D5.16

Data integration platform for healthcare professionals – v1

ABSTRACT

One of the main objectives the InteropEHRate project is to be open and incremental: based on open specifications, connecting for-profit and no-profit data providers with different levels of interoperability, starting from a low level for secure exchange of unconverted data, to a high level by combining knowledge extraction and adaptive data integration, to translate data to a common HL7 FHIR profile and into the natural language of the consumer; the back-end platform (infrastructure and fundamental services) that support the integration and interoperability methods defined in Data mapping and conversion to FHIR as well as Information extraction and translation. The platform includes healthcare knowledge and related tools organised into linguistic (NLP), terminological, and ontological layers, formally representing lexical units, schemas and encoding standards used by member countries, as well as mappings between these wherever applicable. This document accompanies the release of the software components of the Platform.

Delivery Date	January 21st, 2020
Work Package	WP5
Task	T5.5
Dissemination Level	Public
Type of Deliverable	Demonstrator
Lead partner	SIVECO

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.

This work by Parties of the InteropEHRate Consortium is licensed under a Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/).

CONTRIBUTORS

	Name	Partner
Contributors	Gábor Bella	UNITN
Contributors	Simone Bocca	UNITN
Contributors	Andrei OANCA	SIVECO
Contributors	Simona BICA	SIVECO
Contributors	Simona ALEXANDRU	SIVECO
Reviewers	Adrian Bradu	SIVECO

LOGTABLE

Version	Date	Change	Author	Partner
0.1	12-11-19	Creation of the document's template (TOC)	Andrei OANCA	SIVECO
0.2	18-11-19	Added text	Simona ALEXANDRU	SIVECO
0.3	18-11-19	Revisions and added text	Simona BICA	SIVECO
0.4	21-11- 19	Revisions and added text	Nicu JALBA	SIVECO
0.5	23-12-19	Revision and quality check	Adrian BRADU	SIVECO
0.6	09-01-20	Restructuring and extension to cover all progress made on the platform	Gábor BELLA, Simone BOCCA	UNITN
0.7	09-01-20	Quality Review	Argyro Mavrogiorgou	UPRC
1.0	21-01-20	Fixed document structure to be more conform to template	Gábor BELLA, Simone BOCCA	UNITN
VFinal	21-01-20	Final version for submission	Laura Pucci	ENG

ACRONYMS

Acronym	Term and definition
EHR	Electronic Health Record
JDK	Java Development Kit
HCP	Health Care Professional
REST	REpresentational State Transfer
HTTP	Hypertext Transfer Protocol
JSON	JavaScript Object Notation
SEHR	Smart Electronic Health Record
FHIR	A standard providing data schemas for healthcare data exchange.
ICD	International Classification of Diseases, an international standard.
LOINC	An international standard (set of identifiers, names, and codes) for identifying health measurements, observations, and documents.
SNOMED CT	SNOMED Clinical Terms is a systematically organized computer processable collection of medical terms providing codes, terms, synonyms and definitions used in clinical documentation and reporting.
WHO ATC	The World Health Organization's classification system for active substances of pharmaceutical products. The active substances are classified in a hierarchy with five different levels.

TABLE OF CONTENT

1. INTRODUCTION 1
1.1. Scope of the document 1
1.2. Intended audience 1
1.3. Structure of the document 1
1.4. Updates with respect to previous version (if any) 1
2. SW DESCRIPTION 2
2.1. Concept Translation Library for Knowledge Query 2
2.2. Health Knowledge Converters for Knowledge Integration 2
2.3. Terminal IHS HR Exchange 2
2.4. Knowledge Modeller and Viewer Tools 3
2.5. Data Mapper and Viewer Tools 3
3. OVERVIEW 4
3.1. Concept Translation Library for Knowledge Query 5
3.1.1. Building guide 5
3.1.2. Installation guide 5
3.1.3. User guide 5
3.2. Health Knowledge Converters for Knowledge Integration 6
3.2.1. Building guide 6
3.2.2. Installation guide 6
3.2.3. User guide 6
3.3. Terminal IHS HR Exchange 6
3.3.1. Building and installation guide 7
3.3.2. User guide 7
3.4. Knowledge Modeller and Viewer Tools 8
3.4.1. Building guide 8
3.4.2. Installation guide 9
3.4.3. User guide 9
3.5. Data Mapper and Viewer Tools 10
3.5.1. Building guide 10
3.5.2. Installation guide 10
3.5.3. User guide 10

LIST OF FIGURES

Figure 1 – Architecture of the Data Integration Platform

Figure 2 – Output for the Terminal IHS HR

Figure 3 – Screenshot from the Knowledge Modeller Tools

Figure 4 – Screenshot of the Knowledge Viewer Tool

Figure 5 – Screenshot of the Data Mapper Tool

Table 6 – Release information on Data Mapper and Viewer Tools

LIST OF TABLES

Table 1 – Release information on the Concept Translation Library component

Table 2 – Release information on Health Knowledge Converters for Knowledge Integration

Table 3 – Release information on Terminal IHS HR Exchange

Table 4 – Release information on Terminal IHS HR Exchange Tests

Table 5 – Release information on Knowledge Modeller and Viewer Tools

Table 6 – Release information on Data Mapper and Viewer Tools

1. INTRODUCTION

1.1. Scope of the document

This document provides details about the first version of Data integration platform for healthcare professionals. The details refer to:

- a general overview of the components that compose the Platform;
- the Platform components modified, newly developed, or deployed as is;
- the functionalities that have been implemented;
- release modalities;
- screenshots and an overview of each component;
- next steps.

1.2. Intended audience

The document is intended to all people interested to have an overview of how the integration platform can be implemented or can be used.

1.3. Structure of the document

For each Platform component developed so far, section 2 provides short factual details about the software package, while section 3 provides an overview, as well as building, installation, and succinct user guides.

1.4. Updates with respect to previous version (if any)

Not applicable, this is the first version of the deliverable.

2. SW DESCRIPTION

2.1. Concept Translation Library for Knowledge Query

SW TITLE	Concept Translation Library
SW VERSION	1.0.0
PROGRAMMING LANGUAGES	Node.js
SUPPORTED PLATFORM(s)	Cross Platform
SOURCE CODE	http://iehrgitlab.ds.unipi.gr/interopehrate/health-services/label-translation.git

Table 1 – Release information on the Concept Translation Library component

2.2. Health Knowledge Converters for Knowledge Integration

SW TITLE	Health Knowledge Converters
SW VERSION	1.0.0
PROGRAMMING LANGUAGES	Python
SUPPORTED PLATFORM(s)	Cross Platform
SOURCE CODE	http://iehrgitlab.ds.unipi.gr/interopehrate/health-tools/knowledge-mgmt-tool.git

Table 2 – Release information on Health Knowledge Converters for Knowledge Integration

2.3. Terminal IHS HR Exchange

SW TITLE	Terminal IHS HR Exchange
SW VERSION	1.0.0-SNAPSHOT
LICENCES AND PATENTS	Apache License
PROGRAMMING LANGUAGES	JDK 13
SUPPORTED PLATFORM(s)	Cross Platform
SOURCE CODE	http://iehrgitlab.ds.unipi.gr/interopehrate/reference-hcp-app/terminal-ihs-hr-exchange.git
EXECUTABLE	https://drive.google.com/open?id=1gs4HLOhcFSXdp6Tykr52BDLPKrbfVDVI

Table 3 – Release information on Terminal IHS HR Exchange

SW TITLE	Terminal IHS HR Exchange Tests

SW VERSION	1.0.0-SNAPSHOT
LICENCES AND PATENTS	Apache License
PROGRAMMING LANGUAGES	JDK 13
SUPPORTED PLATFORM(s)	Cross Platform
SOURCE CODE	http://iehrgitlab.ds.unipi.gr/interopehrate/reference-hcp-app/api-test-clients/terminal-ihs-hr-exchange-tests.git
EXECUTABLE	https://drive.google.com/open?id=11FYljgttZrm2vxDxiGJCYRC31DDCBBQQ

Table 4 – Release information on Terminal IHS HR Exchange Tests

2.4. Knowledge Modeller and Viewer Tools

SW TITLE	Knowledge Modeller and Viewer Tools
SW VERSION	1.0.0
LICENCES AND PATENTS	LICENSES OR PATENTS
PROGRAMMING LANGUAGES	Node.js
SUPPORTED PLATFORM(s)	Cross-platform
SOURCE CODE	n/a
EXECUTABLE	n/a

Table 5 – Release information on Knowledge Modeller and Viewer Tools

2.5. Data Mapper and Viewer Tools

SW TITLE	Data Mapper and Viewer Tools
SW VERSION	1.0.0
LICENCES AND PATENTS	LICENSES OR PATENTS
PROGRAMMING LANGUAGES	Java, Node.js
SUPPORTED PLATFORM(s)	Cross-platform
SOURCE CODE	n/a
EXECUTABLE	n/a

Table 6 – Release information on Data Mapper and Viewer Tools

3. OVERVIEW

The role of the Platform is to provide fundamental syntactic and semantic data integration and query functionalities to the *InteropEHRate Health Services* (IHS). Figure 1 below shows the high-level architecture of the Platform.

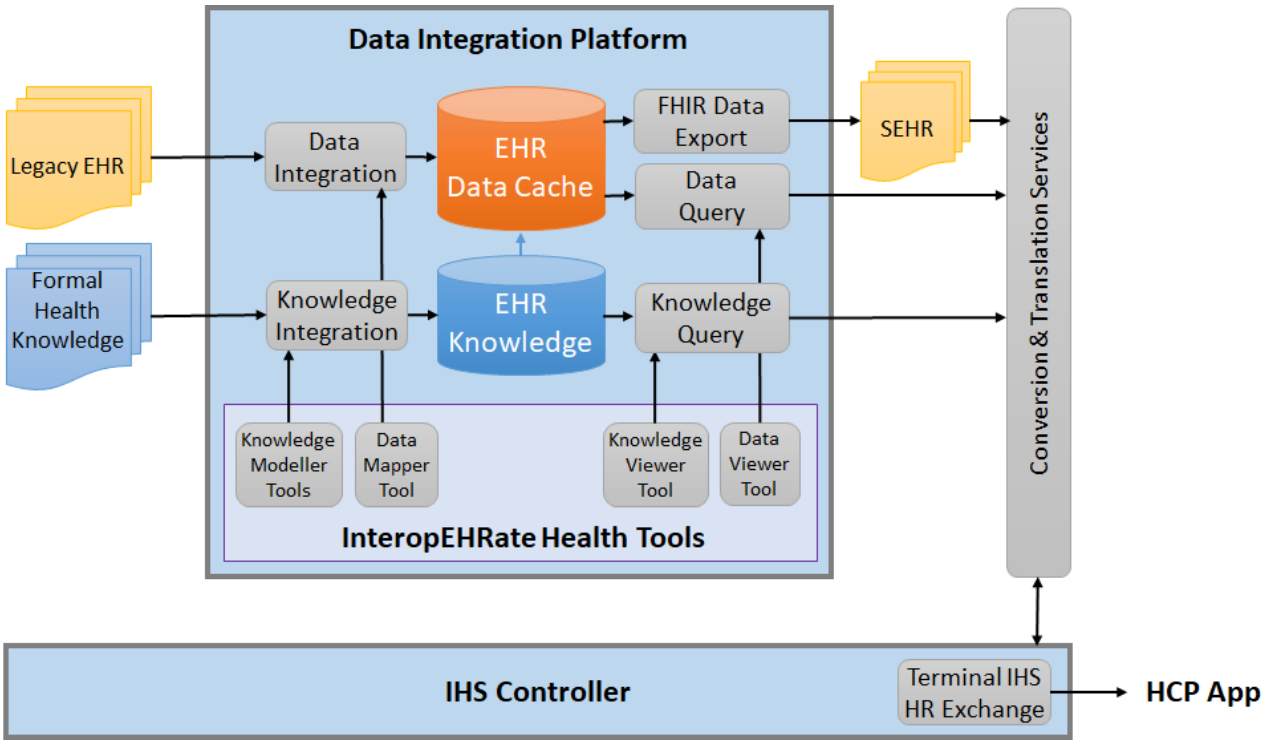

Figure 1 – Architecture of the Data Integration Platform

For more detailed specifications of the Platform, please consult D5.7 [6]. This document provides details on the progress of development with respect to its various components. The "Platform" task (T5.2) and the corresponding deliverables, such as this one, beyond the Data Integration Platform itself strictly speaking, also cover auxiliary components that are responsible for integrating the Platform functionalities with the hospital infrastructure, such as the *IHS Controller* and the *Terminal IHS HR Exchange*.

In terms of development requirements, the components fall into the following categories:

- **to be used as is** (components developed by UniTN before the start of the IEHR project and used by the project without modification):
  - EHR Data Cache DB,
  - EHR Knowledge DB,
  - Knowledge Modeller Tools,
  - Data Integration,
  - Knowledge Viewer Tool,
  - Data Query;
- **to be extended** (components developed by UniTN before the start of the IEHR project and extended within the framework of the project):
  - Knowledge Integration,
  - Knowledge Query,

    - Data Mapper Tool,
    - Data Viewer Tool;
- **new** (components newly developed as part of IEHR):
    - FHIR Data Export,
    - IHS Controller;
    - Terminal IHS HR Exchange.

According to the above, in terms of sharing of results within the project, the following policy is applied:

- used as is: the components are deployed for demonstration but the source code is not shared;
- extended: the components are deployed for demonstration, the source code of the extensions is shared on the project GitLab repository;
- new: the components are deployed for demonstration, the source code is shared on the project GitLab repository.

As of the end of Y1 of the project, the Platform is in a working prototype stage, capable of carrying out and demonstrating the integration of a patient summary, execute schema and terminology mapping functions towards the standards defined by the IEHR Interoperability Profile [2], and provide a graph-based version of the patient summary for searching and query. The Platform also provides access to the health knowledge it stores, that can be used by outside libraries and tools for the conversion and the translation of content.

In Y2, development will continue with the IHS Controller component, that will allow the integration of the Platform into the hospital environment, as well as the FHIR export library that will allow the generation of entire SEHRs in FHIR format.

3.1. Concept Translation Library for Knowledge Query

The translation of concept labels (such as coded value descriptions but also any health domain terminology labels) across languages is a requirement for cross-border EHR interoperability. In order to support this on the platform level, a Concept Translation Library has been developed. It is considered as part of the Knowledge Query component. It will be used as a low-level API service by the higher-level IEHR Translation Services.

3.1.1. Building guide

The concept translation component is a web service developed in Node.js, so the npm JavaScript environment has to be present in the system where the service has to run. In order to have a correct installation of the service is only necessary to download the component's working directory which includes all the files needed for the web service execution.

3.1.2. Installation guide

This component is a web service running in a Node.js environment. In order to switch on the service, it is necessary to run the following command:

```
npm run start
```

After that the web service will be waiting for a request.

3.1.3. User guide

Once the service is running, it is possible to perform a request using the following url:

http://213.249.46.208:8443/translate?lang=target-lang

In the url above, the value assigned to the lang parameter is the translation language for the output. Here a request example using the curl command:

curl -X POST -d @file-to-translate.json http://213.249.46.208:8443/translate?lang=eng

--header 'content-type: application/json'

Where, file-to-translate.json is the content that has to be translated in the correct target language specified as english (eng).

3.2.Health Knowledge Converters for Knowledge Integration

The Platform needs formally to represent massive amounts of health knowledge, in particular terminologies and classifications, among others:

4. ICD 9 and 10 codes and their descriptions in multiple languages (hundreds of thousands of labels);
5. SNOMED CT codes and multilingual labels (millions of labels);
6. LOINC codes and multilingual labels (hundreds of thousands of labels);
7. FHIR data structures (thousands of attributes);
8. WHO ATC drug codes and descriptions.

In order to automate the integration of these healthcare resources, new knowledge integration scripts have been developed. They are considered as extensions to the Knowledge Integration component and their source code is released.

3.2.1. Building guide

The converters are stand-alone Python scripts that do not need any specific building procedure.

3.2.2. Installation guide

The only installation pre-requisite for the scripts is Python 3.0 or above being installed on the machine.

3.2.3. User guide

The scripts can be executed through the command:

python script-name.py

where *script-name* refers to the script corresponding to the type of knowledge being converted (e.g. LOINC or SNOMED).

3.3.Terminal IHS HR Exchange

One of the most important functionalities that are described in the Use Cases of the InteropEHR project is the ability of translating the information stored about the citizen's EHR. This translation includes both language translation and other necessary translation about coding numbers, data registers that are different from country to country.

In response to this request of the InteropEHRate list of user requirements, the Terminal IHS HR Exchange module has been developed. The Terminal IHS HR Exchange is a client of the InteropEHRate Health Services that can be used by the applications that run on HCP terminals. The module was initially designed in order to

be integrated in HCP App considering the requirement to be open for integration into other further applications. This component is part of the Data integration platform for healthcare professionals.

As of December 2019, The Terminal IHS HR Exchange project incorporates a REST service client and able to send HTTP requests having patient summary in JSON format as body. The request contains initially the patient summary in source language and the response contains the patient summary in translated language. In this initial version supports the translation from Italian to English as supported by the IHS and will evolve with the IHS implementation.

The *Terminal IHS HR Exchange Tests* project tests the component *Terminal IHS HR Exchange*. It sends a patient summary in JSON format having the content in Italian language and receives the English version of this patient summary. Both versions are printed to the console. This component was developed because the Terminal IHS HR Exchange was designed to be integrated by other components and doesn't have an “entry point” to be executed.

3.3.1. Building and installation guide

The component is built together with the other components of the Platform. Separate building and installation instructions are thus not available.

3.3.2. User guide

The client can be executed using the command:

```
java -jar terminal-ihs-hr-exchange-test-1.0.0-SNAPSHOT
```

The output will be something similar with the one illustrated in the following picture

```
        "unit": "mEq/L",
        "system": "http://unitsofmeasure.org"
      }
    }
  },
  {
    "resource": {
      "resourceType": "Observation",
      "id": "515fb293-425e-4b91-afc9-a138f6722ccd",
      "meta": {
        "profile": [
          "http://hl7.org/fhir/uv/ips/StructureDefinition/observation-laboratory-uv-ips"
        ]
      },
      "status": "final",
      "code": {
        "coding": [
          {
            "system": "http://loinc.org",
            "code": "2069-3",
            "display": "Chloride Bld-sCnc"
          }
        ]
      },
      "subject": {
        "reference": "Patient/5b302664-99fc-4679-be2e-7ee6d1735c45"
      },
      "effectiveDateTime": "2017-11-17T08:45:00",
      "valueQuantity": {
        "value": 109.1,
        "unit": "mEq/L",
        "system": "http://unitsofmeasure.org"
      }
    }
  },
  {
    "resource": {
```


Figure 2 – Output for the Terminal IHS HR

3.4.Knowledge Modeller and Viewer 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. Currently, the Platform is missing the FHIR Data Export component whose development has just started.

3.4.1. Building guide

The knowledge modeller 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. Furthermore this component also needs the directory which includes the code for each applets of modeler and viewer tool. This folder will be addressed by the instructions inside the docker-compose file.

3.4.2. Installation guide

Having the correct setting (for url and available ports) in the docker-compose file, together with the files needed for the web application, 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 machine, on the port specified in the docker-compose file for this service.

3.4.3. User guide

The final version of the tools will be delivered with a detailed user guide. As the tools have a complex and powerful GUI, its description is beyond the scope of this document. Below are screenshots from the Knowledge Modeller and Viewer tools.

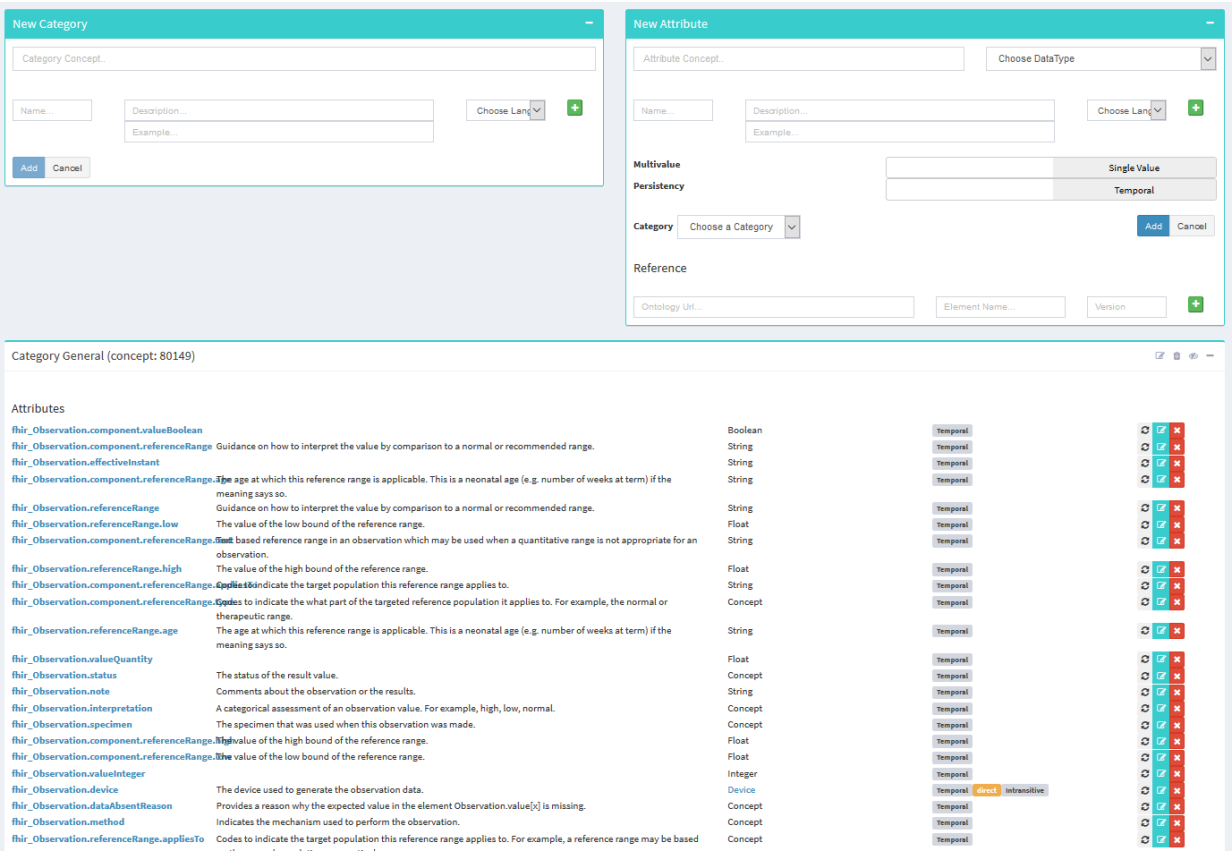

Figure 3 – Screenshot from the Knowledge Modeller Tools

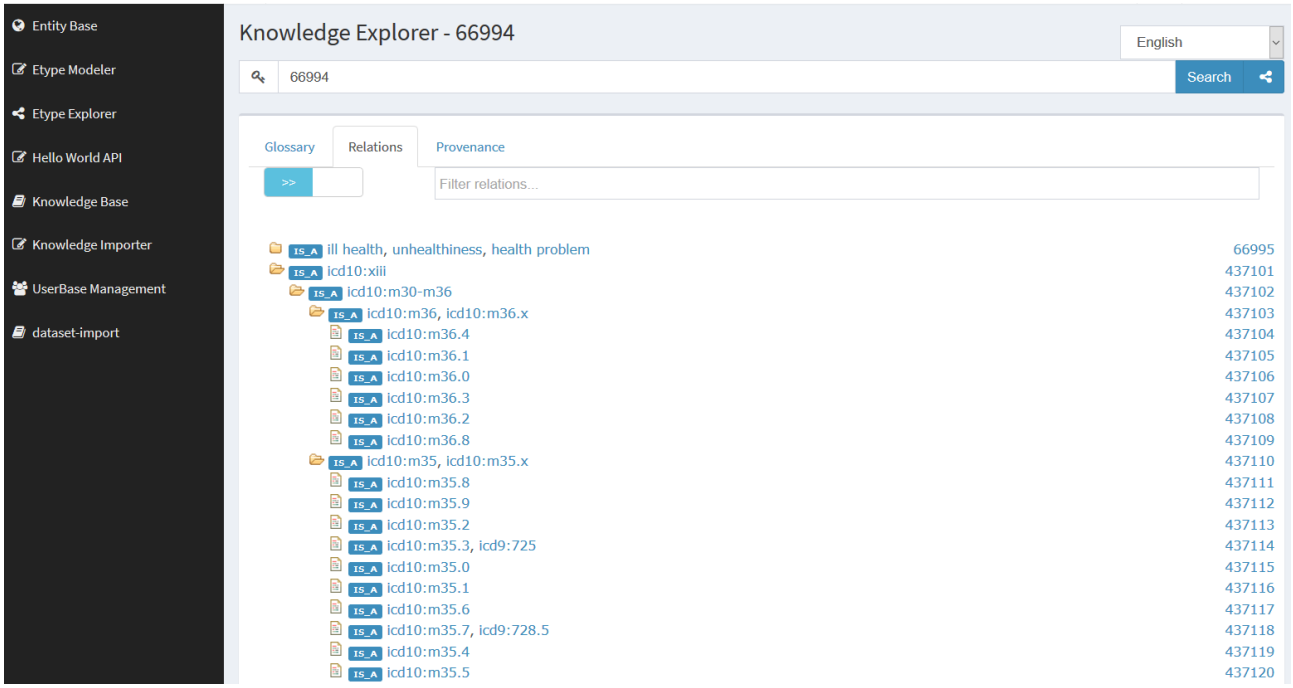

Figure 4 – Screenshot of the Knowledge Viewer Tool

3.5.Data Mapper and Viewer Tools

The Data Mapper and Viewer tools are UI-based tools responsible for defining the mappings between local EHRs and the interoperable SEHR, as well as for displaying the results of the automated conversion according to these mappings for review purposes.

3.5.1. 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.

3.5.2. Installation guide

Having the correct setting (for url and available ports) 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 machine, on the port specified in the docker-compose file for this service.

3.5.3. User guide

The final version of the tools will be delivered with a detailed user guide. As the tools have a complex and powerful GUI, its description is beyond the scope of this document. Below are screenshots from the Data Mapper and Viewer tools.

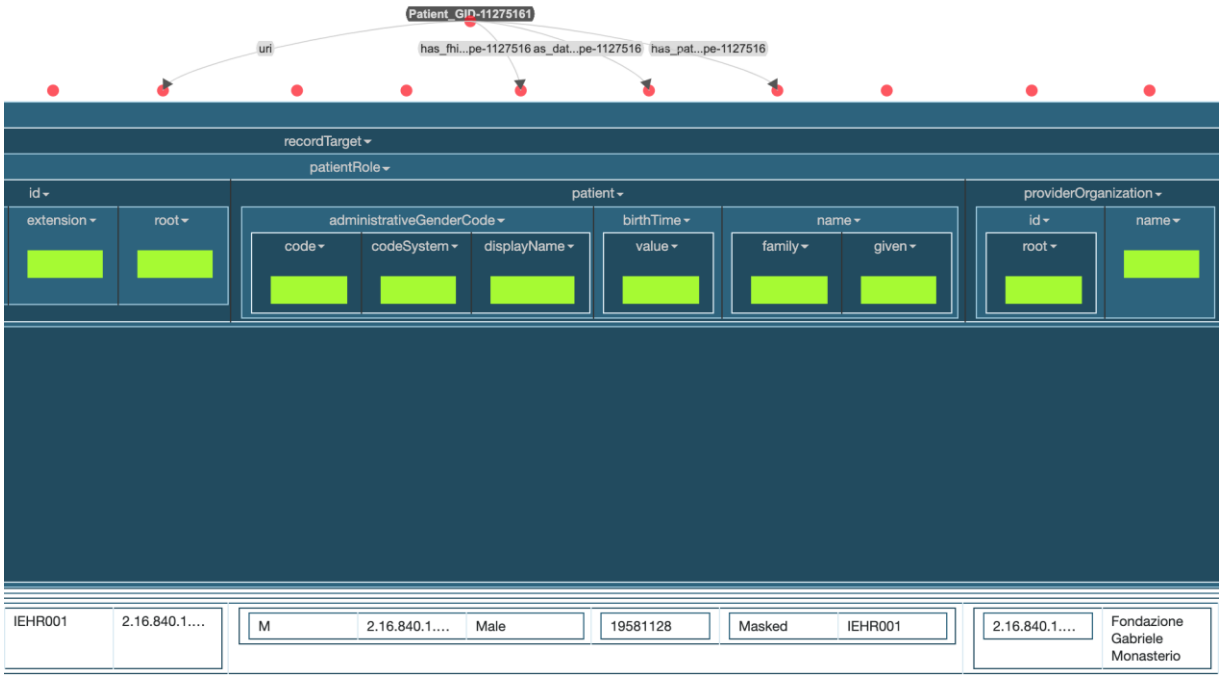

Figure 5 – Screenshot of the Data Mapper Tool

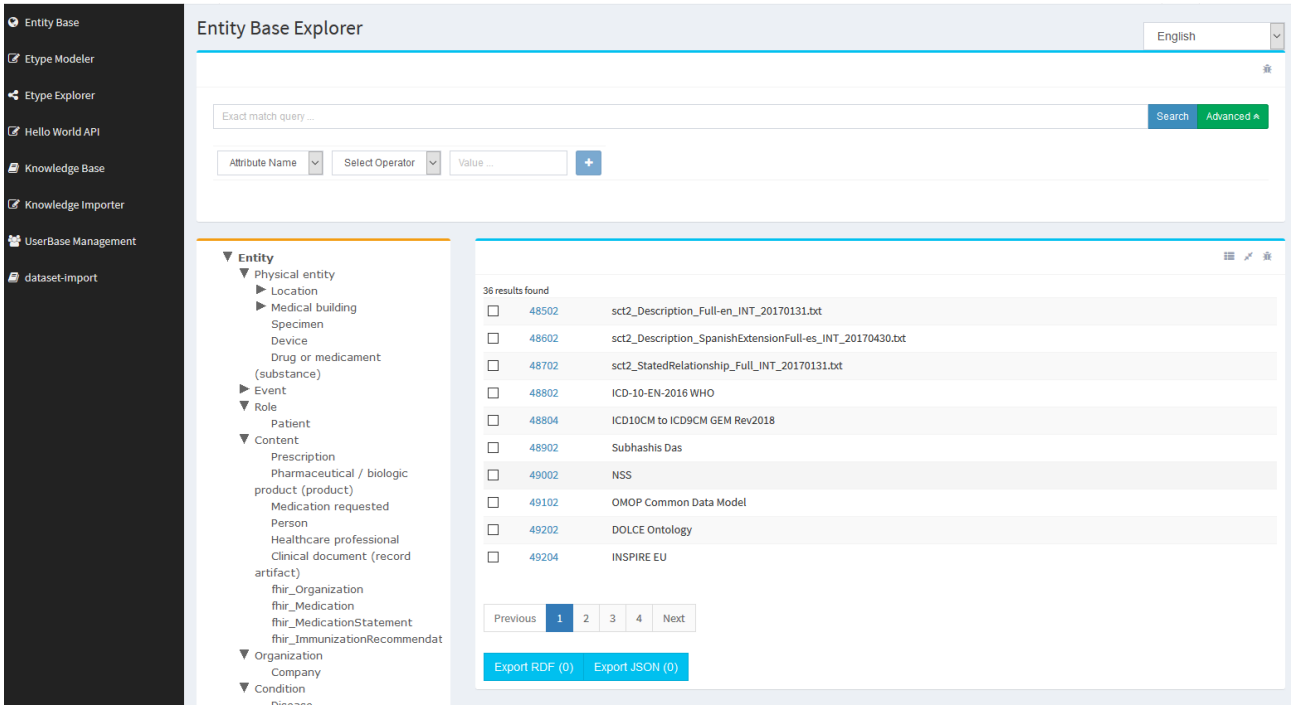

Figure 6 – Screenshot of the Data Viewer Tool

REFERENCES

[1] **[D2.1]** InteropEHRate Consortium, User Requirements for cross-border HR integration - V1, 2019. https://www.interopehrate.eu/resources/

[2] **[D2.7]** InteropEHRate Consortium, FHIR Profile for EHR interoperability - V1, 2019. https://www.interopehrate.eu/resources/

[3] **[D4.4]** InteropEHRate Consortium, Design of libraries for remote D2D HR exchange - V1, 2019. https://www.interopehrate.eu/resources/

[4] **[D5.1]** InteropEHRate Consortium, Software Requirements specification of an integrated EHR web app for HCP - V1, 2019. https://www.interopehrate.eu/resources/.

[5] **[D5.4]** InteropEHRate Consortium, Design of an integrated EHR web app for HCP - V1, 2019. https://www.interopehrate.eu/resources/

[6] **[D5.7]** InteropEHRate Consortium, Design of the Health Data Integration Platform - V1, 2019. https://www.interopehrate.eu/resources/