D5.19

Library and tool for data mapping and conversion - v2

ABSTRACT

This document accompanies the release of the demonstrator software components that are responsible for the conversion of electronic health records across local and national healthcare standards inside the InteropEHRate project. It describes both the tools and the health knowledge resources that support the process of knowledge-based data mapping and conversion.

Delivery Date	July 9th, 2021
Work Package	WP5
Task	T5.5
Dissemination Level	Public
Type of Deliverable	Demonstrator
Lead partner	UNITN

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826106.

This document has been produced in the context of the InteropEHRate Project which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826106. All information provided in this document is provided "as is" and no guarantee or warranty is given that the information is fit for any particular purpose.

CONTRIBUTORS

	Name	Partner
Contributors	Gábor Bella	UNITN
Contributors	Simone Bocca	UNITN

LOGTABLE

Version	Date	Change	Author	Partner
0.1	16/06/2021	Created from v1	Gábor Bella	UNITN
0.2	24/06/2021	Updates	Gábor Bella	UNITN
0.3	28/06/2021	Updates	Gábor Bella	UNITN
0.4	29/06/2021	Reorganised and added material	Gábor Bella	UNITN
0.5	30/06/2021	Added missing material	Simone Bocca	UNITN
0.6	30/06/2021	Ready for QC	Gábor Bella	UNITN
0.7	01/07/2021	Quality check	Argyro Mavrogiorgou	UPRC
VFinal	9/7/2021	Technical final revision and submission	Francesco Torelli Laura Pucci	ENG

ACRONYMS

Acronym	Term and definition
EHR	Electronic Health Record
FHIR	Fast Healthcare Interoperability Resources
ICD	International Classification of Diseases
LOINC	Logical Observation Identifiers Names and Codes
SNOMED CT	Systematized Nomenclature of Medicine -- Clinical Terms
WHO ATC	World Health Organization's for active substances of pharmaceutical products
CDA	Clinical Document Architecture

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

1.1 Scope of the document

This document provides details about the final (v2) demonstrator release of data, tools, and libraries for health data mapping and conversion. The details refer to:

- an overview of the knowledge-based data mapping and conversion process and the components, tools, and resources involved;;
- how to build, install, and use the InteropEHRate Health Tools for supporting data mapping and conversion;
- formal health knowledge made available by the project;
- release modalities.

1.2 Intended audience

The document is intended for people interested in deploying and running the InteropEHRate demonstrators on data mapping and conversion, as well as for those looking for an overview of the related functionalities the InteropEHRate project has implemented.

1.3 Structure of the document

Section 2 contains three sub-sections describing the tools, the pivot (international) knowledge, and the source (local) knowledge packages. Section 3 provides a textual overview of the data mapping and conversion process, and how the tools and knowledge are used within it. Sections 4, 5, and 6 provide building, installation, and succinct user guides for the interactive Knowledge Management Tools, the batch-mode Knowledge Importer, and the Data Mapper Tool, respectively. Section 7, finally, describes the knowledge resources released by the InteropEHRate project.

1.4 Updates with respect to previous version (if any)

The deliverable has been completely reorganised for general understandability as well as in order to make a clear distinction between tools and knowledge resources. The tools and their purpose are described much more clearly in sections 3-6. A user guide for each tool is now provided. Details about tool capabilities and knowledge support have been updated according to progress in the project.

2. SW DESCRIPTION

2.1 Tools

SW TITLE	Knowledge Management Tools
SW VERSION	1.0.0
LICENCES AND PATENTS	proprietary (UNITN)
PROGRAMMING LANGUAGES	Node.js
SUPPORTED PLATFORM(s)	Cross-platform
SOURCE CODE	Docker image

Table 1 – Release information on the Knowledge Management Tools

The Docker image for the Knowledge Modeller Tools can be provided on request by University of Trento.

SW TITLE	Data Mapper Tool
SW VERSION	1.0.0
LICENCES AND PATENTS	Apache-2.0
PROGRAMMING LANGUAGES	Java
SUPPORTED PLATFORM(s)	Cross-platform
SOURCE CODE	https://github.com/usc-isi-i2/Web-Karma
EXECUTABLE	https://github.com/usc-isi-i2/Web-Karma

Table 2 – Release information on the Data Mapper Tool

2.2 Pivot Knowledge

STANDARD NAME	SNOMED CT International
IMPORT FILE FORMAT	Excel (xls)
REPOSITORY	http://iehrgitlab.ds.unipi.gr/interopehrate/health-tools/knowledge-mgmt-tool.git (Import Files/SNOMED)

Table 3 - SNOMED Standard

STANDARD NAME	ICD-10
IMPORT FILE FORMAT	CSV
REPOSITORY	http://iehrgitlab.ds.unipi.gr/interopehrate/health-tools/knowledge-mgmt-tool.git (Import Files/ICD)

Table 4 - ICD-10 Standard

STANDARD NAME	FHIR 4.0 IPS
IMPORT FILE FORMAT	Excel (xls)
REPOSITORY	http://iehrgitlab.ds.unipi.gr/interopehrate/health-tools/knowledge-mgmt-tool.git (Import Files/FHIR)

Table 5 - FHIR Standard

STANDARD NAME	LOINC
IMPORT FILE FORMAT	Excel (xls) (zip)
REPOSITORY	http://iehrgitlab.ds.unipi.gr/interopehrate/health-tools/knowledge-mgmt-tool.git (Import Files/LOINC)

Table 6 - LOINC Standard

MAPPING NAME	FHIR Formal Ontology Model
IMPORT FILE FORMAT	OWL
REPOSITORY	http://iehrgitlab.ds.unipi.gr/interopehrate/health-tools/knowledge-mgmt-tool.git (Etypes/iehr-owl-09012020.owl)

Table 7 - FHIR Ontology Model

2.3 Mappings to Pivot Knowledge

STANDARD NAME	ICD-9-to-10 mappings
IMPORT FILE FORMAT	CSV
REPOSITORY	http://iehrgitlab.ds.unipi.gr/interopehrate/health-tools/knowledge-mgmt-tool.git (Import Files/ICD)

Table 8 - ICD-9-to-10 Mappings

MODEL NAME	CDA to FHIR Data Mapping Model
MODEL FILE FORMAT	TTL
REPOSITORY	http://iehrgitlab.ds.unipi.gr/interopehrate/health-tools/data-mapping-tool.git (Models/cda_result.xml-model_final.ttl)

Table 9 - CDA to FHIR Data Mapping Model

3. OVERVIEW

This deliverable demonstrates the configuration and use of data conversion operations within the InteropEHRate Health Services (IHS) deployed at a healthcare institution. As it is described in detail in the specification deliverables [D5.8] (Data Integration Platform) and [D5.10] (Data Mapping and Conversion), the conversion process is controlled both by Platform software components, responsible for building a knowledge graph as a basis for interoperable data, and by formal domain knowledge that is defined and uploaded into the Platform according to the local practices and needs of the healthcare institution.

Accordingly, there are three deliverables on the demonstration of the generation of interoperable data by the IHS:

- [D5.17] is the demonstrator of the Platform and the IHS software components;
- the current D5.19 is the demonstrator related to setting up the formal knowledge that governs data mappings and conversions, through the use of InteropEHRate Health Tools;
- [D5.21] is the demonstrator on data translation and information extraction.

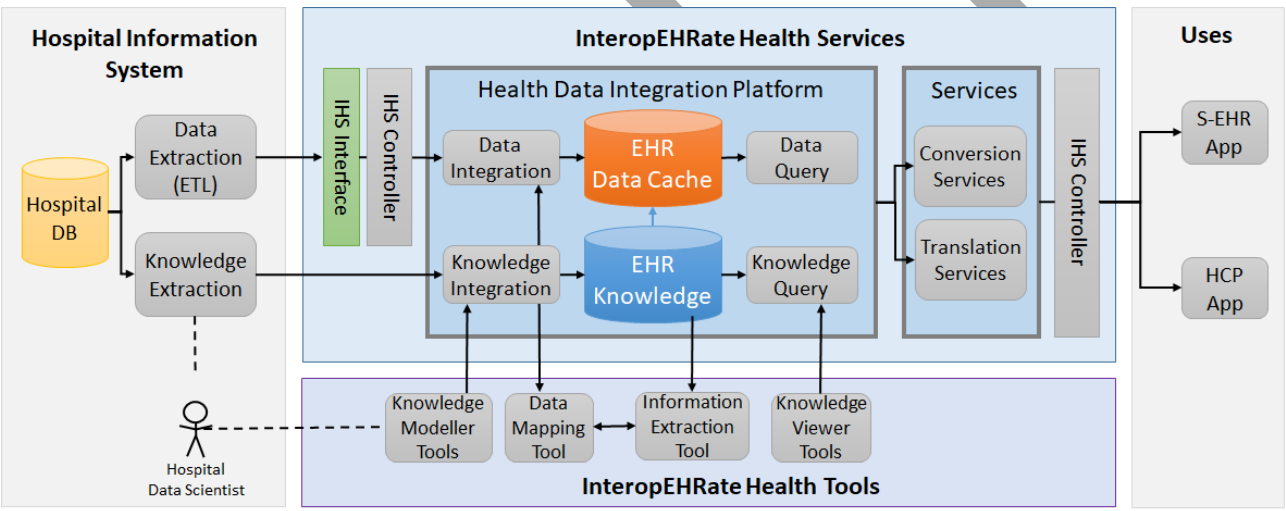

Figure 1 – Architecture of the InteropEHRate Health Services, and within it the Data Conversion and Translation Services and the corresponding interactive tools that are covered by this deliverable

By "data mapping and conversion" the following operations are understood:

- the mapping of health record data structures, as provided by a hospital, to the interoperable FHIR-based data structures adopted by InteropEHRate;
- the conversion of the data values underlying the schemas, due to the requirements of the target schema or the target terminologies used for interoperability.

As detailed in [D5.10], the InteropEHRate mapping and conversion process is mainly *knowledge-driven*, meaning that most of the mapping and conversion operations are defined as declarative knowledge as opposed to procedural business logic. Such knowledge involves:

- target FHIR-based data schemas;

- source domain terms and coded values;
- target domain terms and coded values;
- the mappings between source and target terms and codes;
- the mappings between source and target data schemas;
- the conversion operations to be applied to data values.

In summary, the data mapping and conversion process involves (1) the formal description of all types of knowledge above and their importing into the Data Integration Platform; and subsequently (2) the actual conversion and mapping operations initiated through the InteropEHRate Health Services which, in turn, translate the high-level requests sent by hospitals into low-level calls to the Platform. This latter process is described in detail in [D5.8] and is demonstrated in [D5.17].

The preparation phase (1) above, instead, involves the use of interactive and batch tools for knowledge management that help in working with knowledge efficiently. The following tools are offered and are described in this deliverable:

- a suite of interactive **Knowledge Management Tools** including:
  - an interactive **Entity Type Modeller** that helps in defining target (pivot) data schemas,
  - an interactive **Knowledge Explorer** that allows the navigation of terminological knowledge for validation and information purposes,
  - an interactive **Entity Explorer** that allows the browsing of health data records (entities) that have been imported into the Data Integration Platform, for validation purposes;
- a batch **Knowledge Importer** tool that allows an automatable, spreadsheet-based importing of large amounts of terminological knowledge (terms, coded values) and their mappings;
- an interactive **Data Mapper Tool** that allows the definition of schema mappings and data value conversions that can then be reapplied in a fully automated manner.

The tools are released as part of the *InteropEHRate Health Tools* macro-component, itself described within the Platform deliverable [D5.8]. The table below provides the list of operations supported by the tools.

Requirement	Software Component
Batch definition and importing of terms and codes, both local and international	Knowledge Importer
Review of terms and codes imported, and their mappings	Knowledge Explorer
Interactive design and importing of (FHIR) data schemas	Entity Type Modeller
Interactive and programmatic (batch) conversion of local EHRs to the FHIR-based SEHR format	Data Mapper Tool
Review of automatically imported EHRs	Entity Explorer

Table 10 - Implementation status of conversion services and tools

A functional first version of the Platform, including the InteropEHRate Health Tools, was deployed and is running on the IEHR project server dedicated to data integration and conversion. This platform instance has been populated with health knowledge and is ready to accept EHR integration requests. Both the knowledge and integrated EHRs can be queried and searched through the Knowledge and Data Query components, and can be visualised using the Knowledge Viewer and Data Viewer tools.

4. THE KNOWLEDGE MANAGEMENT SUITE OF TOOLS

The Knowledge Management Suite is a set of interactive tools designed to make the manipulation (editing and viewing) of knowledge easier to the user, namely the Data Scientist typically employed by the hospital where the InteropEHRate Health Tools are deployed. It consists of a *Knowledge Explorer*, an *Entity Type Modeller*, an *Entity Explorer*, as well as a graphical front-end for the batch *Knowledge Importer* (described in section 5). The list of tools can be seen on the left-hand side of figures 2 and 4. The building and installation is done only once for the entire suite, as detailed in the sections below. We then provide separate user's guide sections for each tool. While the detailed description of the usage of each tool is beyond the scope of this document, we provide succinct descriptions of the functionalities provided by them.

4.1 Building guide

The Knowledge Management Suite is implemented as a component running in a *docker container*, so an instance of *docker* is needed in order to create the containers following the instructions of the *docker-compose* file. Furthermore, this component also needs the directory which includes the code for each modeler and explorer tool applet. This folder will be addressed by the instructions inside the docker-compose file. In order to guarantee the correct installation of each applet, the user has to verify that the *docker-compose* file and the above-mentioned folder are located in the same path.

4.2 Installation guide

Once the correct deployment settings (URL and available ports) have been provided in the docker-compose file, together with the files needed for the web application, in order to install this component it is only necessary to execute the following command:

```
docker-compose up
```

After that, the service for this component will be available on the *localhost* machine, over the port specified in the docker-compose file for this service.

4.3 User guide: Knowledge Explorer

The Knowledge Explorer consists of the following elements.

- A **search bar** on the top of the screen that lets the user type in terms and codes to be searched inside the knowledge base inside the Data Integration Platform. The possible elements corresponding to the query are looked up in real time while typing. Note that the terms need to be typed in in the language set in the tool.
- A **language setting** in the top right-hand corner of the screen: this sets both the query language and the language in which the knowledge is displayed (as the Platform natively supports multilingual knowledge).
- A **browsing interface** taking up the largest bottom part of the screen. The interface displays the search results inside multiple tabs:
    - a **Glossary tab** that provides the supra-lingual concept corresponding to the query result, as well its translation into the currently selected language;

- a **Relations tab** that puts the current concept into the context of a browseable knowledge hierarchy (more general and more specific concepts);
- a **Provenance tab** that provides provenance information for the concept, defined as part of the knowledge importing procedure.

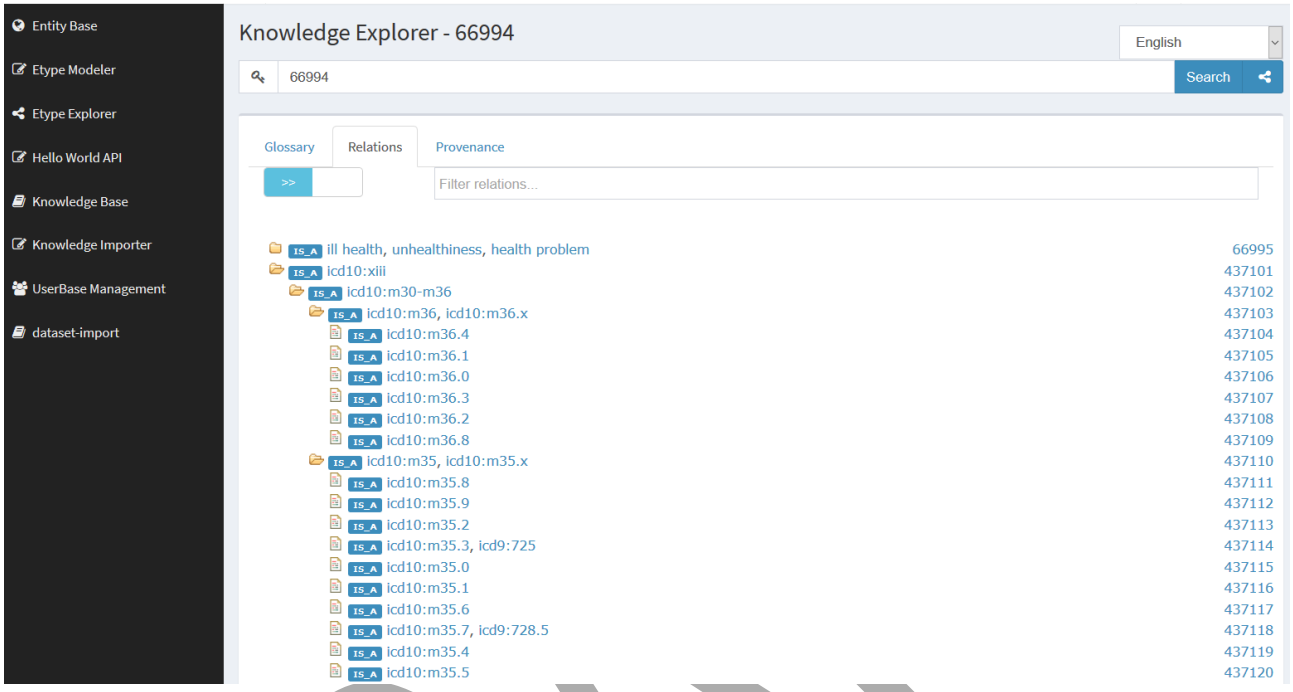

Figure 2 – Screenshot of the Knowledge Explorer Tool

4.4 User guide: Entity Type Modeller

The Entity Type Modeller is an interactive tool that lets the user create data schemas---called *entity types*---through a graphical interface. An *entity type* consists of:

- the **concept of the entity type** that describes its meaning: for example, for an entity type describing medics, the concept is the concept of *Medic*; as both concepts and entity types are organised into a hierarchy, the entity type of *Medic* automatically inherits all attributes from its parent entity type *Healthcare Provider*;
- a **set of attributes** that describe the person (name, date of birth, place of birth, etc.), each of them formally described by the corresponding concept and given a datatype;
- one or more **identifying sets of attributes**: a subset of the attributes above that univocally designate an individual instance. For example, each *person* (individual of the entity type Person) is uniquely identified by the identifying set {name, dateOfBirth, placeOfBirth, nameOfParent} and also by another identifying set {nationalIDNumber};
- **metadata**, such as provenance or category.

A very important prerequisite for using the Entity Type Modeller is thus to have the concepts (and the natural language labels describing those concepts) already imported into the Platform. In other words, the creation and editing of entity types always *follows* the creation of concepts and language labels.

Figure 3 shows a screenshot from the tool. The top left-hand-side window defines a new entity type including its concept. The top right-hand-side window allows the addition of new attributes, and the identifying sets to which they belong. The bottom window, finally, allows the editing of attributes already created.

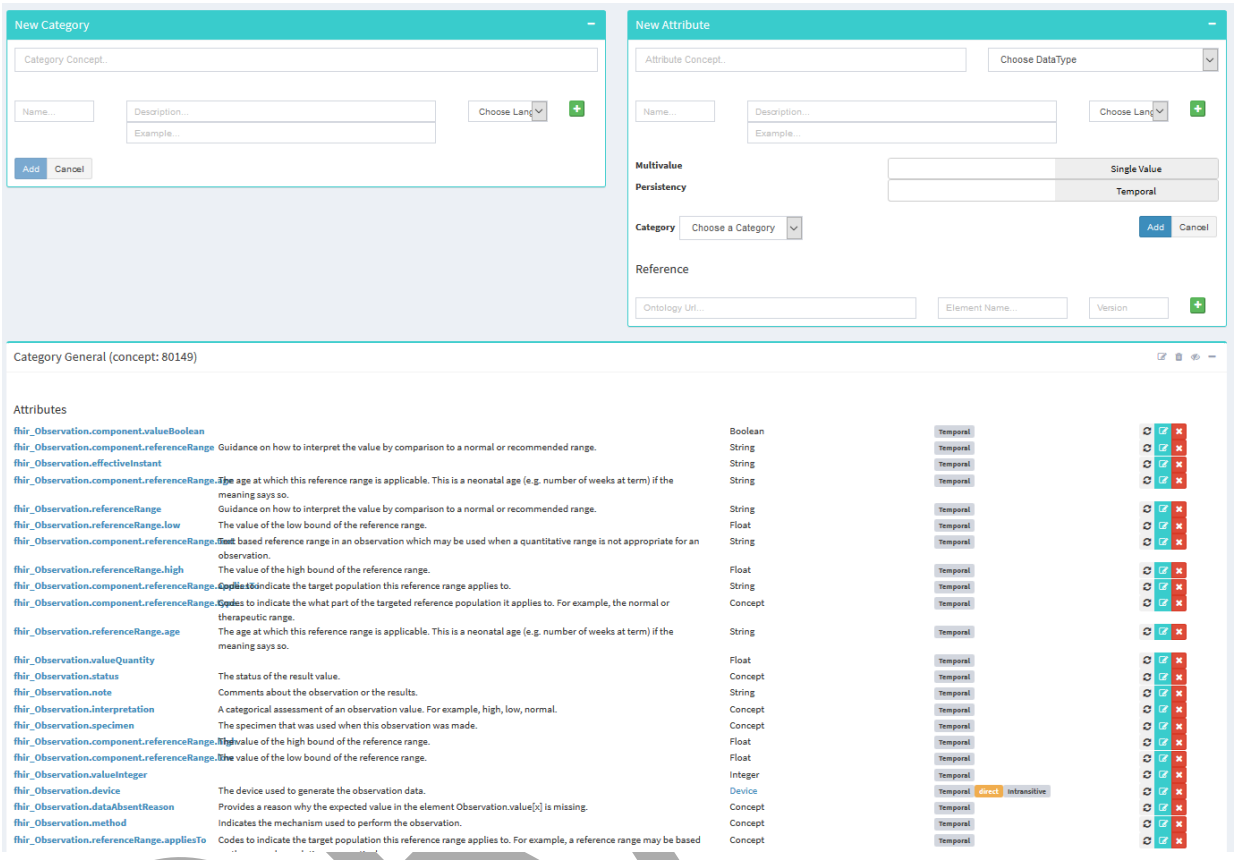

Figure 3 – Screenshot from the Entity Type Modeller tool

4.5 User guide: Entity Explorer

The Entity Explorer allows the browsing of entities---actual data instances or "records"---describing patients, visits, prescriptions, etc., already imported into the Data Integration Platform. Imported entities are represented inside the Platform as a knowledge graph, structured and unstructured data having been converted into a graph structure, and informal textual labels having been converted into supra-lingual concepts or other formal datatypes. As most of the conversion operations are done automatically, the tool allows the verification of the end result: whether the automated methods functioned correctly. In the negative case, the knowledge governing the conversion process needs to be updated through the use of the Knowledge Modeller tools, in an iterative manner.

The Entity Explorer consists of three windows:

- a **search window** (top) that allows the lookup of specific entities (data instances) by defining a complex query involving a combination of attribute values and operators;

- an **entity type browser** (bottom left-hand-side) that allows the browsing of all entities belonging to a given entity type, by selecting the entity type from the entity type hierarchy;
- an **entity browser** (bottom right-hand-side) which displays the list of entities corresponding to the criteria set in the other two windows, and where each entity can be explored individually by displaying its attributes and following its relations to other entities as hyperlinks.

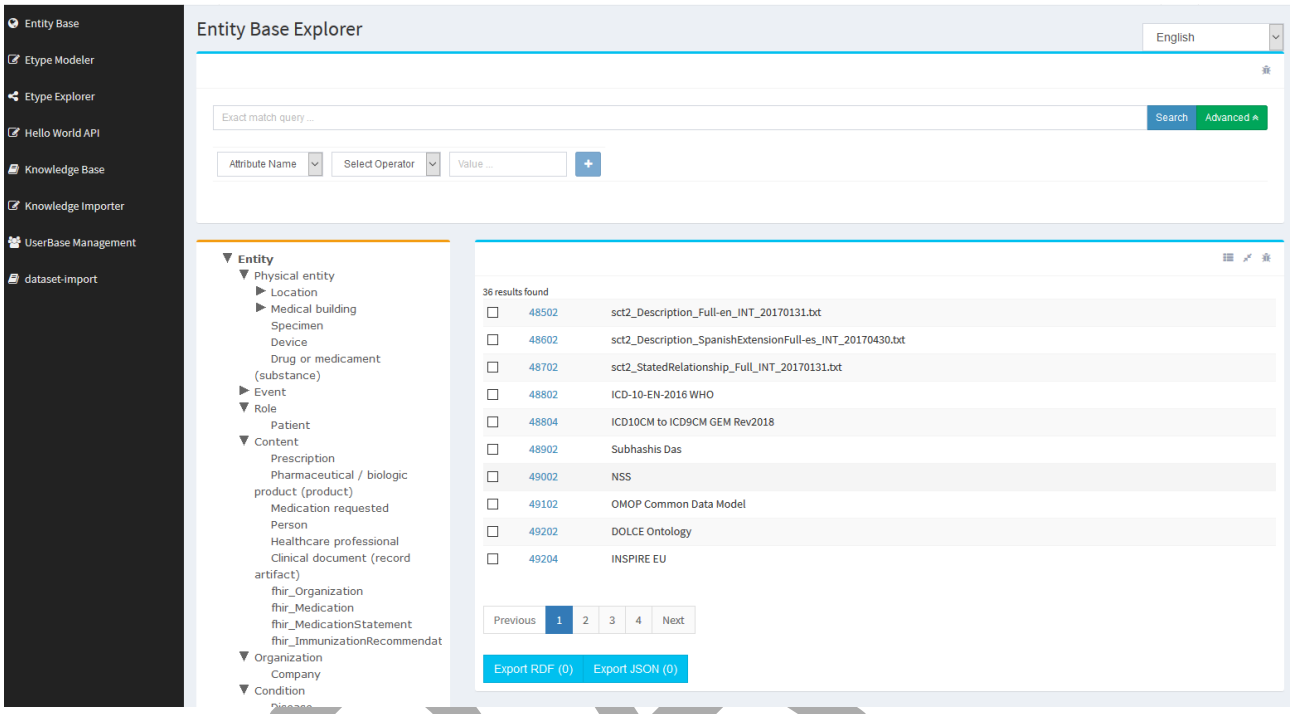

Figure 4 – Screenshot of the Entity Explorer Tool

5. KNOWLEDGE IMPORTER

The Knowledge Importer is a batch tool for importing formal concepts describing the meaning of healthcare terms and codes, as well as natural-language labels in multiple languages that express the terms and codes in multiple languages and standards. The tool imports knowledge described inside Excel spreadsheets following a specific format described in [D5.10].

5.1 Building and installation guide

The Knowledge Importer is built and installed as an integral part of the Data Integration Platform, therefore no specific building and installation operations are needed. The building and installation of the Platform demo itself is described in [D5.17].

5.2 User guide

The spreadsheet containing the knowledge to be imported can be fed to the Knowledge Importer through the command line as follows:

```
/appassembler/bin/updateuk -i import-file.xls
```

Where *import-file.xls* is the spreadsheet file containing the knowledge. The importing process involves the (syntactic and semantic) validation of the data and any potential error is displayed in the command line. Note that the above command, launched in the main Platform folder, executes a specific component called “updateuk”, located in the directory “/appassembler/bin/” that is provided as part of the Data Integration Platform .

Alternatively, the importer can also be run graphically through a simple front-end inside the Knowledge Management Tools.

6. THE DATA MAPPER TOOL

The role of the Data Mapper Tool is to allow the user to define the mappings between source (hospital) and the entity types corresponding to the target (pivot) health record data schemas, schema attributes, as well as the necessary conversions of the underlying attribute values. After having defined the mapping and conversion rules manually, the tool records all rules and is able to replay them in a fully automated manner over a large number of health records.

It is assumed that the tool is used once all knowledge describing the target data schemas as well as the key terminology and codes appearing in the source and/or target data have previously been imported into the Platform using the tools described in sections 4 and 5 above.

6.1 Building guide

The data mapper and viewer tool is a component running in a docker container, so an instance of docker is needed in order to create the containers following the instructions of the docker-compose file.

6.2 Installation guide

Having the correct setting (the URL and ports for accessing the tool) in the docker-compose file, to install this component it is only necessary to execute the following command:

```
docker-compose up
```

After that, the service for this component will be available on *localhost*, on the port specified in the docker-compose file for this service.

The tool also requires that the target (FHIR) entity types should be placed in the following directory (consider the path indicated starting from the IHS [D5.8] installation directory) as an OWL file:

```
/resources/schema/
```

In order to obtain the entity types in OWL---already inside the Data Integration Platform---they need to be exported from the Platform using its `GET data/exportTypesRDF` endpoint, as described in [D5.8].

6.3 User guide

The complexity of the Data Mapping Tool prevents us from providing a complete user guide here. A screenshot is shown in Figure 5. The usage of the tool is divided into the following phases, described in more detail in [D5.10]:

2. importing of the source data to be mapped and of the target data schemas;
3. defining the mapping between the source and target schemas;
4. defining the data value conversion operations, including information extraction;
5. saving the mapping and conversion model (the “recipe”).

6.3.1 Data Importing

The Data Mapper Tool allows the mapping of input data in CSV (tabular), JSON, and XML formats. The preferred format from a usability point of view is CSV, or the other formats with shallow nesting (ideally no more than 3-4 levels). The target data schemas must be imported in OWL format, exported from the Platform as described above.

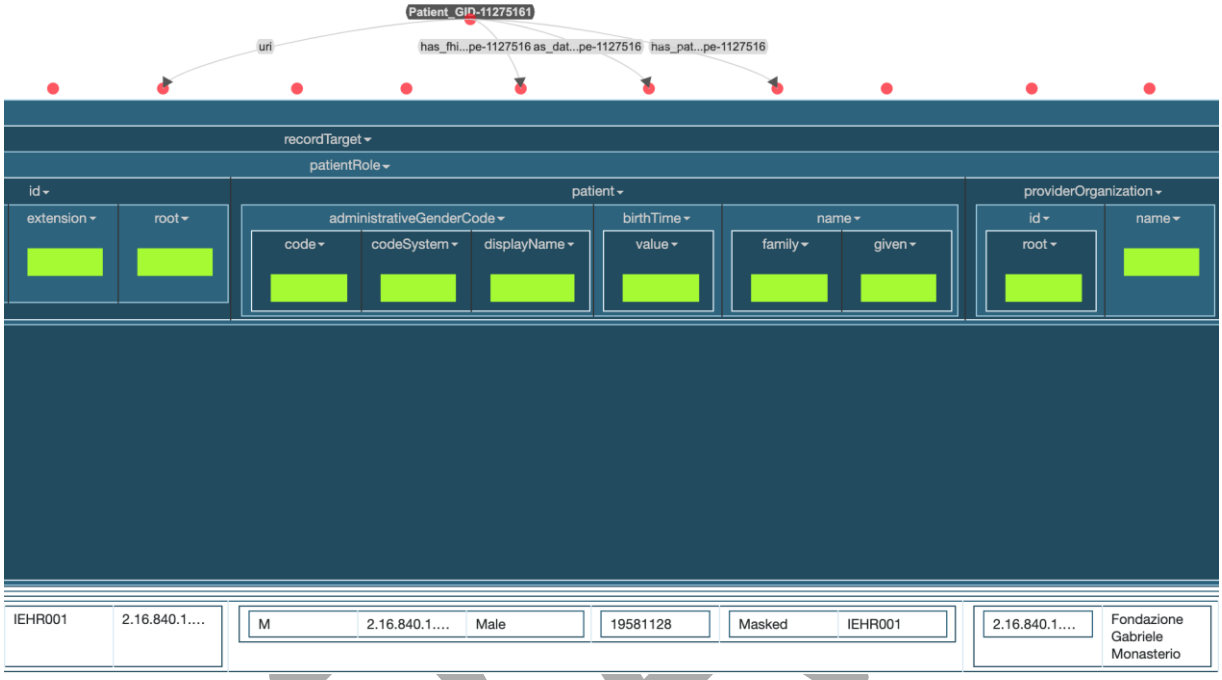

Figure 5 – Screenshot of the Data Mapper Tool

6.3.2 Schema Mapping

Schema mapping is represented on the top of Figure 5. For every data attribute (represented as blue columns under the red dots) that needs to be mapped to FHIR, a mapping towards the corresponding FHIR attribute(s) (represented as labelled red dots) must be specified. The tool supports one-to-many and many-to-one mappings: of course, these imply a subsequent data value conversion rule for the splitting or merging of data values, respectively. The tool maps one source data file at a time. In case data from multiple files need to be "fused" (e.g. patient data is assembled from multiple sources), the mapping needs to be done separately for each source, and the connection between data records will be assured automatically by the Platform provided that the individuals can be correctly identified by their identifying attributes (e.g. the same patient ID is contained in each source file and is mapped correctly).

6.3.3 Data Value Conversion

The conversion rules for the data values of each attribute are defined by clicking on the header of each source data column (represented as blue boxes). Simple operations such as character-based split or merge can be defined interactively. More complex operations are defined through snippets of Python code that can combine data values from the entire source dataset. For example, the conversion of heterogeneous date representations can be converted to the ISO standard representation with a single line of Python code.

The most unique feature of the Data Mapper Tool is its ability to parse natural-language text inside structured data and convert it into a supra-lingual concept-based representation. This is launched through a data conversion dialogue box as any other conversion operation (left-hand-side of FiIgure 6). The result is obtained as concepts linked to the terms automatically annotated by the tool inside the column (right-hand-side of Figure 6). Clicking on a concept opens up its detailed definition inside the Knowledge Explorer Tool.

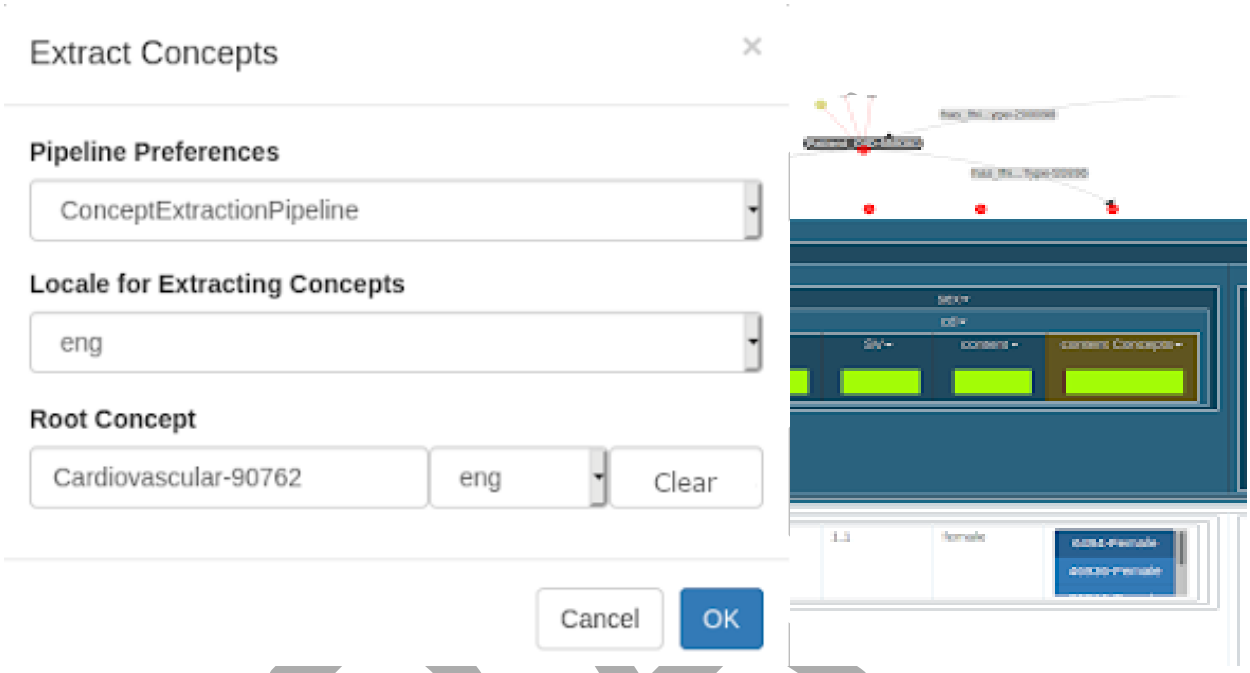

Figure 6 – Information Extraction control dialogue box (left) and the result obtained (right)

6.3.4 Model Export

At the end of the manual mapping and conversion process, the list of operations (that are editable inside the tool) need to be saved by the user. This list of operations is called “mapping model” or, informally, “recipe”. In order to automate the mapping process in production, the path to the file containing the model then needs to be specified as a configuration parameter to the InteropEHRate Health Services component, as described in [D5.8].

7. KNOWLEDGE RESOURCES

In this section we provide details on the knowledge resources used inside the InteropEHRate project. We distinguish between, on the one hand, *pivot knowledge* that refers to the international standards used for interoperability and, on the other hand, *local knowledge* and its mappings to the pivot knowledge that refer to the way health records are described inside hospitals.

7.1 Pivot Knowledge

By "pivot knowledge" we mean the formal representation of international healthcare standards that are part of the Interoperability Profile [D2.8]. By "implementation" we mean:

- conversion of these standards from the formats in which they are distributed to the formats expected by the InteropEHRate Data Integration Platform;
- archival of such converted resources for future reuse, including documentation;
- importing of the standards into working instances of the InteropEHRate Health Services.

Furthermore, whenever available, pivot knowledge needs to be implemented in five languages, corresponding to the four hospitals participating in the InteropEHRate project (Italian, French, Romanian, Greek) plus English as a lingua franca.

The table below shows the elements of pivot knowledge used in the pilot studies of the InteropEHRate project and that, consequently, are available for demo purposes at the time of writing this deliverable. All knowledge is currently found under the InteropEHRate GitLab project: http://iehrgitlab.ds.unipi.gr/interopehrate/health-tools/knowledge-mgmt-tool.git

The list of supported resources will be extended with new resources and new languages as support for them is further developed.

Standard	Use	Size	Status	Language support	Where to find
SNOMED CT International	reference terminology	1.5 million labels	full 2017 version imported	English, Spanish, supra-lingual representation	Import Files/SNOMED
ICD-10	disease codes	69,000 labels	WHO version imported	English, Italian, supra-lingual representation	Import Files/ICD
LOINC	lab observations	91,388 labels	fully imported	English, supra-lingual representation	Import Files/LOINC
WHO ATC	drugs	>13476 labels	to be done	English, Italian, supra-lingual representation	Import Files/ATC
FHIR 4.0 IPS	EHR schemas for International	458 attributes	90% coverage	English, supra-lingual	Import Files/FHIR

	Patient Summary			representation	
UCUM	unity of measure codes	711 attributes	fully imported	English, supra-lingual representation	Import Files/UCUM

Table 11 - Pivot knowledge implemented in the project

7.2 Local Knowledge and its Mappings

Local health records from the two hospitals providing source health data to the project, namely FTGM Pisa (Italy) and CHU Liège (Belgium), including the terminology and codes used within, are being mapped to the international standards above at the time of writing this document. The table below provides details on the kinds of mapping knowledge already available or in progress of being developed.

Local Standard	Local standard user	Pivot Standard	Status	Language support	Where to find
ICD-9	FTGM (Italy)	ICD-10	only 1-to-1 equivalence mappings that are covered by ICD-10	From Italian towards English, supra-lingual representation	Import Files/ICD
Patient Summary XML CDA	FTGM (Italy)	FHIR 4.0 IPS	mapping only covers the contents of the example file provided by FTGM	From Italian towards English, supra-lingual representation	http://iehrgitlab.ds.unipi.gr/interopehrate/health-tools/data-mapping-tool.git (Models/cda_result.xml-model_final.ttl)
SumEHR Patient Summary	CHU (Belgium)	FHIR 4.0 IPS	TBD	From French towards English, supra-lingual representation	
SumEHR local terminology	CHU (Belgium)	LOINC	mapping covers the CHU local standard codes used in Patient summary	From French towards English, supra-lingual representation	import Files/CHUlocalTerms
XML CDA Laboratory Results	FTGM (Italy)	FHIR 4.0 IPS	mapping only covers the contents of the example file provided by FTGM	From Italian towards English, supra-lingual representation	http://iehrgitlab.ds.unipi.gr/interopehrate/health-tools/data-mapping-tool.git (Models/CDALabResult2FHIR.ttl)
XML SumEHR Laboratory	CHU (Belgium)	FHIR 4.0 IPS	TBD	From French towards	TBD

Results				English, supra-lingual representation	
XML SumEHR Prescriptions	CHU (Belgium)	FHIR 4.0 IPS	TBD	From French towards English, supra-lingual representation	TBD
JSON CHU Local Format Echocardiogram	CHU (Belgium)	FHIR 4.0 IPS	TBD	From French towards English, supra-lingual representation	TBD
DICOM x-ray	CHU (Belgium)	TBD	TBD	From French towards English, supra-lingual representation	TBD

Table 12 - Mapping knowledge implemented in the project

REFERENCES

[D2.3] InteropEHRate Consortium, User Requirements for cross-border HR integration - V3, 2021. https://www.interopehrate.eu/resources/

[D2.8] InteropEHRate Consortium, FHIR Profile for EHR interoperability - V2, 2020. https://www.interopehrate.eu/resources/

[D5.8] InteropEHRate Consortium, Design of the Health Data Integration Platform - V2, 2021. https://www.interopehrate.eu/resources/

[D5.10]: InteropEHRate Consortium, Design of the Data Mapper and Converter to FHIR - V2, 2021. https://www.interopehrate.eu/resources/

[D5.17]: InteropEHRate Consortium, Data integration platform for healthcare professionals – v2, 2021.
https://www.interopehrate.eu/resources/

[D5.21]: InteropEHRate Consortium, Data integration platform for healthcare professionals – v2, 2022.
https://www.interopehrate.eu/resources/