on the base of the analysis of the requirements. It provides a description of how the technique should be implemented as a DPU in ODN.

It also serves as an input for the ODN development in WP4. The input for the ODN developers will be a list of requirements for each DPU identified in this deliverable. Any developer of a particular DPU gets information about the technique that needs to be implemented and the requirements on those techniques that specify how the technique is applied in the process of publication of each related dataset.

As the result of the analysis of the dataset, we prepared metadata for each dataset that will be later used by the COMSODE project to create a data catalogue of COMSODE datasets. Prepared metadata are presented in Attachment B of this deliverable.
2. Deliverable context

2.1. Purpose of deliverable

This deliverable represents the summary report on user requirements and techniques for data transformation, quality assessment, cleansing, data integration and intended data consumption of the datasets selected for publication in the COMSODE project. These datasets were described in the deliverable D3.1. This deliverable summarizes outputs of the Tasks 3.2, 3.3 and 3.4 of COMSODE project.

Deliverable objectives are:

1. Describe requirements and required techniques for the data transformation and integration (linking).
2. Describe requirements and required techniques for the quality assessment and cleansing.
3. Describe requirements and required techniques for the intended data consumption.

The outputs of this deliverable will serve as inputs for developers of individual DPUs (Data Processing Units) which will implement techniques identified by the requirements in the ODN. The goal is to identify a number of highly reusable techniques which will be implemented in the ODN as so called data processing units (DPUs). It will be possible to reuse a DPU for various datasets which require the use of the technique implemented by the DPU. The outputs to this deliverable are summarized in this document. The detailed list of requirements and DPUs is in the Attachment A. It is a spreadsheet which contains the following sheets:

- “Datasets” contains the list of datasets analyzed in D3.2. There are the datasets from D3.1 and few additional datasets which were identified as useful during the works on D3.2.
- “Transformation and Linking Requirements” contains the list of requirements identified in Task 3.2. For more details, see Chapter 3 of this deliverable.
- “Data Quality Requirements” contains the list of requirements identified in Task 3.3. For more details, see Chapter 4 of this deliverable.
- “Data Consumption Requirements” contains the list of requirements identified in Task 3.4. For more details, see Chapter 5 of this deliverable.
- “Techniques - DPUs” contains the list of identified techniques and data processing units which will implement them in ODN. For more details, see respective Chapters (3-5) of this deliverable.
- “Techniques - Quality Metrics” contains the list of identified quality metrics which we will use to measure the quality of the published datasets. For more details, see Chapter 4 of this deliverable.
- “Target Ontologies” contains the list of ontologies the published datasets will be aligned with. For more details, see Chapter 3 of this deliverable.

The developers will gather the requirements on each DPU identified in this deliverable and implement it in WP4. The requirements identify the standards and techniques that need to be supported by ODN (e.g., XSLT transformation) as well as types of input and output data storage types (e.g. relational database, FTP storage). This includes requirements on configuration possibilities of individual DPUs (e.g. possibility to specify and use a username and password). The specific configuration of a DPU for usage in
processing of a specific dataset (e.g. a specific XSLT script or a specific SPARQL query) is not part of this deliverable.

In addition to these required supported techniques the requirements also contain requirements on linking of datasets among each other. Each requirement on linking is specified including which source entities will be linked to which target entities of which dataset and how (e.g. NUTS codes from inspection results of CTIA to NUTS codes hierarchy of ramon.eu NUTS dataset). The specific source and column names etc. are, however, not part of the requirements.

The deliverable also contains requirements on aligning datasets, which will be published as Linked Open Data, with existing ontologies. It is not the purpose of COMSODE to develop new ontologies. The goal is to reuse existing ontologies as much as possible with a preference of those ontologies which are commonly reused by the Linked Open Data community. Therefore, the deliverable contains requirements on aligning datasets with those commonly reused ontologies. Each such requirement specifies that a given dataset must be aligned with a given ontology. However, the alignment script is not provided as it will be developed in WP4.

It is possible that a new dataset that was not specified in D3.1 was added during D3.2 because we found out that it is useful for linking.

During the analysis of the datasets, we also recorded metadata for each dataset which form its catalogue record. We will use the metadata later to build a data catalogue for datasets published by the COMSODE project. We do not describe the metadata in a more detail in this document. The metadata can be found in Attachment B to this deliverable.

2.2. Related Documents

List of related documents from project:
- DOW, version date 2013-08-06, pages 10-11
- Deliverable D3.1 Final version of the selected datasets list

2.3. Methodology used

2.3.1. Methodology

The main idea of this deliverable is to gather user requirements similar to the first step of a Software Development Life Cycle (SDLC) process [Sommerville 2010]. A SDLC is a process used by a software analysts to develop a software based on five phases. In particular, this work considers the first phase of the process that is requirements analysis. The requirement analysis is composed by three groups of requirements that are:

- requirements related to data transformation and integration;
- requirements related to data quality assessment and cleansing;
- requirements related to data consumption by data users.

For each group of requirements we provided the following methodology:

1. As the first step of the methodology we provide a list of the selected datasets (input from D3.1) by specifying their ID and their name. Further, we specify two features of the datasets that are:
a. **Available dataset**, determines whether the dataset is accessible given the URL;

b. **Analysed dataset**, determines that the structure and semantics of the data is known.

2. As the second step of the methodology, we provide for each dataset a list of requirements of the given type. For each dataset there is 1..N requirements and for each requirement there is a DPU which will be used by the developers in WP4 to implement techniques required by the requirement.

3. As the third step of the methodology we describe each DPU that we want to have in the context of the requirements gathered.

The complete list of gathered requirements is in Attachment A. It is a set of interconnected tables covering Task 3.2 (*Transformation and Linking Requirements, Target Ontologies*), Task 3.3 (*Data Quality Requirements, Quality Metrics*) and Task 3.4 (*Data Consumption Requirements*). There are two tables that all the Tasks use:

- “Datasets”, that lists all COMSODE datasets identified in D3.1 and some more that were required for better linking, and
- “Techniques - DPUs” that lists all techniques that are to be implemented and that cover all the identified requirements.

The “Datasets” table contains for each COMSODE dataset its ID and name from D3.1 (or a new one if the dataset was not present in D3.1).

The “Techniques - DPUs” table contains identified data processing techniques, which will be implemented as Data Processing Units (DPUs). For each DPU there is its ID, title, role (DPU Type) and its scope, i.e. whether it is reused in multiple datasets or in a single one (or a small related group of datasets). The identified requirements in the previous tables are mapped to those DPUs. The DPUs are split into following categories according to their role:

- **Extract** - a DPU which collects input data from data sources; it provides inputs to the dataset processing pipeline.
- **Transform** - a DPU which transforms the data (structural transformation, content transformation, anonymization and ontology alignment).
- **Quality** - a DPU that measures a chosen data quality metric
- **Load** - a DPU which loads the result of the dataset processing pipeline (produced data as well as metadata) to its final location (file, database, etc.)

The specific configurations of these DPUs such as SPARQL queries, XSLT style sheets or mappings from tabular data to RDF are not part of this deliverable as they will be developed later in WP4.

The requirements from Task 3.2 are further described in Section 3. The requirements from Task 3.3 are described in Section 4 and the requirements from Task 3.4 are described in Section 5.

### 2.3.2. **Datasets**

There were specified 166 datasets for publication by COMSODE in Deliverable 3. We extended this list by several additional datasets and achieved **178** datasets analyzed in D3.2 in total. This document summarizes the requirements and techniques identified for each analyzed dataset. The complete results of the analysis are in Attachments A and B.
to this document. This document refers to particular datasets by their ID. If the reader is interested in the details about the dataset, he can use the ID to find the details in Attachment A (requirements for the dataset) and Attachment B (dataset metadata like title, description, owner, EUROVOC classification, URL if it exists, etc.).

As we have already noted, there were few changes to the list of datasets identified in D3.1. One of the datasets, IT_X_08, became unavailable and another one, SK_CODELIST_0086_STD, was found to be a duplicate. These two datasets were removed from D3.2. On the other hand, we included 12 more datasets that will be processed within COMSODE and 6 datasets useful for linking, which will not be processed in COMSODE (we will only link our datasets to them).

2.3.3. Partner contributions

In this Deliverable, the work was split as follows:

- CUNI is the leader for this deliverable
- CUNI is the leader for T3.2
- UNIMIB is the leader for T3.3
- EEA is the leader for T3.4
- In each of T3.2, T3.3 and T3.4 each of the partners responsible for some datasets (CUNI, UNIMIB, MoI, SPINQUE) filled the requirements in each of the tasks for their own datasets.
3. Requirements and techniques for data transformation of the dataset and its integration with other datasets

We have identified various requirements on techniques of data transformation and integration (DPUs) by downloading and analyzing the datasets identified in D3.1. In total, we have identified 1794 requirements which specify how COMSODE datasets should be transformed and linked (or integrated) before they will be published on some ODN instance. For the complete list of the requirements, see Attachment A (sheet “Transformation and Linking Requirements”).

We classified the requirements into categories according to their role in the data transformation process from data extraction to data publication. The categories are:

- **Input** - a requirement on how and where to get the source data for processing in ODN.
- **Structural Transformation** - a requirement on transformation of data without a change in its meaning. Usually, this means changing the data model, e.g., from a table to RDF, from XML to RDF, from RDF data to RDF file, compressing and decompressing files, etc.
- **Content Transformation** - a requirement on transformation of data where meaning of the data is changed, usually improved.
- **Anonymization** - specifies a special kind of transformation which anonymizes the data, e.g., removes data items (table columns, XML elements, RDF triples, etc.) with birthdates, names and other identifiers, or aggregates data, etc.
- **Ontology Alignment** - makes sense only for datasets that are supposed to be published as Linked Open Data. It is a requirement that ensures that proper ontologies and vocabularies are used where appropriate. The goal of COMSODE is not to develop new ontologies. On the contrary, the goal is to show that there already are vocabularies that can be reused in almost every dataset, which makes the datasets interoperable and easily processed by different applications. We show that on the transformations done with our COMSODE datasets.
- **Integration** - requirement on fusion of data from several sources to a single dataset. Let us note that it is a different requirement from the next requirement which is linking. Linking does not fuse two datasets to a single dataset.
- **Linkage** - requirement on creating links between datasets. With this requirement we also specify the target dataset for linkage. For certain datasets, we identified DPUs that serve as linkers to these datasets. For example, T-ADDRESSCZ is a DPU that accepts a representation of a postal address according to the Schema.org vocabulary and tries to link that address to its equivalent in the CUZK_2 dataset, which contains all addresses in the Czech Republic. Therefore, T-ADDRESSCZ can be used on every dataset containing addresses from the Czech Republic and the result is their linkage to the CUZK_2 dataset to which many other datasets are already linked. This linkage then serves as an integration tool. Linkage does not fuse two datasets to a new dataset. Instead, it creates a new dataset which consists of links. VoID vocabulary understands it as a separate dataset and calls it linkset [VoID 2011].
- **Output** - requirement on format, place and means of the complete data and metadata to be stored for consumption. For example, if the resulting data is supposed to be stored on a local file system, this is a requirement to be implemented by the L-FS DPU, which enables to save data locally.
In this section we summarize requirements on the identified techniques (DPUs) that are independent of a specific dataset, then we summarize the requirements on identified techniques in the categories listed above that are going to be used in transformations of multiple datasets and will be therefore highly reusable and finally, we summarize techniques that are tied to a single dataset or a small number of related datasets and will not be reusable elsewhere.

### 3.1. Requirements on techniques independent of a specific dataset

It has shown that there are generic requirements which must be fulfilled by any DPU. We summarize them in this section.

#### 3.1.1. DPU logging

The DPUs implemented should use the RLOG vocabulary\(^1\) for representation of logs in RDF. This is useful because the logged messages can be further used, processed, stored as Linked Data and queried using SPARQL.

#### 3.1.2. Basic RDF representation

In many requirements on transformation of data to RDF (Linked Data) we talk about a basic RDF representation of tabular data. This is an RDF representation of tabular data that is created directly from the source table without using any content transformation, ontology alignment or linkage. The transformation of tabular data to the basic RDF representation should follow the recommendation which is being developed by W3C CSV On the Web Working Group\(^2\).

#### 3.1.3. Handle large datasets

DPUs need to be able to handle even bigger datasets that have more than 1 billion RDF triples as there are public open datasets of this size. The emphasis in ODN is on being able to process data of this size rather than optimizing for performance, which is beyond the scope of COMSODE.

### 3.2. Requirements on highly reused techniques used in processing of multiple datasets

We have analyzed the requirements on dataset transformations and linkage and identified techniques which are required for the publication process of more than a single dataset. For those techniques we proposed DPUs which will implement them. These datasets will be reused for different datasets so the development team in WP4 should first complete these reusable DPUs.

#### 3.2.1. Extract

We identified three techniques which we need to be able to extract data from their data sources before they are transformed and published. For each technique, we propose a

\(^1\) [http://persistence.uni-leipzig.org/nlp2rdf/ontologies/rlog/rlog.html](http://persistence.uni-leipzig.org/nlp2rdf/ontologies/rlog/rlog.html)

\(^2\) [http://www.w3.org/2013/csvw/wiki/Main_Page](http://www.w3.org/2013/csvw/wiki/Main_Page)
separate DPU. The DPUs and a short summary of the requirements for each DPU and a technique implemented by the DPU is listed below.

- **E-DWNLD - Generic downloader** - This DPU is required to download a file from a URL given in its configuration. Also, it is required to accept a list of URLs for download on its input. It needs to be able to handle HTTP, HTTPS, FTP including authentication (SFTP, FTPES).

- **E-SPARQL - SPARQL extractor** - This DPU is required to execute a given SPARQL CONSTRUCT query on a SPARQL endpoint, both public and accessible only with authentication.

- **E-FS** - This DPU is required to extract a file from a local file system (both Windows and Linux) and pass it on to the ETL pipeline.

3.2.2. Transform

Clearly, most of the requirements are related to how the data extracted from their data sources should be transformed before they are published. We identified 25 transformation techniques and proposed DPUs which implement them. The DPUs and a short summary of the requirements for each DPU and a technique implemented by the DPU is listed below.

- **T-SPARQL-U** - A DPU that transforms input RDF data into output RDF data using a SPARQL UPDATE query in its configuration - the result is the input RDF data updated by the query. The query editor will support strong/weak validation by selected vocabularies and strong/weak autocomplete from selected vocabularies. Strong means that it is not allowed to enter property/class which would not be in (or valid according to) a vocabulary. Weak means a warning. This DPU is heavily reused for Content transformation, Linkage and Ontology alignment requirements.

- **T-SPARQL-C** - A DPU that transforms input RDF data into output RDF data using a SPARQL CONSTRUCT query in its configuration. The result is the RDF data constructed by the query from the input. The query editor will support strong/weak validation by selected vocabularies and strong/weak autocomplete from selected vocabularies. Strong means that it is not allowed to enter property/class which would not be in (or valid according to) a vocabulary. Weak means a warning. This DPU is mostly reused for Linkage and Ontology alignment requirements.

- **T-MDT - Metadata DPU** - This DPU is required to be able to gather all relevant information to be put in the dataset metadata, including cataloging metadata and statistics like, e.g., numbers of triples, subjects, predicates and objects in RDF data. This DPU implements a content transformation technique which enriches a dataset with its metadata and it is reused for each COMSODE dataset.

- **T-XSLT** - A DPU that gets XML data on the input, XSLT template as a configuration and produces the resulting files. The DPU must support XSLT parameters processing and needs to be able to process large XML files (1GB+). This DPU is mostly reused for structural and content transformation and also linkage and ontology alignment. It is able to work only with XML data on the input but it can produce any kind of data (tabular, XML or even RDF data).

- **T-FILE2RDF** - A DPU that transforms input RDF file to RDF data for further processing in the data processing pipeline.

- **T-TABULAR** - Transforms input tabular data (DBF, CSV with headers, CSV without headers, XLS, XLSX, ODS) into its basic RDF representation according to
W3C specification of representation of CSV data in RDF\(^3\). For data without headers a mapping needs to be specified in the configuration. It is heavily reused for structural transformation requirements which need to produce RDF representation from tabular inputs.

- **T-SPARQL-SELECT** - Transforms RDF data to Tabular data using SPARQL SELECT query. The output formats supported need to include CSV. It implements a reversed technique to the technique implemented by T-TABULAR.

- **T-UNZIP** - Unzips input ZIP file and passes on the result. It is reused for all datasets which are produced from a ZIP archive on the input data source.

- **T-RDF2FILE** - A DPU that transforms input RDF data to output RDF file, optionally produces a .graph file containing RDF graph URI which is useful for loading dumps into OpenLink Virtuoso.

- **T-ZIP** - Zips the input files and passes on the ZIP file. It is necessary for datasets for which we will publish dumps.

- **T-LINKER** - Uses Silk\(^4\) to create links from RDF data to other RDF datasets. It is reused for Linkage requirements.

- **T-DCV** - A DPU that helps convert statistical RDF data to Data Cube Vocabulary representation. Internally, it will be a SPARQL transformer, but it will have a graphical wizard in the dialog, allowing users to load sample data and to specify how to create a datacube from it graphically. Similar to the Data Cube Plugin in an RDF visualization tool Payola\(^5\).

- **T-ENTITY-ANN** - A DPU that takes a set of texts and set of RDF resources with (possibly multilingual) labels. It recognizes them in the texts and uses Open Annotations vocabulary to represent the found annotations in RDF.

- **T-CZECHBEIC** - A DPU that transforms input RDF list of Czech Business Entity identifiers into their proper RDF representation using SKOS\(^6\) and optionally ROV\(^7\) vocabularies and links to the MFCR_1 dataset with data about Czech business entities.

- **T-PICKFILE** - Picks files from File Data Unit based on the file name or a regular expression filter.

- **T-HTMLCSS** - A DPU that gets an HTML page and parses RDF data from it using mapping from CSS selectors to RDF properties. It is heavily reused for structural transformation requirements which need to produce RDF representation from HTML inputs.

- **T-ADDRESSCZ** - A DPU that takes schema.org s:PostalAddress instances and their properties and links them to a corresponding CUZK_2 object. It is expected that if other countries will provide their address registries for free, it will be possible to develop similar DPUs specific for their countries.

- **T-SPSS** - A DPU that converts an SPSS file to CSV by using the SPSS File Conversion Service located at http://pspp.benpfaff.org/.

---

3 [http://w3c.github.io/csvw/csv2rdf/](http://w3c.github.io/csvw/csv2rdf/)
4 [http://wifo5-03.informatik.uni-mannheim.de/bizer/silk/](http://wifo5-03.informatik.uni-mannheim.de/bizer/silk/)
5 [http://www.payola.cz](http://www.payola.cz)
6 [http://www.w3.org/2004/02/skos/core#](http://www.w3.org/2004/02/skos/core#)
7 [http://www.w3.org/ns/regorg#](http://www.w3.org/ns/regorg#)
T-CZLAW - A DPU that gets a text representation of an identifier of a Czech law and tries to link it to the CZLAW dataset. It is an implementation of a specific linkage requirement.

T-PDF2CSV - Extracts tabular data from PDF to CSV.

T-PROVOZCZ - Links to places of business of Czech Business Entities in MFCR_3 based on addresses, business entities and names. It is an implementation of a specific linkage requirement.

T-CHEMICALNAMECZ - A DPU that links representations of chemical compounds in two different datasets with sameAs links on the base of their names. The classes with represent compounds in both datasets are given as parameters. It is an implementation of a specific linkage requirement.

T-BENAME - A DPU that links representations of Business Entities to Czech Business Entities in MFCR_1 based on their name. It is an implementation of a specific linkage requirement.

T-7UNZIP - Unzips input 7ZIP file and passes on the result.

T-PDF - Extracts all texts from a PDF file so that formatting of the PDF file is preserved.

3.2.3. Load

The last but not least, there are requirements on how datasets prepared by transformation DPUs should be loaded to a database or a file system so that other components of ODN can publish them as open data. We identified 3 loading techniques and proposed DPUs which implement them. The DPUs and a short summary of the requirements for each DPU and a technique implemented by the DPU is listed below.

L-RFS - Loads the input files to an SCP/FTP server - this includes all the usual encryption and authentication types and transfer protocols: FTP, FTPS, SFTP, FTPES, SCP and possibly more.

L-SPARQL - Loads the input RDF data to a SPARQL endpoint - compliant with SPARQL 1.1 Update and SPARQL 1.1 Graph Store Protocol including authentication options. Needs to be resistant to database crash and similar unexpected interruptions.

L-FS - Loads the input files to a local file system. Both Linux and Windows paths need to be supported.
3.3. Requirements on techniques specific to a single dataset or a small number of related datasets

In the requirements, we also identified techniques which are specific for particular datasets. They will also be implemented in the form of DPUs. However, these DPUs will not be reused for different datasets.

3.3.1. Extract

These DPUs are used when the dataset needs some specialized functionality to be extracted. This usually means that there is some special file format or request format to get a list of files to be downloaded or that the data is hidden behind some kind of HTML search form that needs correct cookies to be added to each request.

- In the case of **E-RUIAN** one needs to download and parse a text file including links to the actual data files.
- In **E-MHCR-PRICES**, the data is in a form of series of HTML pages.
- In **E-CZ_MZP_02:001**, **E-VAVAL_1:001**, **E-VAVAL_2:001**, **E-VAVAL_3:001**, **E-VAVAL_4:001**, **E-VAVAL_5:001**, **E-VAVAL_6:001**, **E-VAVAL_7:001**, **E-CZ_MZP_04:001** and **E-CZ_MZP_01:001**, **E-SK_53**, **E-SK_VLADA_CRP**, **E-SK_VLADA_CRZ**, **E-SK_MARTIN_CONTRACTS**, **E-SK_MARTIN_TAXDEPT**, **E-SK_DATANEST_ORGS**, **E-SK_ORSR_ORGS**, **E-SK_MARTIN_BUDGET** and **E-SK_SOI_RESULTS** there is an HTML form that needs to be filled out and an appropriate request needs to be sent in order to get HTML pages with data, the lists of items are paged, so that the DPU needs to follow the paging links which are different in each of the targeted portals.
- In **E-MFCR_1:001** and **E-MFCR_3:001** the DPUs need to check a limit for number of downloads in two time intervals during one day.
- **E-MICR_1:001** downloads an XML list of entities including links to details and then needs to download the detailed XML files.
- **E-MICR_3:001** parses HTML pages and creates RDF data.
- **E-SUKL-ATC**, **E-SUKL-MPP** and **E-SUKL-CL** download ZIP files and HTML pages with URLs that need to be calculated according to the current date.
- **E-SK_DOMAIN_ORGS** - A DPU, which for each "ID drzitela" from **SK_DOMAIN_LIST** queries http://whois.sk-nic.sk/index.jsp?whois={ID drzitela} and stores the results as one record in a CSV file.

3.3.2. Transform

These DPUs are used for dataset-specific transformations that cannot be achieved simply by proper configuration of a generic DPU from the previous section. These include **T-TEXT-SPC** and **T-TEXT-PIL** which do various text transformations (mainly domain specific segmentation to sections and paragraphs).
3.4. Overview of statistics of requirements and techniques for data transformation

We have identified a total of 1794 transformation and linkage requirements that will be covered by a total of 64 DPUs. The top DPUs as to the number of requirements can be seen in Figure 1. The development team in WP4 should consider these statistics and start their work with the top DPUs which are highly reused to produced many different COMSODE datasets.

![Figure 1: Number of transformation and linking requirements on DPUs](image)

In Figure 2, there are numbers of requirements per type, the top being Output requirements, Structural transformation requirements and Ontology alignment requirements.

![Figure 2: Types of transformation requirements](image)

In the ontology alignment requirements, there is the target ontology specified. In Figure 3 there are numbers of datasets to be aligned with the specified ontology, which is further described in the attachment.
In Figure 4 there are numbers of datasets described in D3.2 per partner. In Figure 5 there are numbers of requirements specified per partner. In Figure 6 there are average numbers of requirements per partner’s dataset.
Links among datasets are visualized in Figure 7. A line between two datasets means that there are links identified between these two datasets and that those datasets will be linked according to Linked Open Data principles after they are published by the COMSODE project. The larger the dataset on the diagram, the more datasets link to it. The diagram illustrates a map of links which will be published by the COMSODE project after all datasets are published.
Figure 7: Links among datasets

Figure 8 contains the number of inbound links per dataset. These are numbers of other datasets linked to the observed one. The datasets with the largest numbers form cores of COMSODE linked data space. They are CUZK_2, NUTS, SK ORSR ORGS, MFCR_1 and SK_CODELIST_0025_STD. Figure 9 contains the number of outbound links per dataset. These are numbers of datasets linked from the observed datasets. The numbers show that each of our datasets links to approximately the same amount of other datasets.
**Figure 8:** Numbers of inbound links per dataset

**Figure 9:** Numbers of outbound links per dataset
4. Requirements and techniques for the quality assessment and cleansing of the dataset

Data quality problems refer to a set of issues that can affect the potentiality of the applications built on top of data. Data quality, being a multidimensional concept may be evaluated along many dimensions such as accuracy, completeness, timeliness and trustworthiness [Batini 2006]. A dimension refers to an abstract definition used for assessing data quality and thus, the assessment of data quality needs metrics which provide a concrete evaluation of quality. Data quality is commonly conceived as fitness for use, which implies that data useful for a certain application or use case may not be useful for another application or use case. Therefore, the assessment of data quality relies on the subjective requirements of data consumers. Thus, a first step before evaluating the quality is to collect the requirements based on consumer needs. Second step is to assess quality based on intrinsic and contextual metrics. Intrinsic metrics are those that are independent of the consumer/application’s context and can be applied to any dataset. Thus, intrinsic metrics are valid for every application the data is used for, while contextual metrics are valid according to the applications. For this reason, we distinguish two kinds of quality metrics into intrinsic and contextual.

In the following sections we will provide the quality requirements for the selected datasets given in the Attachment A of D3.1. These quality requirements will serve as input for the selection of quality dimensions and in particular the quality metrics. Each metric will be associated with a DPU.

4.1. Requirements and techniques for quality assessment

All necessary quality requirements are specified from the consumer who wants to evaluate the quality of its dataset to guarantee a full exploitation of the published Open Data. The quality requirements proposed may be both at metadata and data level. In total, we have identified 1345 quality requirements which specify how quality assessment should be done for COMSODE datasets before they will be published on some ODN instance. For the complete list of the requirements, see Attachment A (sheet “Data Quality Requirements”).

4.1.1. Normalized scores for quality metrics

The process of measuring data quality is based on metadata annotations as well as data analysis. Our data quality score metrics are based on simple ratio calculation. The simple ratio is measured by subtracting the ratio between the total number of instances that violate a data quality rule (V) and the total number of relevant instances (T) from one, as the following formula shows:

$$DQ\_score = 1 - \frac{V}{T}$$  \hspace{1cm} (1)

This score can be applied for each property of the dataset. In case we want to calculate the quality of the overall properties/attributes in a dataset, the above DQ\_score is multiplied with a weight \(w_i\) representing the importance of the intended task for each property in the dataset and divide by the sum of the weighted DQ\_score by the sum of all weighting factors of the regarded properties (W).

$$DQ\_score\_weighted = \frac{\sum (DQ\_score \times w_i)}{W}$$  \hspace{1cm} (2)

In case of equal importance of the properties for the task at hand or in case it is not possible to annotate importance values, all \(w_i\) are considered equal to 1 and the \(W\) value gives the number of all properties that are tested in the dataset. While in the former
case the DQ_score_weighted is a contextual metric in the latter case it is considered to be an intrinsic metric.

4.1.2. Metadata Quality Requirements and Techniques

The COMSODE datasets have mandatory attributes as described in Deliverable 2.2 (page 12). These attributes will be provided by the consumers/providers of the dataset as one of the first steps in the platform, which will be stored in an internal catalog similar to the attachment A in Deliverable 3.1. Some of the mandatory attributes may be considered as \textit{quality related metadata} because they provide necessary information to understand data and evaluate them. But, some of them contain values such as Unknown and NA thus they cannot be used for further assessment. Metadata quality can be considered as a necessary condition for the quality of a dataset. According to the values provided with the mandatory attributes we may assess the quality of metadata in terms of completeness, accuracy, interpretability and licensing (http://goo.gl/skebft).

Metadata plays an important role in supporting metrics evaluation since they store aspects of data relevant to data quality. Thus, when it comes to assessing the temporal validity of the document, it is not possible since such values are missing. Table 1 gives a list of requirements for metadata quality assessment. Each requirement is associated with a DPU ID.

<table>
<thead>
<tr>
<th>Description</th>
<th>DPU ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Missing metadata elements from a recommended list of metadata</td>
<td>Q-MC</td>
</tr>
<tr>
<td>Missing reference to the language used by the dataset</td>
<td>Q-IN</td>
</tr>
<tr>
<td>Incorrect formats of temporal metadata associated with the dataset</td>
<td>Q-MACC</td>
</tr>
<tr>
<td>Inaccessibility of the record describing the mandatory attributes of the dataset</td>
<td>Q-AV</td>
</tr>
<tr>
<td>Less precision or completeness than expected for the licensing</td>
<td>Q-LC</td>
</tr>
</tbody>
</table>

\textbf{Table 1:} List of requirements for metadata quality assessment and related techniques (implemented by the referred data processing units)

In the following part we provide an explanation for each DPU employed for the metadata assessment and for each DPU we provide a checklist of the metadata quality. Each checklist provides a dichotomic questions for which we can answer with yes (denoted 1) or no (denoted by 0).

\textbf{Q-MC:} A DPU that assesses the metadata completeness. The assessment refers to the presence or not of the metadata considered under the completeness dimension:

- The identity and role of the person responsible for each dataset is specified
- Each dataset includes at least a title and a description
- A creation date is given for each dataset
- A last-updated date is given for each dataset
• The datasets are categorized by domain
• The format of downloadable files is indicated
• The character encoding used in each dataset is declared
• One can access different versions of datasets
• Each dataset is given a unique identifier
•Datasets are accompanied by at least one means of contact for the producer (or maintainer)
• If a dataset contains an expiry date, it is declared
• An update frequency is given for each dataset
• The publisher of the dataset provides information on its origin

Q-IN: A DPU that assesses interpretability. The assessment refers to the presence or not of the metadata considered under the interpretability dimension.
• Each dataset is accompanied by a reference to the language used
• Each dataset is accompanied by a schema

Q-MACC: A DPU that assesses the metadata accuracy. The assessment refers to the presence or not of the metadata considered under the accuracy dimension.
• Dates are given in standard formats
• Dates are available in a standard format

Q-AV: A DPU that assesses the availability. The assessment refers to the presence or not of the metadata considered under the availability dimension.
• The descriptive record contains a direct link to the URL of the data
• It should be checked whether the URL can be dereferenced

Q-LC: A DPU that assesses the licensing. The assessment refers to the presence or not of the metadata considered under the licensing dimension.
• The datasets are accompanied by a license
• The license sets out the conditions of attribution, reuse, redistribution and commercialization
• Data producers declare their policy on releasing data
• The datasets are accompanied by a summary and a link to the full version of the license

The above list of DPUs proposes guidelines to support metadata annotators which can also be considered as a first step for metadata improvement.

4.1.3. Data Quality Requirements and Techniques

Table 2 gives a list of data quality requirements. Each requirement is associated with a DPU ID. The following quality requirements propose a generic list.

<table>
<thead>
<tr>
<th>Quality Dimension</th>
<th>Quality Requirement</th>
<th>DPU ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>Inaccuracy of literals of property</td>
<td>Q-ACC_1</td>
</tr>
<tr>
<td>Accuracy</td>
<td>Inaccurate RDF modeling of the dataset</td>
<td>Q-ACC_2</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>----------------------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>Accuracy</td>
<td>Incorrect values not prescribed in the value set of the reference dataset</td>
<td>Q-ACC_3</td>
</tr>
<tr>
<td>Accuracy</td>
<td>Inaccuracy of values/literals of an attribute/property</td>
<td>Q-ACC_4</td>
</tr>
<tr>
<td>Accuracy</td>
<td>Inaccuracy of values of an attribute</td>
<td>Q-ACC_5</td>
</tr>
<tr>
<td>Accuracy</td>
<td>Incorrect values according to a specified pattern</td>
<td>Q-ACC_6</td>
</tr>
<tr>
<td>Accuracy</td>
<td>Incorrect formats of date values</td>
<td>Q-ACC_7</td>
</tr>
<tr>
<td>Completeness</td>
<td>Empty literal attached to a particular data type property</td>
<td>Q-C_1</td>
</tr>
<tr>
<td>Completeness</td>
<td>Empty literal attached to all data type properties in the dataset</td>
<td>Q-C_2</td>
</tr>
<tr>
<td>Completeness</td>
<td>The absence of a particular data type property for a certain instance</td>
<td>Q-C_3</td>
</tr>
<tr>
<td>Completeness</td>
<td>The absence of values for a certain data type property with respect to the reference dataset</td>
<td>Q-C_4</td>
</tr>
<tr>
<td>Completeness</td>
<td>The absence of understandability metadata</td>
<td>Q-C_5</td>
</tr>
<tr>
<td>Completeness</td>
<td>The absence of temporal metadata</td>
<td>Q-C_6</td>
</tr>
<tr>
<td>Consistency</td>
<td>Inconsistent values</td>
<td>Q-CN_1</td>
</tr>
<tr>
<td>Consistency</td>
<td>Inconsistent values in RDF data</td>
<td>Q-CN_2</td>
</tr>
<tr>
<td>TRQD</td>
<td>Out-of-date data</td>
<td>Q-CU_1</td>
</tr>
<tr>
<td>TRQD</td>
<td>Out-of-date triples</td>
<td>Q-CU_1</td>
</tr>
<tr>
<td>TRQD</td>
<td>Out-of-date data with respect to the specific task</td>
<td>Q-T</td>
</tr>
</tbody>
</table>

**Table 2:** A list of data quality requirements and related techniques (implemented by the referred data processing units)

The quality dimensions that we considered for the datasets are: completeness, accuracy, timeliness and consistency.
Completeness refers to the degree to which all required information is present in a particular dataset. Completeness can be characterized in a more precise way with respect to:

(a) presence/absence and meaning of null values;
(b) concept of reference dataset.

Case (a) is covered by using a data profiling tool which identifies the null values in the table. In the model with null values specific definitions for completeness can be provided by considering the granularity of the model elements: property (of a triple) and dataset:

- **Q-C_1**: property completeness measures the number of null values of a specific property in a dataset;
- **Q-C_2**: dataset completeness captures the presence of null values in a whole dataset.

The number of instances that violate a data quality rule (V) is thereby composed by the number of null values. Then, we calculate the property completeness by using formula (1) and the dataset completeness by using formula (2).

Case (b) is covered if a reference dataset is provided. Given the dataset D, we assume that there is a reference dataset of D, called ref(D). ref(D) is the dataset containing all the triples that represent objects of the real world that constitute the present true extension of the schema.

**Quality evaluation based on reference datasets**

Let us consider the case that there exists a reference dataset representing the real-world data or the user specifies the required classes/properties. Consider we have our dataset that we want to test (test dataset) and the reference dataset. First we need to apply a search space reduction in order to apply further steps such as the comparison functions. A comparison function measures the similarities of two values. For example, if we need an accuracy of the city names, we need a comparison function that measures the distance between the city name in our dataset and the city name given in the reference dataset. Comparison functions have been widely investigated, especially string comparison functions such as edit distance, n-grams, Soundex code, Jaro algorithm, Hamming distance, Smith-Waterman, TF-IDF [Navarro 2001].

In the model with missing values several specific definitions for completeness can be provided by considering:

- **Q-C_3**: schema completeness is the degree to which classes and properties of a schema are represented with respect to the reference dataset or as input specified by the user;
- **Q-C_4**: property completeness is the degree of missing values for a specific property of a dataset with respect to the reference dataset;
- **Q-C_5**: understandability of a resource is measured as the presence of human readable labels and descriptions of resources;
- **Q-C_6**: temporal metadata completeness of a resource is measured as the presence of temporal metadata such as the creation and last modification date of a resource.

Incomplete values can be captured as the proportion of the true data that is stored in the dataset D:

$$DQ\_score\_ref=\frac{|\text{ref}(D)\cap D|}{|\text{ref}(D)|} \quad (3)$$
In case of Q-C_3, the classes and properties identified in the dataset D will be compared with those present in the reference dataset and the intersection will provide the number of unique objects (classes and properties) belonging to ref(D) and D.

In case of Q-C_4, the values of a property identified in the dataset D will be compared with those present in the same property of the reference dataset and the intersection will provide the number of unique values belonging to ref(D) and D.

In case of Q-C_5, the properties used to represent understandability information identified in the dataset D will be compared with those present in the reference dataset and the intersection will provide the number of unique properties belonging to ref(D) and D.

In case of Q-C_6, the properties used to represent temporal metadata information identified in the dataset D will be compared with those present in the reference dataset and the intersection will provide the number of unique properties belonging to ref(D) and D.

**Accuracy** refers to the extent to which entities and facts are correct, that is, the degree to which they correctly represent the real-life phenomenon. Accuracy can be classified into:

(a) syntactic accuracy, defined as the degree to which values correctly represent the domain values of the underlying vocabularies and the degree to which values conform to the syntax of its definition;

(b) semantic accuracy is defined as the degree to which data values correctly represent the real world facts.

Case (a) comprises a set of metrics used for the assessment of syntactic accuracy:

- **Q-ACC_1**: detection of ill-typed literals which do not abide by the lexical syntax for their respective data type that can occur if a value is (i) malformed or (ii) is a member of an incompatible data type;
- **Q-ACC_2**: detecting syntax errors using validators;
- **Q-ACC_3**: use of legal value rules given by the reference data. Each literal value is compared against the legal value through a comparison function such as, edit distance, jaro algorithm, etc;
- **Q-ACC_4**: numerical range values where each attribute value should be delimited by its range defined in the schema or in the reference dataset or as input specified by the user;
- **Q-ACC_6**: use of patterns to identify incorrect values such as the postal address, the phone number, the email address the personal identification number, etc;
- **Q-ACC_7**: use of regular expressions to identify date information in date, dateTime, gYearMonth, gYear, gMonthDay, gDay and gMonth formats.

Case (b) comprises a set of metrics used for the assessment of semantic accuracy:

- **Q-ACC_5**: detecting inaccuracies by using functional dependencies [Kolahi 2009].

*Time-Related Quality Dimensions* (TRQD) capture important aspects of data regarding changes and updates in time (i.e. the dynamic nature of LD). Important aspects of time-related quality dimensions are data freshness over time (currency) and data freshness over time for a specific task (timeliness). We distinguish between: (a) currency concerns how promptly data are updated, and (b) timeliness expresses how current data is for the task at hand.
Case (a) should provide a set of metrics for the assessment of data up-to-dateness:

- **Q-CU_1**: the currency of a document is measured as the age of the document, where the age of a document is computed as the difference between the current time (the observation time) and the time when the document was last modified.

- **Q-CU_2**: the currency of a triple/set of triples is measured as the average currency of the documents describing the entities occurring in the triple.

Case (a) should provide a set of metrics for the assessment of data timeliness:

- **Q-T**: is measured based on two components that are, currency and volatility. For currency see case (a) above. Volatility is the length of time the data remains valid which is measured by the frequency of change of the value for a property.

All metrics belonging to this group will be given a value in a continuous scale from 0 to 1, where score of 1 implies that the data is timely and 0 means it is completely outdated.

**Consistency** captures the violation of semantic rules defined over a dataset. It comprises a set of metrics for the assessment:

- **Q-CN_1**: A DPU that assesses that the values are consistent by using business rules.

- **Q-CN_2**: A DPU that assesses that the values are consistent by using test SPARQL queries.

### 4.2. Requirements and techniques for data cleaning

In order to improve the quality we should first consider the scores provided by the quality assessment of the datasets. In general, data cleaning can be provided either during (a) the process of data transformation where the user is aware about the quality issues occurring in the dataset or (b) if the user is not completely sure about the quality issues occurring in the dataset, in which case he will ask for an assessment phase and then according to the scores he retrieves, he will decide which cleaning DPUs to apply.

In this section, we provide two DPUs where two of them are considered to be generic. The two generic DPUs will cover most of the quality issues specified by the user and identified by the DPUs during the assessment.

- **T-Pattern**: use patterns represented by regular expressions to correct value such as numbers, date, postal addresses, email addresses

- **T-EntityCoreference**: find which identifiers refer to the same real world entity between the data source and the reference data source and then clean the values of that entity with respect to the values of the same entity in the reference data source.
4.3. Overview of statistics of requirements and techniques for data quality assessment

We have identified a total of 1345 quality requirements that will be covered by a total of 23 DPUs. The top DPUs as to the number of requirements can be seen in Figure 10. The development team in WP4 should consider these statistics and start their work with the top DPUs which are highly reused to produce many different COMSODE datasets.

![Figure 10: Number of quality requirements on DPUs](image)

Figure 11 presents the number of quality requirements per type of quality dimensions, the top being Accuracy, Accessibility, Metadata quality dimensions (e.g. Metadata Completeness), Completeness, TRQD and Consistency.

![Figure 11: Quality requirements grouped by quality dimensions](image)
In Figure 12 there are numbers of quality requirements specified per partner.

Figure 12: Number of quality requirements by partner

In Figure 13 there are average numbers of quality requirements per partner’s dataset.

Figure 13: Average number of quality requirements per partner’s dataset
5. Requirements for consuming curated datasets

The goal of Task 3.4 was to specify for each dataset a way curated datasets (cleansed, integrated, and quality assessed) can be consumed as data dumps in open and machine readable formats (such as CSV, RDF), or via APIs (application programming interfaces, such as RESTful service, or SPARQL endpoint). The responsible project partner of each dataset selected the most suitable combination of data dumps (their formats) and APIs (their protocols and formats) for each dataset based on the contents and quality of the dataset, its intended audience and expected usage.

For the complete list of the requirements, please see Attachment A, sheet Data Consumption Requirements. For each dataset, the responsible partner specified how the dataset should be consumable (i.e., which types of data consumption should be available for the given dataset):

- **CSV dump** - Yes/No - The value yes means that data consumers should be able to download CSV dumps of the curated dataset. Data consumers may also download metadata about the dataset.

- **CSV API - REST SERVICE** - Yes/No - The value yes means that data consumers should be able to access CSV data of the curated dataset via REST API service, which will allow data consumers to get all the data or certain projections/selections on top of that data..

- **RDF dump** - Yes/No - The value yes means that data consumers should be able to download RDF dumps of the curated dataset. Data consumers may also download metadata about the dataset. If the responsible partner has a preferred format of the RDF dump, the partner can specify such format in the column RDF dump format (e.g., RDF/XML, or TTL).

- **RDF API - SPARQL ENDPOINT** - Yes/No - The value yes means that data consumers should be able to access RDF content of the curated dataset using SPARQL endpoint - they may query the RDF data using SPARQL Query language.

- **Dumps in other formats** - If the curated dataset should be available for data consumption also in dumps in other formats, e.g., XML, JSON, the responsible partners may specify such need.

5.1. Overview of statistics

Figure 14 shows how many datasets support the outlined types of data consumptions - CSV (dump/API), RDF (dump/API).

As depicted in Figure 14, most of the datasets will be available as RDF dumps, via RDF API (SPARQL endpoint). Responsible partners required TTL (Turtle) as RDF dumps’ format for 50 datasets, other datasets published as RDF dumps may be published using arbitrary RDF data serialization. Responsible partners did not require any other format apart from RDF and CSV.

![Figure 14: Types of data consumptions](image-url)
6. Conclusions

Deliverable provides a summary report of identified requirements and techniques for data transformation and linkage, data quality and cleansing, and data consumption of the datasets that will be published by the COMSODE project as (linked) open data.

The requirements and techniques were identified on basis of a detailed analysis of the content and expected usages of the datasets. We started by analysing of 166 datasets identified in D3.1 and extended this list by several new datasets. Finally, we analyzed 178 datasets. For these there were identified 1794 requirements on data transformation and linkage, and 1345 requirements on data quality and cleansing of the datasets. We also defined how the datasets should be published by the ODN publication platform for the dataset users (application developers).

Even though only 30 datasets should be published as Linked Open Data according to the original COMSODE plan, this deliverable proposes to publish almost all of them as Linked Open Data (161 out of 178 datasets). For each dataset, we also registered metadata that will be later used by the COMSODE project to create a data catalogue of the published datasets. All identified requirements are listed in Attachment A. All metadata are listed in Attachment B.

The results of the deliverable serve as an input for WP4. The deliverable introduces a list of data processing units (DPUs) which must be implemented in ODN. The deliverable provides requirements and specification for each identified DPU and how the DPU will be used to produce respective datasets. To publish each dataset, we will combine the developed DPUs together as specified by this deliverable. The combination will be called a data processing pipeline, or simply a pipeline. A pipeline extracts data from a primary data source, transforms the data to a proper (linked) open data form and publishes the resulting dataset for consumption by the dataset users. For these particular steps, DPUs proposed by this deliverable will be reused.
List of figures and tables

**Figure 1**: Number of transformation and linking requirements on DPUs  
**Figure 2**: Types of transformation requirements  
**Figure 3**: Number of datasets aligned with ontologies  
**Figure 4**: Numbers of datasets per partner  
**Figure 5**: Number of transformation and linkage requirements by partner  
**Figure 6**: Average number of requirements per partner’s dataset  
**Figure 7**: Links among datasets  
**Figure 8**: Numbers of inbound links per dataset  
**Figure 9**: Numbers of outbound links per dataset  
**Figure 10**: Number of quality requirements on DPUs  
**Figure 11**: Quality requirements grouped by quality dimensions  
**Figure 12**: Number of quality requirements by partner  
**Figure 13**: Average number of quality requirements per partner’s dataset  
**Figure 14**: Types of data consumptions

**Table 1**: List of requirements for metadata quality assessment and related techniques (implemented by the referred data processing units)  
**Table 2**: A list of data quality requirements and related techniques (implemented by the referred data processing units)
References


