MetaDataManager - User Manual

Release 2.1.10

ISTAT

Nov 28, 2023

CONTENTS

CHAPTER

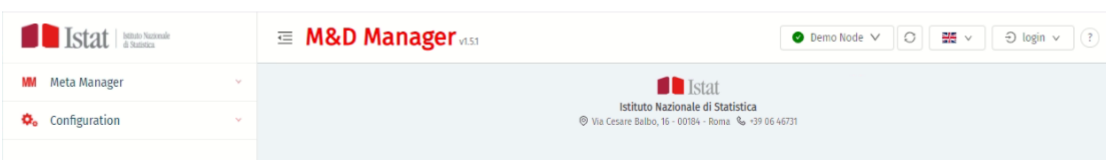
ONE

HOME

To start the 'Data & Meta Manager' tool, the url must be retrieved through the internet browser.
The url is provided by the system administrator.
The first screen to be displayed is the **Welcome Mask**.
The Suite presents on the left the Application Menu that varies according to the user profile logged in or not.
The Welcome Screen shows the Suite header and the version of the application package; at the bottom there is the footer with the Istat logo.
This information is always fixed in the application graphics.
Through drop-down menus located in the upper right corner of the mask welcome mask, it is possible to set:

- Node
- Language
- Login

Starting from this mask the System is accessible by a profiled user or alternatively by an anonymous non-profiled user.
The System automatically sets the predefined language and defaults that allow an Anonymous User to access the connection.
The Application Menu is always present and it can be displayed in full or in compressed form (with icons only).



CHAPTER

TWO

ACRONYMS AND GLOSSARY

API - ***Application programming interface***

Thanks to this interface, computer programs communicate with each other. It is comparable to the way a programmer sends instructions between programs.

Source: http://schoolofdata.org/handbook/appendix/glossary

Artefact

In order to describe and document data, the SDMX standard refers to entities that are called "artefacts" that are organized in such a way as to represent the data and reference it appropriately. The **Id**, **agency** and **version** are the three identifying elements of an SDMX **artefact**.

Source: https://www.istat.it/it/files/2013/07/Step_funz_client_SDMXWS1.pdf

DATABASE

A database is a set of structured data, i.e. homogeneous in content and format, stored in a computer, representing the digital version of a data archive or filing cabinet.

Source: https://it.wikipedia.org/wiki/Base_di_dati

Directive 96/9/EC[2] defines b.d. as: "a collection of works, data or other independent elements systematically or methodically arranged and individually accessible by electronic means or otherwise" (Art. 1. no. 2); and provides for legal protection of "databases which, by reason of the selection or arrangement of the material, constitute the author's own intellectual creation" (Art. 3).

Source: https://www.altalex.com/documents/news/2013/07/02/banche-dati-la-tutela-giuridica-nella-giurisprudenza-della-corte-di-giustizia

Category Scheme

SDMX artefact, is a set of categories, hierarchically organised, that classify DataFlows.

Source: https://www.istat.it/it/files/2013/07/Step_funz_client_SDMXWS1.pdf

CKAN - ***Comprehensive knowledge archive network***

A data management system that makes data accessible by providing tools to simplify its publication, sharing, retrieval and use. It is intended for data publishers (national and regional administrations, companies and organisations) who wish to make their data open and accessible.

Source: http://ckan.org/

Code List

SDMX artefact, is a list of codes associated with components (dimensions and coded attributes) of DSD.

Source: https://www.istat.it/it/files/2013/07/Step_funz_client_SDMXWS1.pdf

Concept Scheme

SDMX artefact, is a grouping of concepts referencing components (dimensions, attributes, measures) of DSD.

Source: https://www.istat.it/it/files/2013/07/Step_funz_client_SDMXWS1.pdf

HCL

Hierarchical Code List

CORDIS

The main public repository and portal of the European Commission, used to disseminate information on all EU-funded research projects and their results.

Source: http://cordis.europa.eu/home_it.html

CSV - ***Comma separated values***

The "comma separated values" file format is often used for exchanging data between different applications of the same type. The "comma separated values" file format is compatible with KDE spreadsheets. It is compatible with KSpread, OpenOffice Calc and Microsoft Excel spreadsheets. Many other applications support the format for importing and exporting data.

Source: http://edoceo.com/utilitas/csv-file-format

DataFlow

SDMX artefact, is a structure that describes the content of a dataset that the producing organisation provides for different time periods.

Source: https://www.istat.it/it/files/2013/07/Step_funz_client_SDMXWS1.pdf

DDB - ***Dissemination Data Base***

Dissemination database of statistical data.

Source: https://ec.europa.eu/eurostat/web/sdmx-infospace/sdmx-it-tools/sdmx-ri

DCAT - ***Data catalogue vocabulary***

RDF vocabulary allowing interoperability of data catalogues.

See also: W3C - http://www.w3.org/TR/vocab-dcat

DCAT-AP - ***DCAT application profile***

Common vocabulary, based on DCAT, to describe the hosted datasets of data portals in Europe.

See also:
https://joinup.ec.europa.eu/asset/dcat_application_profile/description

DSD - ***Data Structure Definitions***

SDMX artefact, is the definition of a data structure in terms of its components, i.e. dimensions, attributes and measures.

Dimensions: components that identify and describe the observed phenomenon.

Attributes: components that only describe the observed phenomenon.

Measures: components that represent the observed phenomenon(s).

Source: https://www.istat.it/it/files/2013/07/Step_funz_client_SDMXWS1.pdf

ENDPOINT SPARQL

A service that accepts SPARQL queries and provides results in the form of SPARQL result sets. Data providers follow the good practice of indicating the URL address of their SPARQL endpoint to allow access to their data programmatically or through a web interface.

Source: W3C - http://www.w3.org/TR/ld-glossary/#sparql-endpoint

IIS - ***Internet Information Services***

A set of Internet server services for Microsoft Windows operating systems.

Source: https://it.wikipedia.org/wiki/Internet_Information_Services

JSON

"JavaScript object notation" is an open-standard format that uses human-readable text to transmit data objects consisting of attribute-value pairs.
It is the most common data format used for asynchronous browser/server communication/navigation (AJAJ).

Source: https://en.wikipedia.org/wiki/JSON

Mapping Assistant

Application that facilitates the mapping between structural metadata provided by a DSD and data residing in the dissemination database of a dissemination environment.
The Mapping Assistant stores mappings between SDMX and the local data storage schema in the Mapping Store database, or MSDB.

Source: https://ec.europa.eu/eurostat/web/sdmx-infospace/sdmx-it-tools/sdmx-ri

METADATA

Metadata is structured information that describes, explains, positions or enables easier retrieval, use or management of an information resource.
They are often referred to as 'data about data'.

Source: http://www.niso.org/publications/press/UnderstandingMetadata.pdf

MSD - ***Metadata Structure Definitions***

SDMX artefact, is a model for reporting and disseminating reference metadata.

In addition to what is called "structural metadata" which mainly concerns the definition of the data structure, it may be useful to share other information such as, for example, information on methodology, data quality, sources, contact details.. This data in SDMX is called "referential metadata" and finds its framework in the Metadata Structure Definition and appropriate additional concepts.

Source: "Measuring the Data Universe: Data Integration Using Statistical Data and Metadata Exchange" by authors Reinhold Stahl and Patricia Staab.

MSDB - ***Mapping Store database***

Database created by the Mapping Assistant application

Source: https://ec.europa.eu/eurostat/web/sdmx-infospace/sdmx-it-tools/sdmx-ri

NSI_WS

The NSI Web Service on SOAP or REST request collects the data from the DDB and the structural metadata from the MSDB, creates the SDMX-ML data and sends it to the client.
The NSI Web Service can be in Java or .NET.

Source: https://ec.europa.eu/eurostat/web/sdmx-infospace/sdmx-it-tools/sdmx-ri

OECD - ***Organization for Economic Co-operation and Development***

The Organisation for Economic Co-operation and Development (OECD) is an international organisation for economic studies for its member countries, which are developed countries sharing a market economy.

Source: https://it.wikipedia.org/wiki/Organizzazione_per_la_cooperazione_e_lo_sviluppo_economico

ONTOLOGY

A formal model for representing knowledge in a specific field. An ontology describes the types of existing objects (classes), the relationships between them (properties) and the logical ways of using the classes and properties together (axioms).

Source: http://www.w3.org/TR/ld-glossary/#ontology

PDF

The 'portable document format' is a file format used to present and exchange documents independently of the available software, hardware or operating systems.
This open standard is managed by the International Organisation for Standardisation (ISO).

Source: https://acrobat.adobe.com/be/en/products/about-adobe-pdf.html

RDF - ***Resource description framework***

A family of international standards for data exchange on the web. It is based on identifying objects using web identifiers or HTTP URIs and describing resources in terms of simple properties and property values.

Source: http://www.w3.org/TR/ld-glossary/#rdf

RDFa

"Resource description framework in attributes" is a W3C recommendation that adds a number of extensions in the category of attributes to XML, HTML and several types of XHTML-based documents to integrate rich metadata into web documents.

Source: https://en.wikipedia.org/wiki/RDFa

REPRESENTATION

The physical representation of a data set. Each resource can be a file of any type, a link to a file on the web or a link to an API. For example, if data is provided in various formats or split between different areas or time periods, each file is a "resource" that should be described separately.

Source: https://data.europa.eu/euodp/it/glossary

SDMX - ***Statistical data and metadata exchange***

The Statistical Data and Metadata Exchange is an international initiative to standardise and modernise mechanisms and procedures for the exchange of statistical data and metadata between international organisations and Member States.

Source: https://en.wikipedia.org/wiki/SDMX

DATA SET

A collection of related data sets consisting of distinct elements, but which can be treated as a unit and accessed or downloaded in one or more formats.

Source: https://data.europa.eu/euodp/it/glossary

SPARQL

Query language for data represented using RDF, analogous to SQL (Structured Query Language) for relational databases.

Source: http://www.w3.org/TR/ld-glossary/#sparql

URI - ***Uniform resource identifier***

A string that uniquely identifies virtually anything, from a building to more abstract concepts such as colours. It may or may not be resolvable on the web.

Source: http://www.w3.org/TR/ld-glossary/#uniform-resource-identifier

URL - ***Uniform resource locator***

A global identifier generally called a 'web address'. A URL is resolvable on the web. All HTTP URLs are URIs; however, not all URIs are URLs.

Source: http://www.w3.org/TR/ld-glossary/#uniform-resource-locator

URN - ***Uniform resource name***

Historical definition of a unique resource identifier (URI).

Source: https://en.wikipedia.org/wiki/Uniform_Resource_Name

XML - ***Extensible markup language***

A markup language that defines a set of standards for encoding documents in a machine-readable, human-readable format.

Source: https://en.wikipedia.org/wiki/XML

CHAPTER

THREE

ABOUT

The 'Data & Meta Manager' tool is a modern open source web-based SDMX application suite, fully integrated with the SDMX-RI released by Eurostat.
Based on a Loosely Coupled Architecture, 'SDMX Data & Meta Manager' allows the user to streamline the dissemination and reporting process, the management of metadata and to facilitate the publication of open (statistical) data according to the requirements detailed by the European and Italian Digital Agenda.
With a few steps an organization can easily build a dissemination/reporting database driven by SDMX structural metadata, expose the datasets through an SDMX Web Service and disseminate the data catalogue using the W3C Recommendation "DCAT" and the CKAN v3 API.

The purpose of this guide is to provide a tutorial for the most frequent use cases of applications within the "Data & Meta Manager" tool, for the user with any kind of profiling.

To facilitate the description of the applications, images captured during the use of the application itself have been included.
These images may have graphic differences depending on the configuration used.

3.1 Licence

….. coming soon …….

3.2 Overview

The 'Data & Meta Manager' tool consists of 5 main sections which are comprehensively set out on the various pages.

The main sections can be summarised as follows:

- *Meta Manager*
  Complete management of SDMX structural metadata (Codelist, Concept Schemes, Category Scheme, Data Streams, DSD, etc.), i.e. SDMX artefacts can,
  depending on user profiling, SDMX artefacts can be created, updated, archived, cloned and deleted.
- *Data Manager*
  Data publication available in just 4 steps:
  - Builder: creation of a DDB schema (data cubes) from DSD and related artifacts
  - Mapping: creation of mappings between imported files and cubes
  - Loader: data loading from CSV/SDMX-ML files
  - Dataflow Builder: creation and publication of dataflows

- *Utility*
  This section provides a series of tools that allow the quickest management and manipulation of the artefacts, for example the function Compare Item Scheme allows to compare two by two: Code Lists, Category Schemes and Concept Schemes.
- *Metadata Management*
  - Full management of referential metadata: production and display
    - Compilation of metadata reports
    - Machine readable metadata in SDMX-JSON via REST API
    - Human readable metadata in HTML via widgets, shareable or embeddable in another website
  - Dedicated interface for the completion of standard DCAT-AP metadata.
- *Mapping Handler*
  This section provides 4 steps to create, map and publish dataflows:
  - DDB Connection: creation or editing of a DDB connection from which datasets can be queried
  - Dataset Builder: creation or editing of datasets built with SQL queries performed on the DDB connections
  - Mapping Set: creation or editing of categorised dataflows and of the relative mapping set
  - Dataflow Publishing: publish dataflows created in the previous step

In addition to these sections, two more are includable:

- *User and Permission Management*
- *Application Management*

3.3 Changelog

….. coming soon …..

CHAPTER

FOUR

BASIC INFORMATION ABOUT SDMX

What is SDMX?

SDMX (Statisical Data and Metadata eXchange) is a collaboration of some of the most important international organisations with the aim of encouraging and improving the exchange and sharing of statistical data and metadata.
SDMX is:

- A Logical Model used to describe statistical data and also providing guidelines on how to structure the content.
- An Architecture which enables efficient automated machine-to-machine exchange and sharing of data and metadata.
- A technology that supports standardised information and methodological tools that can be used by all those involved in data exchange and processing.

Which benefits does it bring?

By accepting the use of a common description of the data, it is then used as a parameter guiding the exchange and processing of the data.
Data descriptions are made available to all, so that those interested in a certain topic can understand and use the data for different purposes.
SDMX is the leap from diverse and complex exchange systems to a common, harmonised and standardised exchange system.

The Standard

SDMX provides a way to model statistical data, structural metadata and the data exchange process and also defines a model for further explanatory metadata, the so-called reference metadata, which are generally in textual format.
In order to describe and document data, the standard refers to entities (in SDMX "**artefacts**") that are organised in such a way as to represent and refer to the data appropriately:
The **Id**, **agency** and **version** are the three identifying elements of an SDMX artefact.
The **Id** is the identification code of the artefact.
Agency is the name of the organisation that is the creator and/or owner of the artefact.
Version gives the version of the artefact. In particular, if the artefact is finalised, it is not possible to modify it unless a new version of the artefact is created.

The main artefacts used to describe data and statistical metadata are:

> **Dataflow**: structure describing the content of a dataset that the producing organisation provides for different reference time periods.
>
> The characterising element of Dataflow, is the **Data Structure Definition (DSD)** as it defines the constitutive structure in terms of components (dimensions, attributes, measures).

NOTE:
Remember that the ***Dimensions*** are qualitative characteristics of the statistical units (e.g. reference period, reporting country, frequency, gender, …).
Attributes represent a qualitative characteristic of the observed data (confidentiality, status, version, decimals, unit of measure, table title, …), while ***Measures*** are the values of the observations.

> **Concept Scheme**: is a grouping of concepts that refer to the components (dimensions, attributes, measures) of the DSD.
>
> **Code List**: is a list of codes associated to components (dimensions and coded attributes) of the DSD.
>
> **Category Scheme**: is a set of categories, hierarchically organised, that classify Dataflows.

These and other artefacts will be the focus of the **'Data & Meta Manager'** tool.

CHAPTER

FIVE

INSTALLATION

Check the **MetaDataManager - Installation manual** provided with this user manual.

CHAPTER

SIX

ADMINISTRATION

This section of the guide will highlight the steps needed to properly configure the tool to the user's needs and the actions required to create and manage nodes containing SDMX resources and the profiling of users on these nodes.

6.1 How to configure the application

To manage the Configuration, the System provides an application interface which can be used by a profiled user with **SuperUser** permissions, also referred to in the guide as *superadmin*.
The reference function can be found in the left-hand side menu under **Configuration**.
The item of interest is ***Application***.
After entering the credentials of *superadmin*, the System opens the management mask of the application configuration by positioning itself in the section "**User Interface**".
The following configurations can be made for the Data and Meta Manager applications:

- *User Interface*
- *Agencies*
- *Data Management*
- *Structures default header submit*
- *Superuser credentials*
- *Endpoint Settings*

All application configurations are described in the dedicated sections.

6.1.1 User Interface

In this section the *SuperUser* has the possibility to set the parameters related to the standard of views:

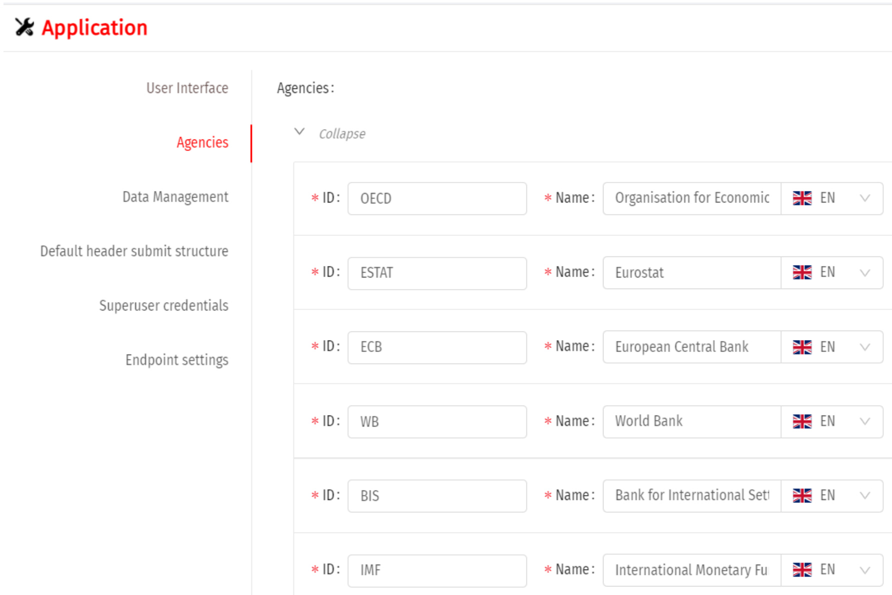
- **Maximum tree nodes for allowing tree visualization**
  This parameter represents the maximum display in tree representation of the elements (e.g. Item of a Code List).
  The system manages vertical scrolling for the display of items and this also allows high values of the parameter.
- **Maximum showable tree nodes using 'Show more' in tree**
  The parameter is linked to the value set in the previous point.
  The system allows the loading of further elements for displaying them through the function Show Others.

The number of elements that are displayed is determined by this pagination parameter in the tree representation.

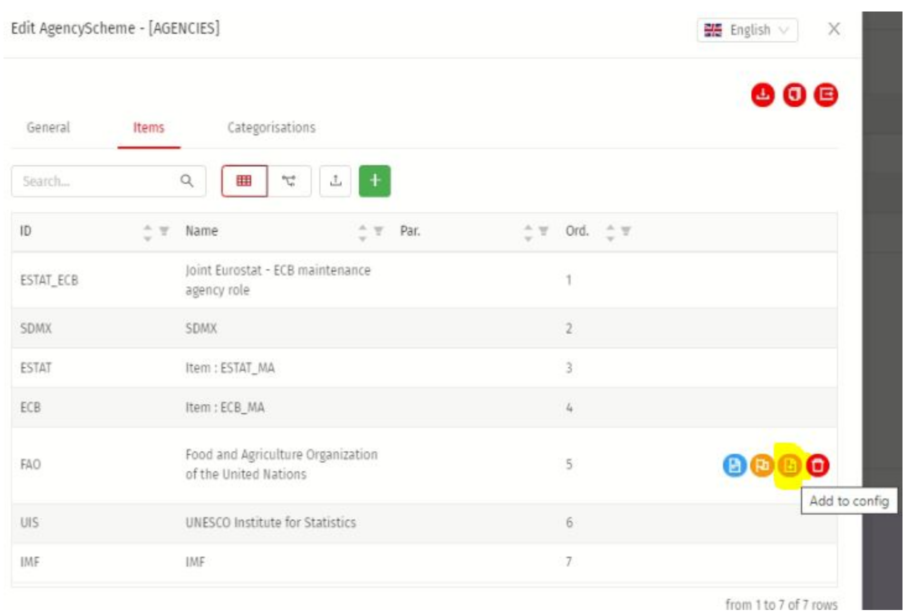
- **Minimum nodes number to enable tree pagination**
  Indicates the minimum number of elements displayed in the tree view.
  The minimum number is indicative and relative to the actual number of elements present in the system.
- **Maximum tree nodes for allowing tree total expansion**
  Indicates the maximum number of elements that can be displayed with the tree fully expanded.
- **Tree page size**
  Indicates the maximum number of siblings that are displayed by default.
- **Default sidebar collapsed**
  The on/off checkbox allows the user to set as default the display of the left sidebar menu in extended (header) or compressed (icons only) mode.
  If this control is "off" i.e., the display of the left sidebar menu is set as extended, then the user can set on/off on the control: "**Default sidebar 2nd level menu**" to start the application with the sidebar menu on the left with all second level items expanded.
- **Languages**
  The Languages section allows the user to define the languages for the Suite.
  The Add Language function opens an editing window where the user can set the language in character format.
  By saving the configuration the user obtains at application level the language management for the graphic interface (labels, titles, etc.)
  The languages are always represented by a pop-up list (combo).
- **Default language**
  The System Administrator can define the default language, which is automatically set in all application sessions.
- **Anonymous pages**
  In this session the Administrator can set the functionalities visible to an anonymous user who logs in without credentials.
  The mask shows all the functions present in the Suite divided between the two main applications Data and Meta Manager.
  By selecting the checkbox associated with the function, the anonymous user is authorised to have read access to the function itself.

6.1.2 Agencies

In this section the *SuperUser* can set the Agencies that will be used for the definition of the Artifacts when creating and/or editing.



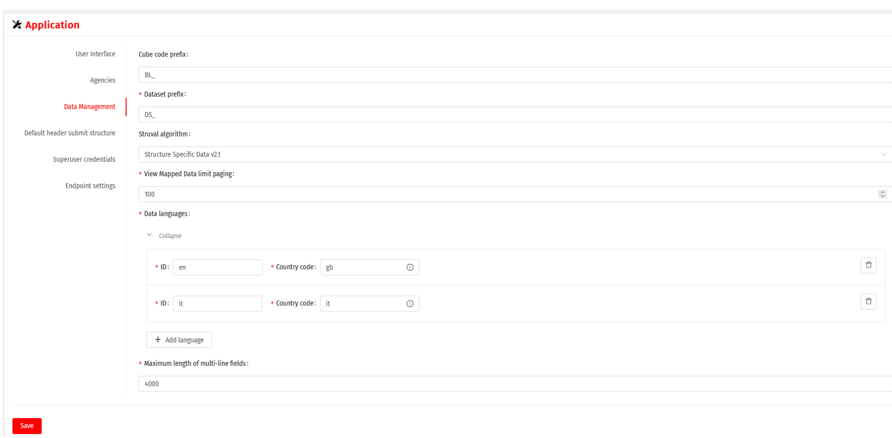
These are the only ones that the administrator user will be able to assign to users in the 'Set Permissions' section, thus giving them the permissions to manage the relevant SDMX Artifacts: these agencies are independent from those defined in the SDMX artifacts of type Agency Schema. However, it is possible to explicitly add an Agency belonging to an Agency Schema to the current node configuration by clicking on the appropriate action in the row.



6.1.3 Data management

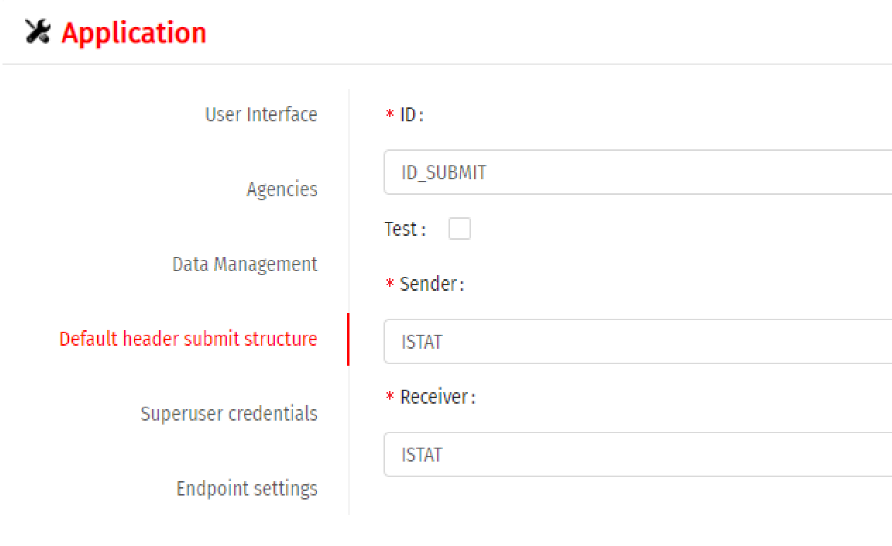
This section allows the user to define:

- **The standard prefix of the Cube's identifier (ID)**
  This prefix will be set in the creation mask of the Cube and cannot be modified.
- **The standard prefix of a dataset used by the dataflow builder**
  This prefix is "DS_" by default and will be used by the dataflow builder. Datasets with this prefix **WILL NOT** be usable in any of the *Mapping Handler* functionalities.
- **The preferred struval algorithm**
  The user is able to select between various struval algorithms supported by MA Web to perform the "Format Validation" in the mapping set feature (part of the mapping handler menu).
- **The pagination size of the view mapped data functionality**
  The user is able to select the page size when previewing the paginated preview of mapped data returned by the MA Web. This applies only to the features of the *Mapping Handler*.
- **Data Languages**
  Language of the data processed in the management phase of Names, Descriptions and Annotations.
- **Maximum length of multi-line fields**
  Maximum number of editable characters of multi-line fields such as Descriptions and Annotations.



6.1.4 Structures default header submit

This section allows the user to parameterize the information related to a submit operation with reference to the sender and receiver prefixes, submit id, data submit of a Dataflow Builder header.

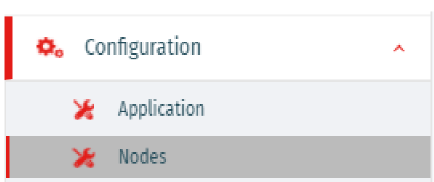


6.1.5 Superadmin credentials

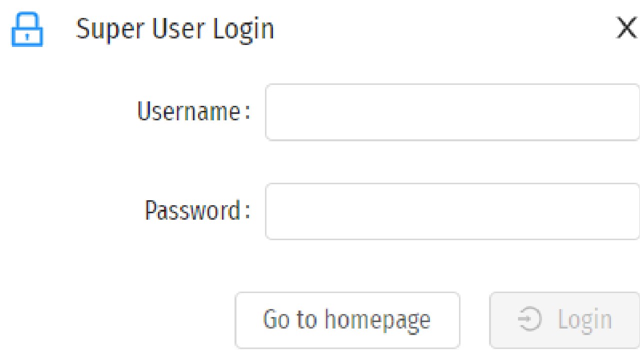
The SuperUser is configured from the configuration file with the value "**Superadmin**".
This allows the SuperUser to access the System with predefined credentials and configure the Suite.
The SuperUser can modify his profile in terms of username and password or delete the default profile and generate a new one.

Login with the SuperUser profile.

Once the user enters the *Welcome Mask* of the Suite, he can access the ***Application*** item of interest in the left menu starting from the Configuration item.



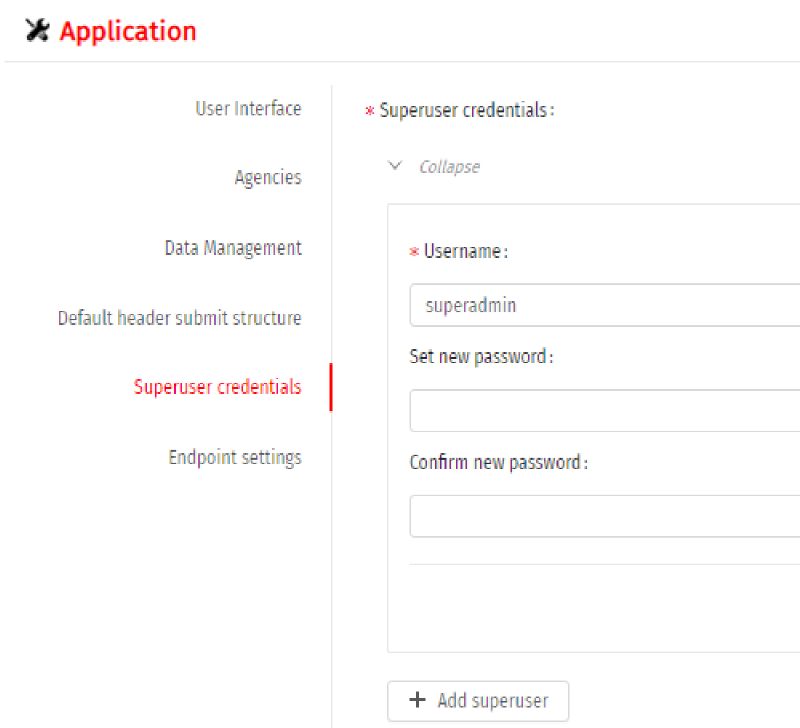
Selecting the function Application, the system asks for authentication of the SuperUser who has the credentials which make him able to configure the application management.



The user sets the SuperUser credentials and presses the **Login** button which is activated once the User and Password have been set in the appropriate edit fields.

Edit SuperUser profile.

After entering the credentials, the system will open the management mask of the application configuration by clicking on the section "**User interface**".
By selecting the section "**SuperUser credentials**" it is possible to change the password and, if necessary, create an additional SuperUser, using the appropriate button.
The need to add more than one SuperUser is subject to the organization within the institution or entity that uses the Suite.

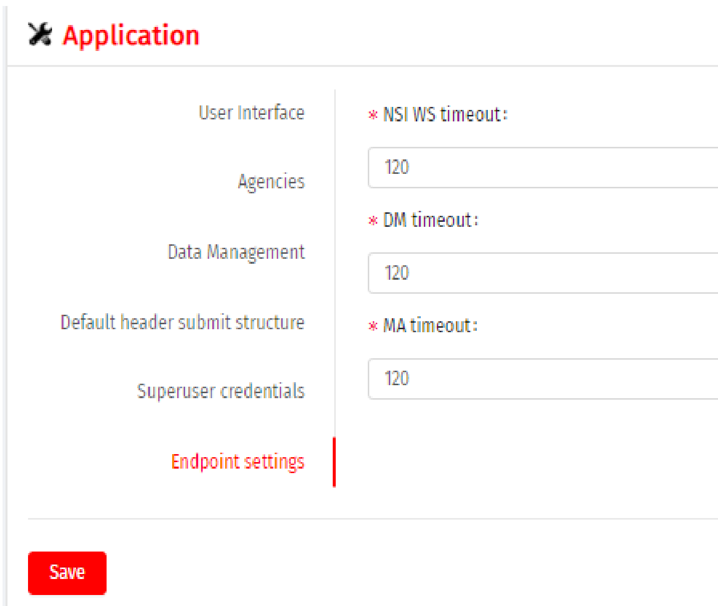


6.1.6 Endpoint Settings

This section allows you to define the timeout of the relevant WebServices in the node in seconds:

- **NSI WS timeout**
- **DM timeout**
- **MA timeout**

By default they are all set to 360 sec. in total.

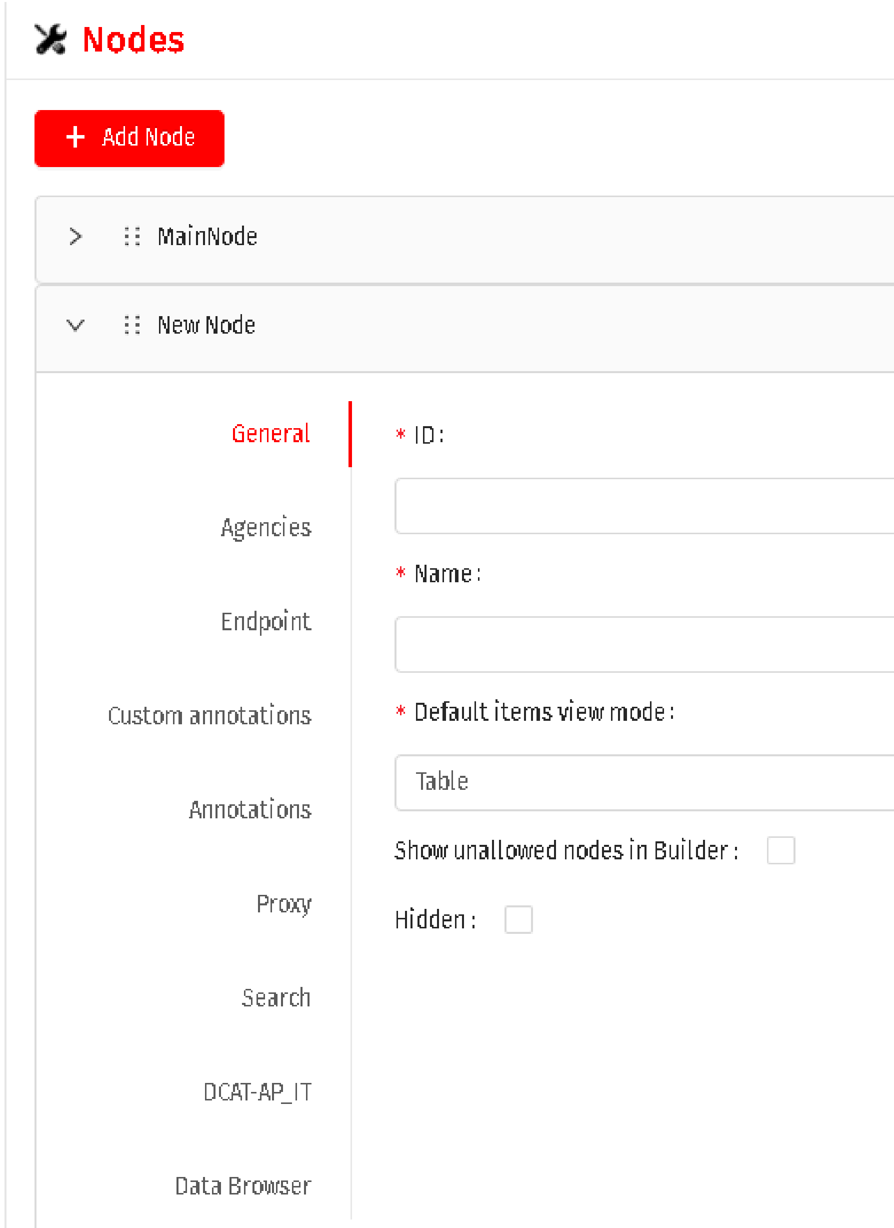


6.2 Node management

- *Configuring a new node*
- *Other Node configurations*
- *Plugin Settings*

6.2.1 Configuring a new node

To configure a Node, the user must access the function Configuration –> Nodes. The system will request access as *SuperUser* and then select the "Add Node" button to launch the mask



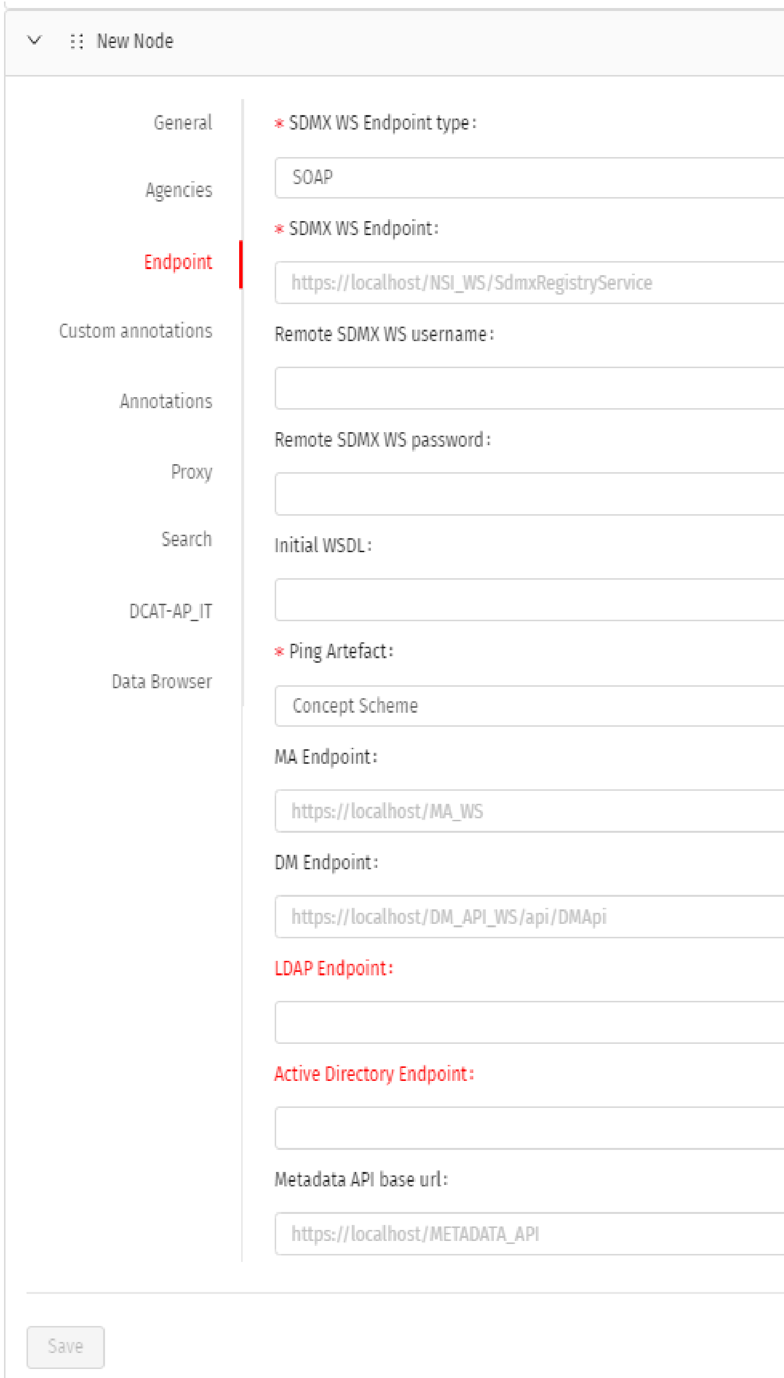
In the "**General**" section set the following fields (the values are indicative for example purposes):

- **ID** : Node1
- **Name** : Node1
- **Default items view mode**: Table as default
- **Show unallowed nodes in Builder**: Allows visualization (in grey) of nodes in the Builder tree on which the user has no permissions.
- **Hidden**: do not select the check as default otherwise the node will be hidden to users.

After setting the general data it can be seen that the "Save" button is not activated as the information is not sufficient to define the Node itself.

Go to the "**Agencies**" section.
This section contains the *Default Agencies* which are set during the initialisation of the database at the first application installation.
The agencies linked to the Node can be modified with respect to the application default and therefore additional agencies can be added or removed.
In this example the agencies are not modified.

Go to the "**Endpoints**" section to set the endpoints that will allow the true profiling of the Node.



As can be seen in the figure above, some fields are set by default, others are not set

and the System provides an example of configuration with regard to the url needed to reach the WS internal/external to the System itself and thus allow at application level the connection to the databases managed by them.
Some WS allow read and write access to the database, others read only, regardless of User profiling.
The System requires mandatory configuration of the SDMX WS Endpoint. To configure it the user must set in the provided field:

- https://localhost/NSI_WS/SdmxRegistryService

where localhost identifies the host where the WS is present.
This can also be reached through a remote virtual machine and in this case the hostname or IP address or the address defined at IIS level during the system-application configuration phase must be set (e.g.: www.org.it - See *Installation Manual*).
After typing the field with the reference to the endpoint, press the "Ping" button to the right of the field itself.
After verification, the system will display the green button to indicate that the endpoint has been correctly defined and can be reached.
If the Ping function does not have a positive result, this may be due to several reasons that can be found in the application logs.
We list the most common errors:

- Incorrect url typed
- Non-reachability of the database due to incorrect definition of the connection string in the configuration files (see *Installation Manual*).

With this minimum configuration it is possible at application level to read the MSDB, Eurostat database, with access to the Meta Manager functions but it does not allow access to the System and writing.
To do this, and access the functions in read and write mode, the Endpoint fields must be set:

- Endpoint MA: https://localhost/MA_WS
- Endpoint DM: https://localhost/DM_API_WS/api/DMApi

and selecting the "Save" button, the Node is created.

Note By pressing the "Save" button the System requests the Administrator user name, if the Administrator user name (admin) is set and "OK" is pressed, the Node is saved.

Node Configuration Scenarios

The System allows the configuration of Nodes with different architectural scenarios in order to access WS and related databases.
There are mainly two scenarios:

The first scenario foresees access to INTERNAL Nodes (read/write) each configured to access WSs and related databases residing on different servers through NodeApi plugging. Each Node will have a different URL that identifies the connection to the WS.
The second scenario involves access to several EXTERNAL Nodes (read-only) with only access to the NSI_WS and the MSDB database.

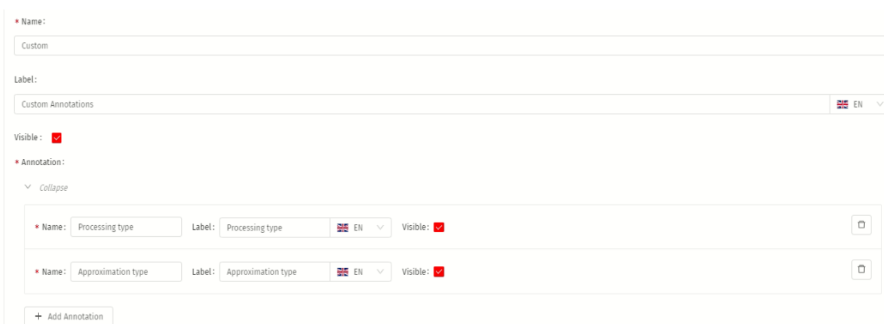
6.2.2 Other Node configurations

To configure/edit a Node the user must access the function Configuration – > Nodes.
The system requires access as SuperUser.
Once logged in and selected the node to be modified, it is possible to choose the section on which to take action:

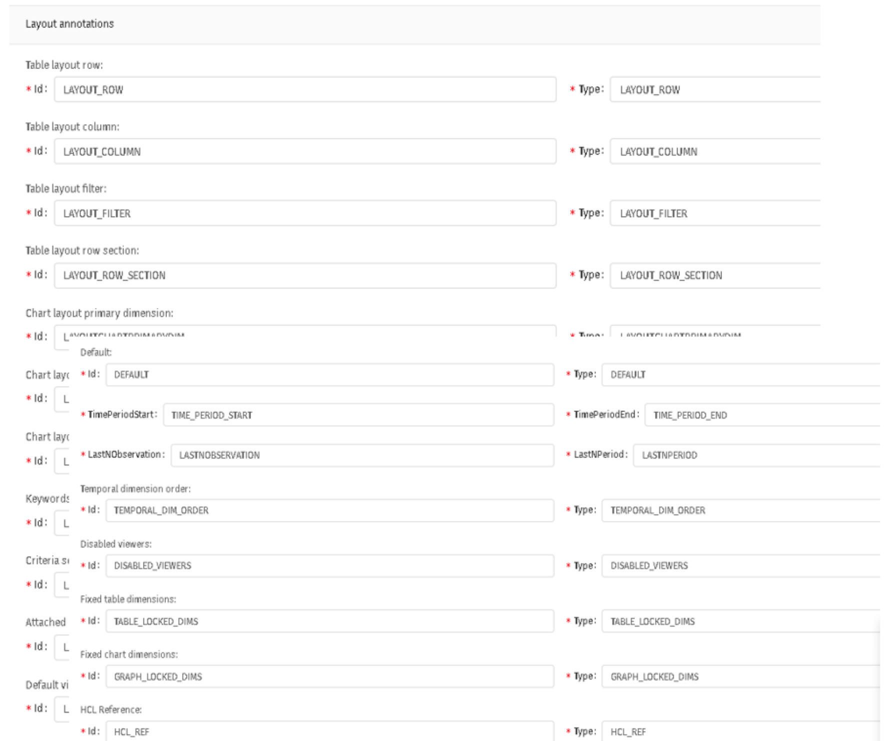
- General
- Agency
- Endpoint

These first 3 sections are necessary for the *node creation*.
Let's look at the next ones:

- **Custom Annotations** This section allows the user to define the Annotation table for the Node in terms of Tab Annotation Name, Annotation Name, Annotation Visibility, Explosion or Collapse of multi-line texts.
  After setting the required fields, the user saves the settings using the save function represented by the Save button.
  Multiple annotations can be configured using the Add Annotation function.
  This section allows the configuration of Custom Annotations. To define annotations, the necessary elements are:
  - Name
  - List of custom annotations that can be acquired by typing in the annotation name.



- **Anotations** This section allows the user to view/edit standard annotations (also called working annotations), e.g. those of the OECD.
  In the Annotation configuration, both the ID and the type of the annotation are configurable: by default these two fields are initialized with identical values however they can be configured in appropriate configuration file.



These annotations have been configured in the node to give greater flexibility to the annotation itself.
The TYPE of an annotation, in fact, could theoretically change according to the Data Viewer external to the Suite that will be used (e.g. the type=LAYOUT_ROW could in a certain viewer be called TABLE_LAYOUT_ROW) so, for this reason, in each node, during configuration, the user has the possibility to redefine the TYPE of the Working Annotations as shown in the previous image.
The Data Browser configurations (in case it is chosen as the data viewer) are found in the "Annotation" section, as sub-daughters of the "Default" configuration, at the node level.

The list of currently used annotations and their use is shown below:

Order annotations

- Concepts order - Annotation to manage the ordering of concepts in a concept scheme
- Categories order - Annotation to manage the ordering of the categories of a category scheme
- Codes order - Annotation for sorting the codes of a codelist
- Categorisations order - Annotation to manage the sorting between categorisations at the same hierarchical level

Layout annotations

- Table layout row - Dimensions to be shown in rows in the table layout
- Table layout column - Dimensions to be shown in columns in the table layout
- Table layout filter - Dimensions to be shown in filters in the table layout
- Table layout row section - Dimensions to be shown in section in the table layout

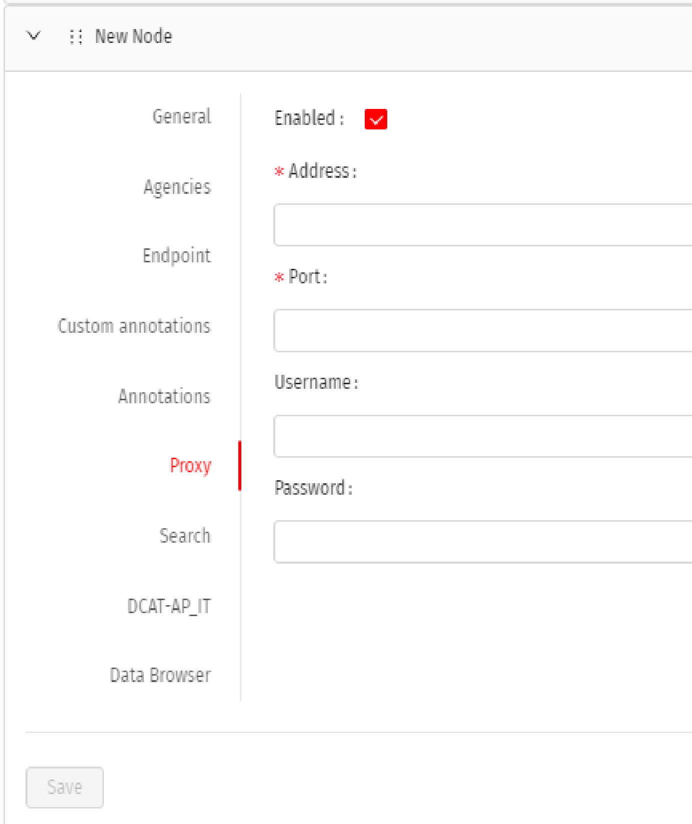
- Chart layout primary dimension - Dimension to show as primary in the chart layout
- Chart layout secondary dimension - Dimension to be shown as secondary in the chart layout
- Chart layout filter - Dimensions to be shown in filters in the chart layout
- Keywords - List of keywords associated to a dataflow
- Criteria selection - Default selection mode for the Data Browser
- Attached data files - file formats for which download is possible
- Default view - Default view of the dataflow in the Data Browser (table, graph or map)
- Decimal separator - Character to be used as decimal separator in the Data Browser
- Number of decimals - Number of decimals for the table representation in the Data Browser
- Metadata URL - Metadata url
- Empty cell placeholder - Characters to be used to replace any empty cells in the Data Browser
- Dataflow notes - Notes associated with the dataflow
- Geo ID - Id of the geographical dimension
- Layout map filter - Dimensions to be shown in filters in the map layout
- Dataflow source - Source of the dataflow (e.g. Eurostat)
- Hidden - Dataflow to be hidden in the Data Browser catalogue
- Not displayed - Dimension (or its values) to be hidden
- Full name - Full name of an item in an itemscheme
- Default - Default values for a dimension (they are used to initialise the Criteria in the Data Browser)
- TimePeriodStart - Default start of TimePeriod for the representation in Data Browser
- TimePeriodEnd - Default end of TimePeriod for the representation in Data Browser
- LastNObservation - Default number of observation for the representation in Data Browser
- LastNPeriod - Number of last default periods to be displayed
- Temporal dimension order - Annotation to manage the ordering of temporal dimension
- Disabled viewers - Annotation to disable the "Table," "Graph," or "Map" viewers in the DataBrowser
- Fixed table dimensions - List of dimensions to be made fixed in the tabular view in the DataBrowser
- Fixed chart dimensions - List of dimensions to be made fixed in the chart view in the DataBrowser

- HCL Reference - Annotation to manage the association between a hierarchical codelist and a dimension or to an encoded attribute of a dsd or dataflow
- Anticipate verification on Max Num Records - Annotation to check in advance whether the threshold of the maximum number of records viewable in Data Browser has been exceeded

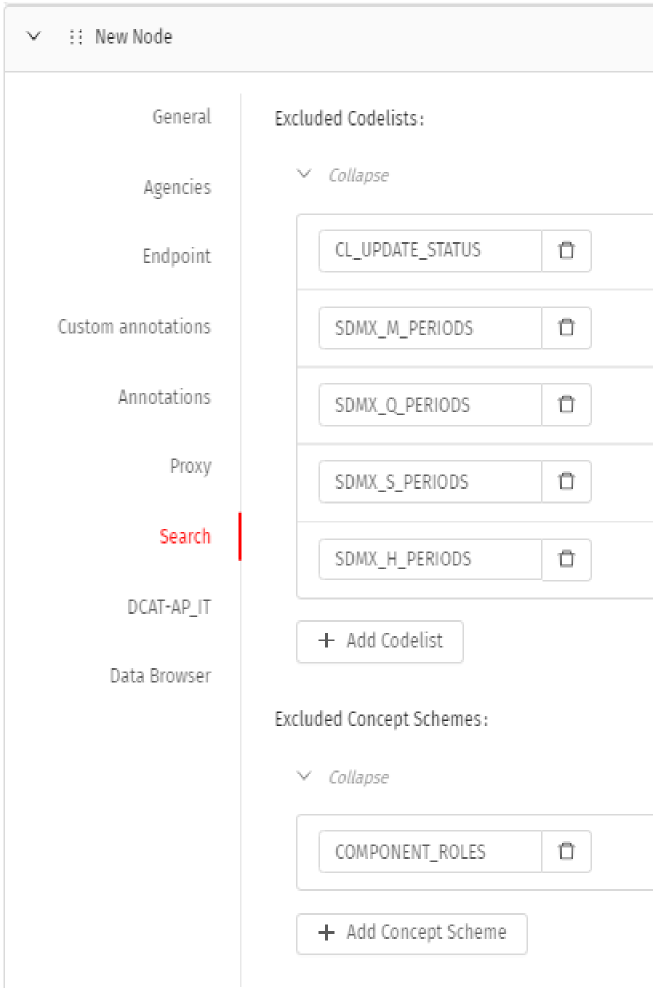
Working annotations

- Linked Dataflow Node - Indicates that a dataflow type is linked and specify the node to which the dataflow is linked
- DDB dataflow - Indicates that a dataflow has its counterpart on the Dissemination Database (and therefore has associated data).
- Custom DSD - Indicates that the DSD has been built automatically after a dimensions reduction when building the data using the Dataflow Builder.
- Associated cube - Indicates that the DSD has an associated cube
- Changed - Indicates that the final codelist was subsequently changed
- Metadataset - Metadataset associated with the artefact
- Have metadata - Indicates that the dataflow has associated metadata
- Restricted for publication - Indicates that there are restrictions for publication
- Attached file path - Indicates the path of any attachments
- DCAT is multilingual - Indicates that a DCAT attribute is multilingual.
- Custom is presentational - Indicates that an attribute is 'Presentational'.
- Last update - Indicates the date of last modification of the dataflow data.
- Dataflow catalog type - Identify dataflow type (e.g. normal, virtual, linked).

Proxy
The section allows the user to define the Proxy for the Node in terms of Enabled, Address, Port, Username, Password. After having set the fields, the SuperUser saves the settings through the save function represented by the Save button. It is left to the System administrator to assess the need for the definition of a network proxy in the organisation.

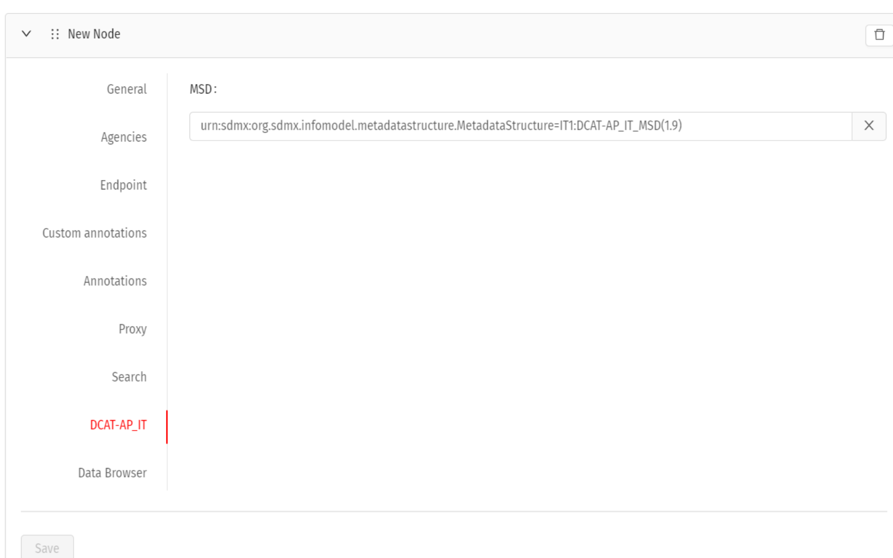


Search
This section allows the user to define the Search mode for the Node excluding Code Lists and Concept Schemes, set with Identifier, by means of the relative buttons **Add Code List** and **Add Concept Scheme**.
This function allows the Super User to insert an additional Code in the exclusion lists. Alternatively a Code in exclusion can be deleted by means of the delete function represented by a trash can icon.
The SuperUser, after setting the information he needs, saves the configuration by clicking the Save button.



DCAT-AP_IT

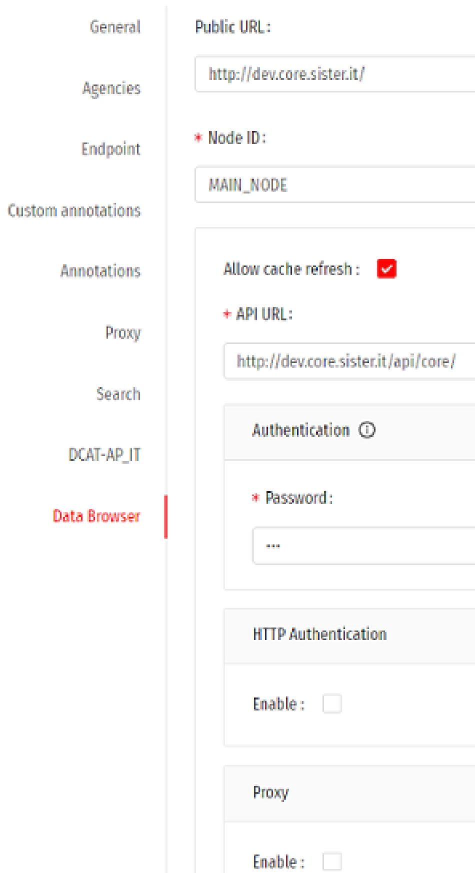
The DCAT-AP_IT section allows the user to define the URN parameters for the MSD, the SuperUser can choose, through a drop-down menu, which is the suitable MSD and saves the choice through the Save button.



Data Browser
The Data Browser section allows the user to define the parameters related to the Data Browser platform to which the user can connect to:

- view the dataflows published (by clicking in the section 'Data Manager' - 'Dataflow Builder');
- allow the cleaning of the catalogue cache (by clicking in the section 'Data Manager' - 'Update Data Browser Cache');
- allow the cleaning of the dataflow cache at the moment of its publication in the section 'Data Manager' - 'Dataflow builder' - 'Publishing';
- allow the cleaning of the DSD cache and all its linked artifact (by clicking in the section 'Data Manager' - 'Update Data Browser DSD Cache').

In this section the user enters the Data Browser url (used to view the data), the ID of the node to manage and, if cache management is wanted, the user checks, the box 'Allow cache refresh' by entering also URL of the Data Browser API, credentials for accessing Data Browser's Api machine-to-machine methods to manage caches, any credentials for authentication http or any proxy.

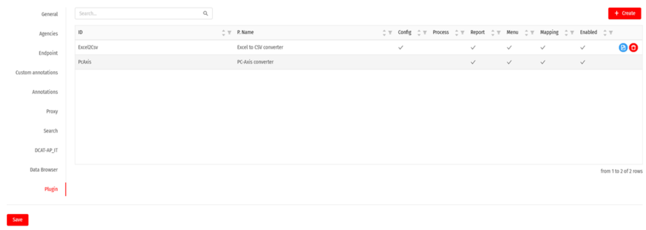


6.2.3 Plugin Settings

The plugin section allows the user to configure which plugins can be used in the node.

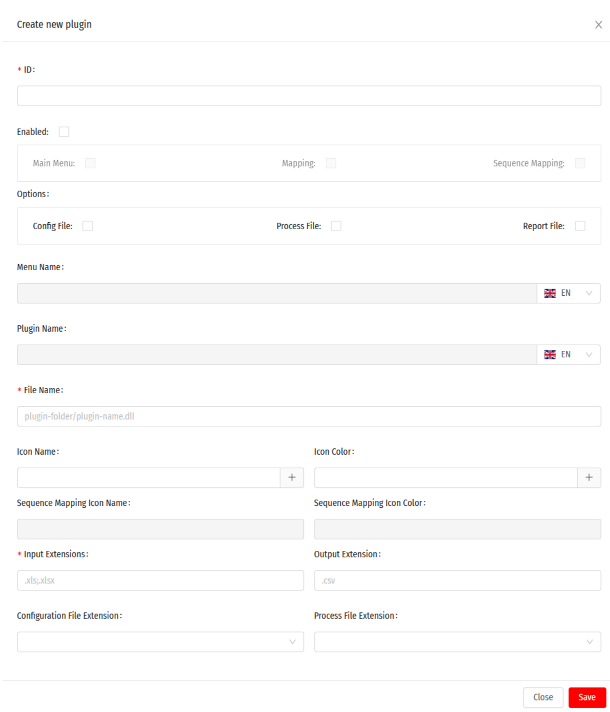
The plugins can be used to convert files in a variety of formats. For each plugin it is possible to specify a variety of settings, which will be explained shortly.

The plugin tab shows a table with all the plugins currently configured for the node. To create a plugin, the user can click on the "Create" button in the top right corner of the screen. If the user wants to edit it, the blue button on the right of the selected row can be clicked. Lastly, if the user wants to delete a plugin from the node, the red button on the right can be clicked.



6.2.3.1 Creating a plugin

When the user clicks on the "Create" button, a modal will pop up with a form, which needs to be filled with the plugin's settings.



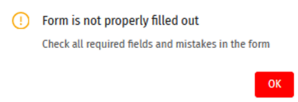
1. The "ID" field must be unique among the plugins of the node, it identifies the plugin univocally and it is required.
2. The "Enabled" flag is needed to set the plugin as visible in the node. To specify where exactly the plugin should be visible, the "Main Menu" and "Mapping" flag will be selectable when the "Enabled" flag is selected.

3. The “Main Menu” flag specifies if the plugin will appear in the “plugin” menu under the “utilities” menu in the sidebar. It will be displayed using the “Menu Name”.
4. The “Mapping” flag specifies if the plugin will appear in the “File Mapping” and “Loader” pages using the “Plugin Name”.
5. The “Sequence Mapping” flag specifies if the plugin is enabled to process sequence mapping files. It is available only when the “Mapping” flag is enabled.
6. The “Config File” flag specifies if the plugin requires a configuration file to support its input file.
7. The “Process File” flag specifies if the plugin requires a process file to support its input file.
8. The “Report File” flag specifies if the plugin will also output a report with its output file.
9. The field “Menu Name” is a multi-language field that specifies the name shown for the plugin in the “Plugin” sidebar submenu under the “utilities” menu. It is disabled until the flag “Main Menu” is enabled, at which point it also becomes required.
10. The field “Plugin Name” is a multi-language field that specifies the name shown for the plugin in the “file mapping” and “loader” pages. It is disabled until the flag “Mapping” is enabled, at which point it also becomes required.
11. The “File Name” field is a required one, and it specifies which DLL on the server the plugin should refer to so that it can perform its conversion. The path MUST be relative to the Plugins folder on the backend. (Example: if the plugin is in “Plugins\Excel2Csv\ExcelWrapper.dll” then the field will be filled in with “Excel2Csv\ExcelWrapper.dll”)
12. The “Icon Name” field is not required. It defines the icon associated with the output file of the plugin used in the Mapping view. You can select the icon from a pre-defined list.
13. The “Icon Color” field is not required. It defines the color of the icon associated with the output file of the plugin used in the Mapping view. You can select the color from a pre-defined list. This color MUST be different from the one in the “Sequence Mapping Icon Color” field.
14. The “Sequence Mapping Icon Name” is enabled only if the “Sequence Mapping” flag is enabled. It specifies the icon name that the sequence mappings generated by the plugin will have.
15. The “Sequence Mapping Icon Color” is enabled only if the “Sequence Mapping” flag is enabled. It specifies the icon color that the sequence mappings generated by the plugin will have. This color MUST be different from the one in the “Icon Color” field.
16. The Input Extensions field is always required, and it specifies the extensions that the plugin will accept as input. They can be multiple, must be written with the leading dot and separated with either a comma or a semicolon (example: “.xls;.xlsx”). Failure to compile the field accordingly will create an error which will not let the user save the plugin.
17. The “Output Extension” field is only required when the flag “Main Menu” is enabled. It specifies which type of file will be the output of the plugin when used in

standalone mode. When used in the file mapping or in the loader, the output of the plugin is always ".csv".

18. The "Configuration File Extension" is a field the is enabled and required only when the flag "Config File" is set. It specifies the extension that the plugin should expect as its configuration file.
19. The "Process File Extension" is a field the is enabled and required only when the flag "Process File" is set. It specifies the extension that the plugin should expect as its process file.

When the plugin has been configured correctly, the user can click on "save". In case there is a validation error, a new modal will appear



In this case all the fields should be reviewed, and any warnings should be addressed.

The plugin should be now visible in the node plugin table. When the save process did not encounter any validation errors, the node MUST be saved to save the plugin permanently in it.

6.2.3.2 Editing a plugin

Editing a plugin can be done by clicking on the blue edit button on the row of the selected plugin.

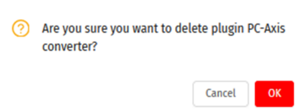
The same modal of the create plugin will pop up, but this time the ID will be unchangeable, and all the fields will be prefilled with the plugin's settings. The same rules of the creation apply to the editing of the plugin.

The user MUST remember to save the node as well, otherwise the changes will not take effect.

6.2.3.3 Deleting a plugin

Deleting a plugin can be done by clicking on the red delete button on the row of the selected plugin. A confirmation modal will pop up and when confirmed, the plugin will be deleted.

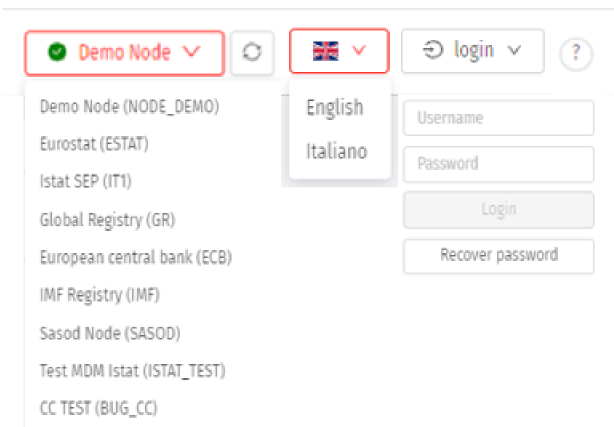
The user MUST remember to save the node as well, otherwise the changes will not take effect.



6.3 User and permission management

6.3.1 Login and User Types

Login
In order to access the application functions in read and write mode, the profiled user (administrator and/or user of the node) must choose the node for which he is profiled, the desired language and log in.
Selecting the **Login** button, located in the top right-hand corner of the *Welcome Mask*, the System asks the user to enter his username and password in order to login.



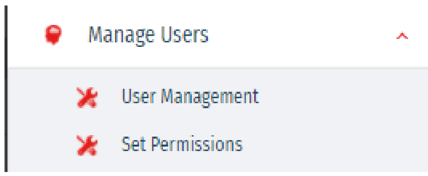
To access the application functions there are different **types of user**:

- SUPERUSER
  User dedicated to the System initialisation, to Application Configuration management(e.g. connections to nodes) and to the definition of other super-users.
- ADMINISTRATOR User dedicated to the complete management of the Suite at node level.
  If the administrator user has been set as user manager, he can also define new users (also administrators) and manage the users at single node level.
- USER Accesses the System through login (user and password).
  Has read and write permissions to manage Artifacts relative to the Agency for which he has been profiled by the System Administrator at node level.
- ANONYMOUS USER Accesses the System without logging in and therefore no credentials are provided for this user.

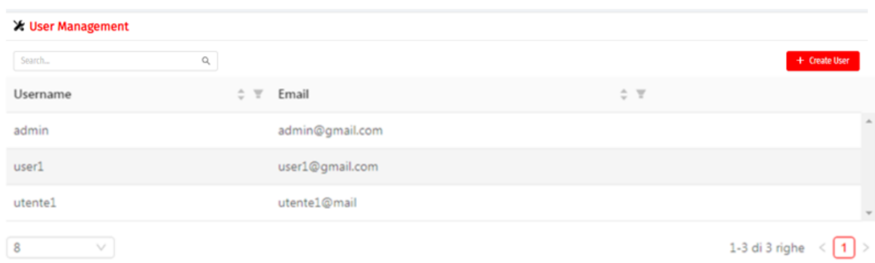
It has only read permissions on the SDMX Artifacts for the various nodes with the possibility of downloading Artifacts in the Meta Manager.
It can perform read-only access (search and display) to DCAT modules.
Does not have access by default to the Data Manager application.

6.3.2 Creating a new user

For user management there is a function that can be activated from the menu on the left-hand side by the administrator user (admin) of the node once the login has been carried out.



Selecting the item User Management the list of users present in the system, if they exist, will be shown.

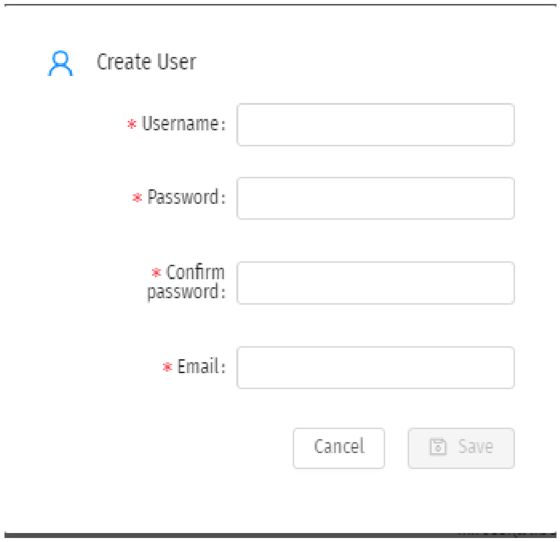


The list of Users defined in the system provides a tabular representation and special features for each list item.
The table shows the columns:

1. **Username** with the user key
2. **E-mail** with the e-mail address associated with the user. For each element of the list there are management functions that can be displayed by positioning the mouse on the element itself.
   The functions available to the Administrator are:
   - *View/Edit* to access the detail screen
   - *Delete* to delete a user

The list of users, if no users have been defined in the system, may be empty.
In this case the SuperUser must create an Administrator user.

Selecting the **Create User** button the form for the acquisition will be opened:



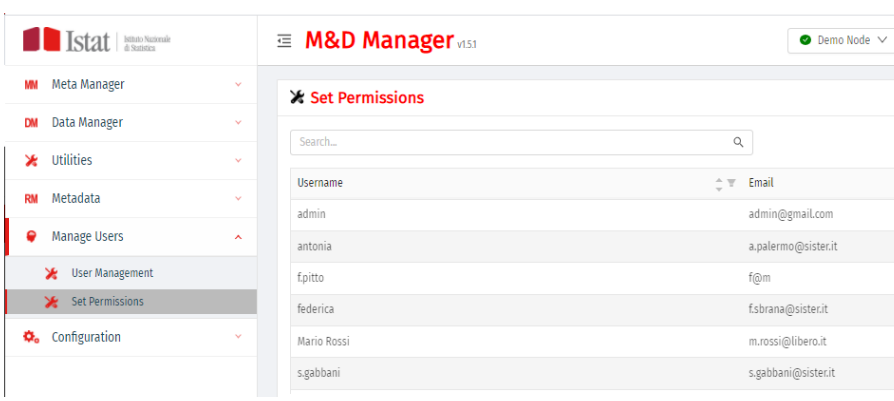
Mandatory fields are those provided for editing:

- *Username* for setting the user name (e.g.: Mario Rossi)
- *Password* for setting the password (e.g. pa$$word)
- *Password* to confirm the user password (e.g. pa$$word)
- *E-mail* to set the user's e-mail address (e.g. mariorossi@gmail.com).

At the end of the compilation, to confirm the creation of the new user, the Administrator must click on the Save button; alternatively click on the Cancel button to cancel the operation and not create the user.
The Save function closes the new user insertion mask and returns to the list of users updated with the new element created.

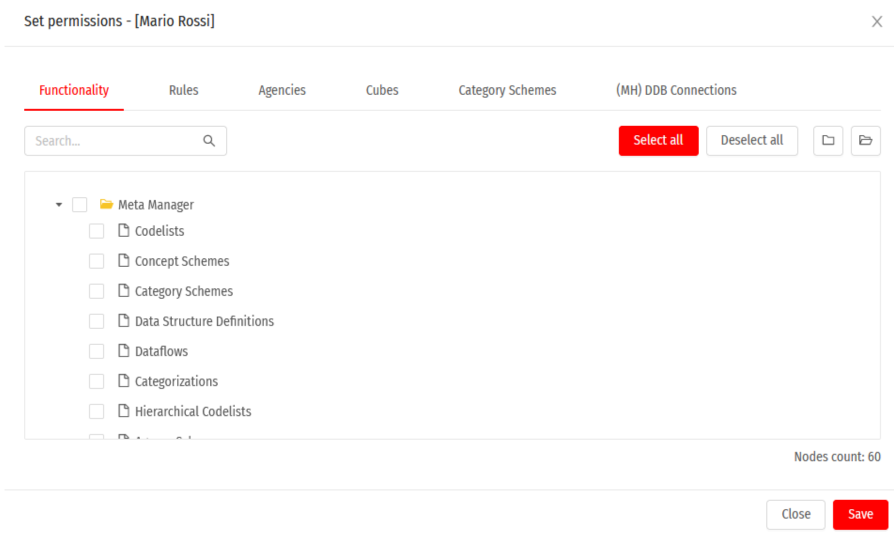
6.3.3 Creating an administrator user

Once the *Login* has been carried out, the Administrator is able to create a *new user* and then profile it.
To profile the new user the Administrator selects the side menu item "Set Permissions".
The following screen is displayed:

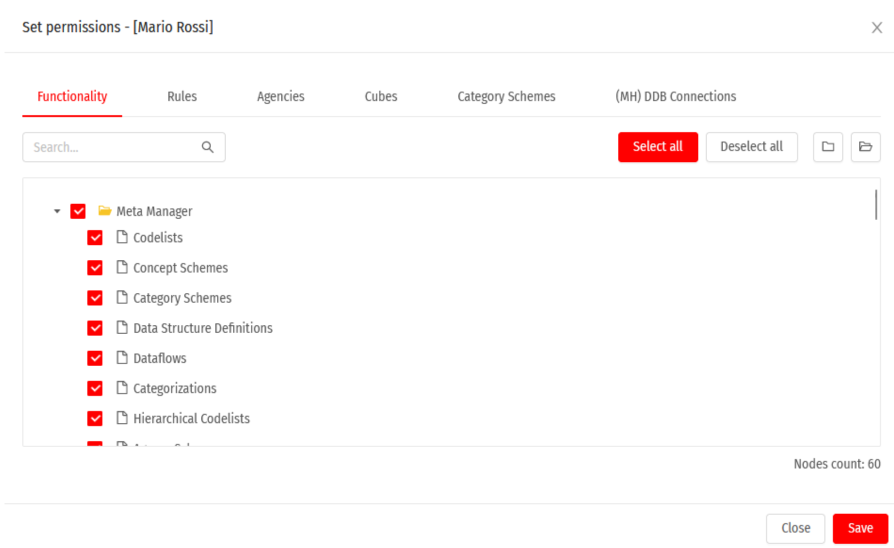


The Administrator has to position the mouse on the user's line (e.g. "Mario Rossi") until he intercepts the "View/Edit" icon and press the left mouse button.

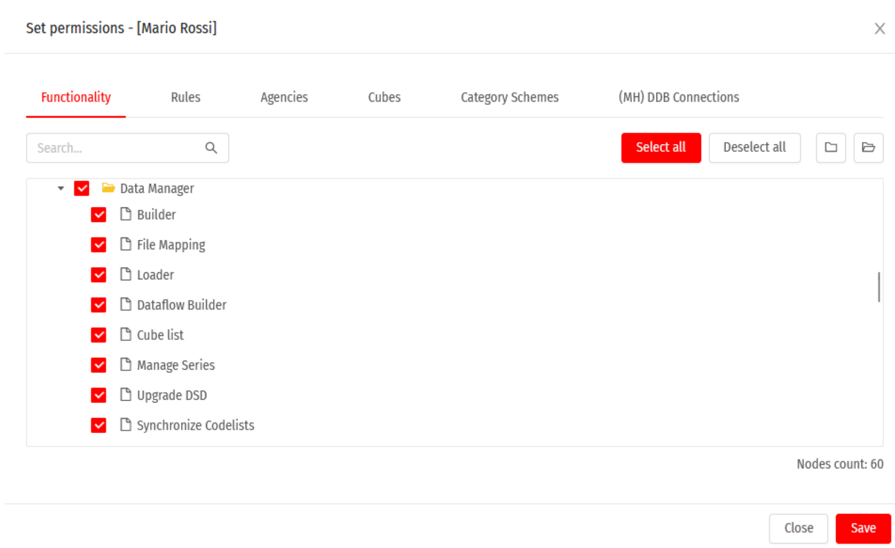
The System displays the user's details, in the mask there are buttons that allow the explosion or compression of the folder representation for the main applications (Data Manager, Meta Manager, etc.).



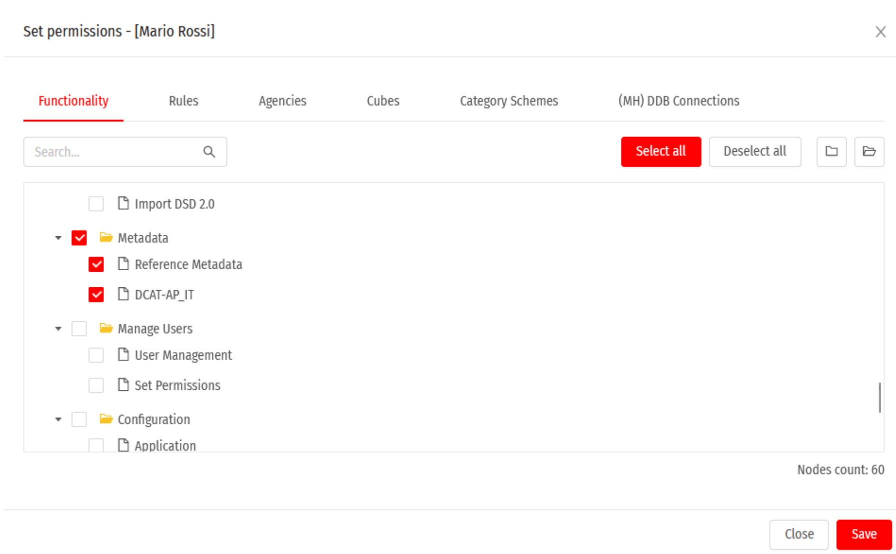
The Administrator who wants to profile the user as an administrator will select all the checks under the Meta Manager folder functionality as shown in the following figure:



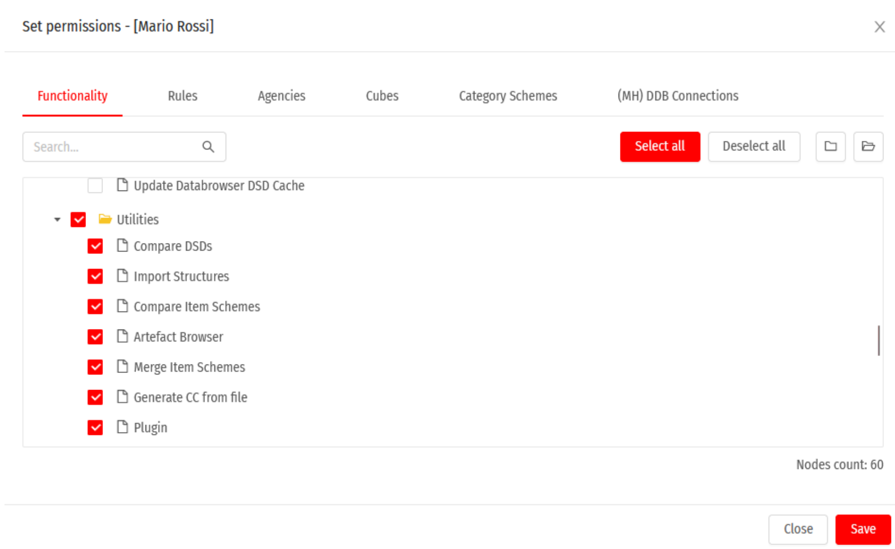
then similarly will select the functionalities of the Data Manager as shown in the following figure:



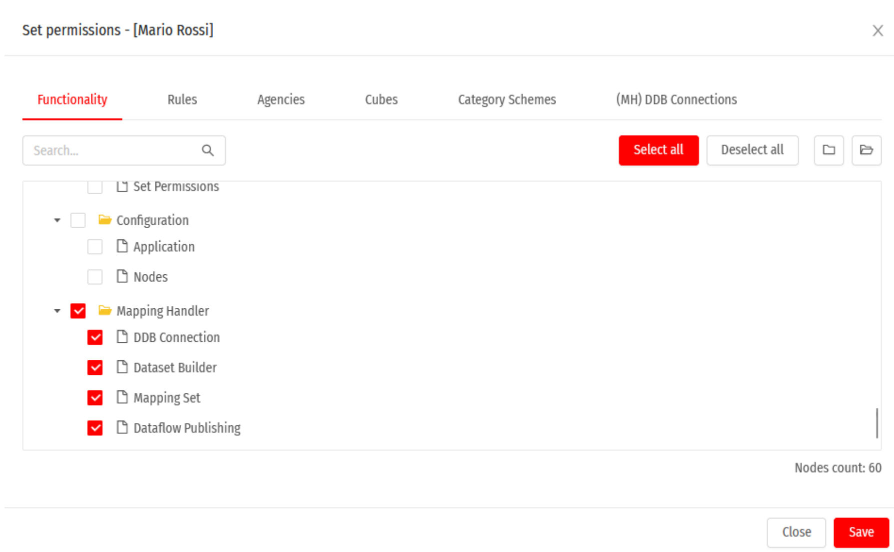
and moreover will select the Metadata functionalities as shown in the following figure:



and then will enable the user to use the tools useful for artifact management:



the administrator could also grant the user permission to use the mapping handler functionalities:



It is also possible to give permissions on user management, although it is advisable to have a single administrator user with user management functions for each node.

The “Rules” tab contains the list of actions that can be done on objects by the user.

Selecting the “Select All” button allows User Mario Rossi to perform all the functions of creation, update, deletion, import, download, display for the whole Suite as Node Administrator.

In the “Agencies” tab, the Administrator can select the Agencies that will be visible to the user when processing an Artifact.
To profile a new administrator user it will be necessary to select them all.

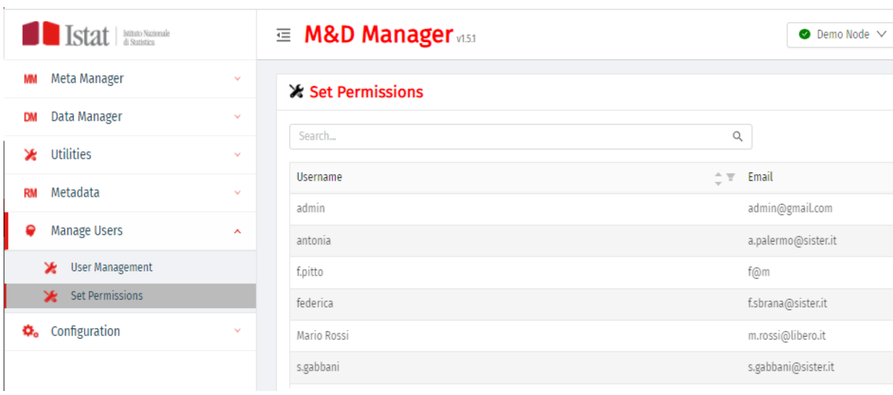
In the “Cubes” tab, the Administrator can select the cubes or the categories of cubes on which the user can work.
To profile a new administrator user it will be necessary to select all the categories.

In the “Category Scheme” tab the Administrator can select specific Category Schemes which will not be shown to other users. In this case the user will not be able to insert dataflows under such categories or view them in the "Category Schemes and Dataflows" in the Meta Manager section.

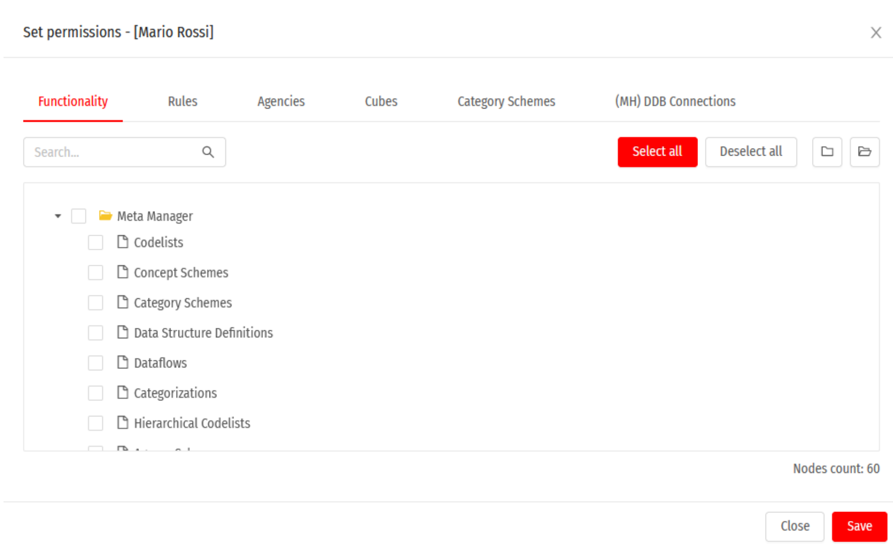
Finally, in the DDB "Connections" tab the Administrator can select specific connections will be shown to the selected user, who will only be able to use those in the Mapping Handler section.

6.3.4 Creating a user dedicated to the Structural Metadata

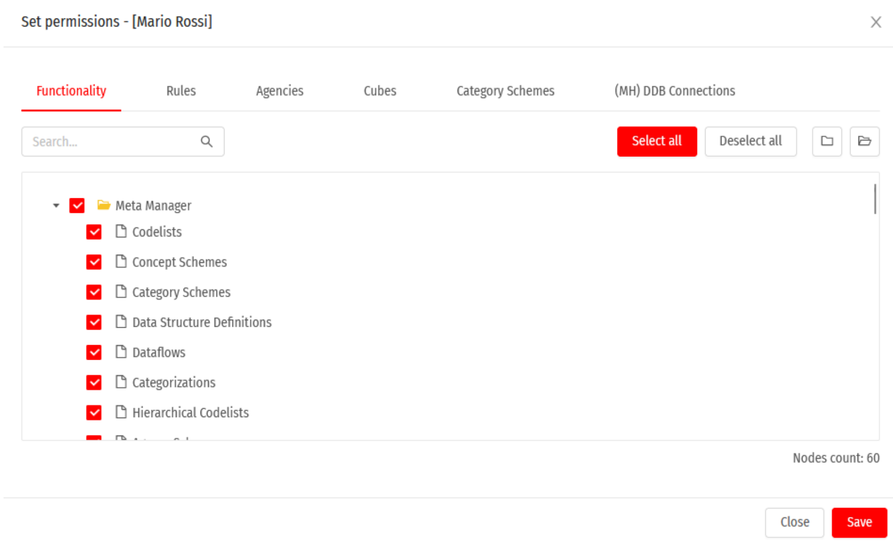
Once the *Login* is done, the Administrator is able to create a *new user* and then profile it.
To profile the new user the Administrator selects the side menu item "Set Permissions".
The following screen is displayed:



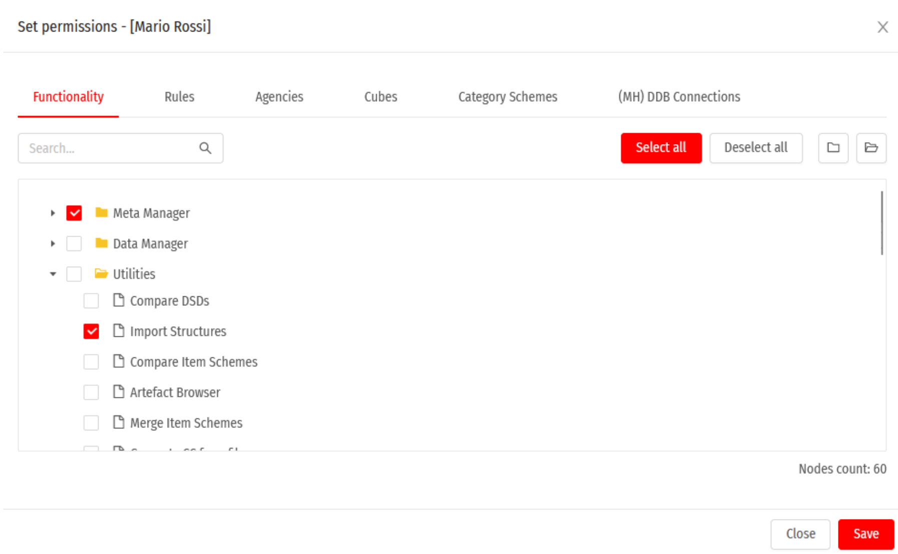
The Administrator has to position the mouse on the user's line (e.g. "Mario Rossi") until he intercepts the "View/Edit" icon and press the left mouse button.
The System displays the user's details, in the mask there are buttons that allow the explosion or compression of the folder representation for the main applications (Data Manager, Meta Manager, etc.).



The Administrator who wants to profile the user as dedicated to Structural Metadata only will select the checks with the Meta Manager functionalities as shown in the following figure:



and then will enable the user to use the tools for managing the artefacts:



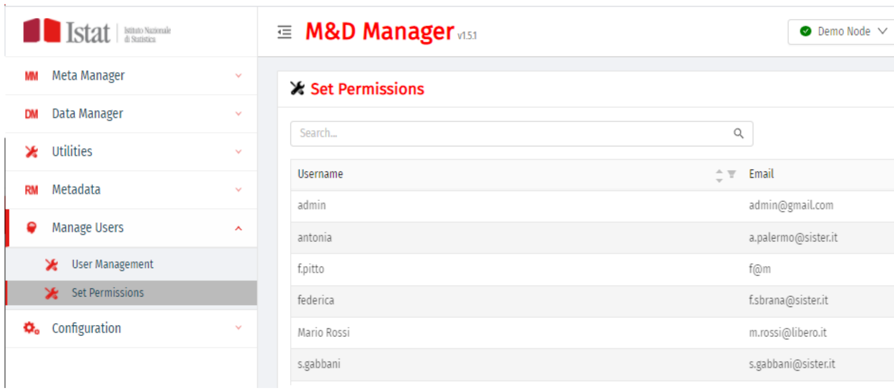
The “Rules” tab contains the list of actions that can be done on objects by the user.

Selecting the “Select All” button and unchecking “AdminRole” allows the user Mario Rossi to perform create, update, delete, import, download, view functions for the Meta Manager application without being a System Administrator.

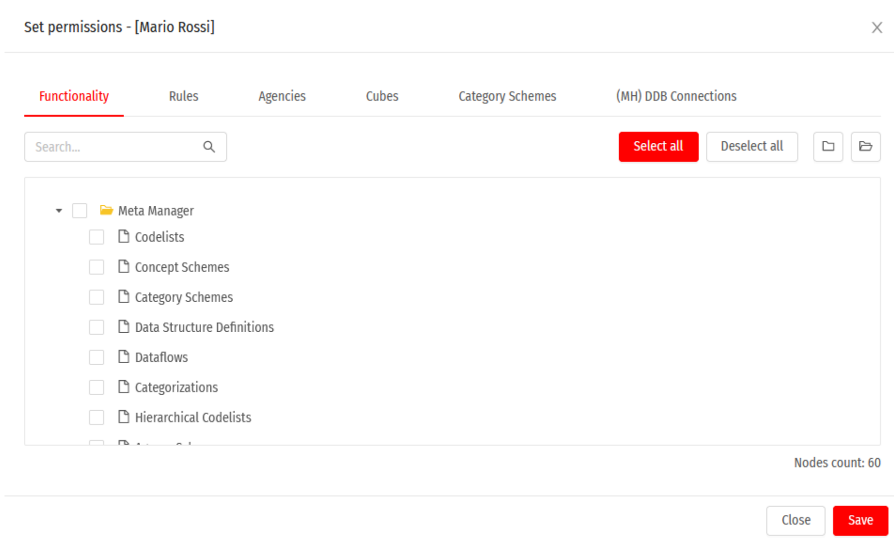
In the “Agencies” tab, the Administrator can select the Agencies to set those that will be managed by the user when processing an Artifact.
The selection of some Agencies means that in the General Data management mask of an Artifact, only the Agencies assigned to the user are present.

6.3.5 Creating a user dedicated to Data Upload

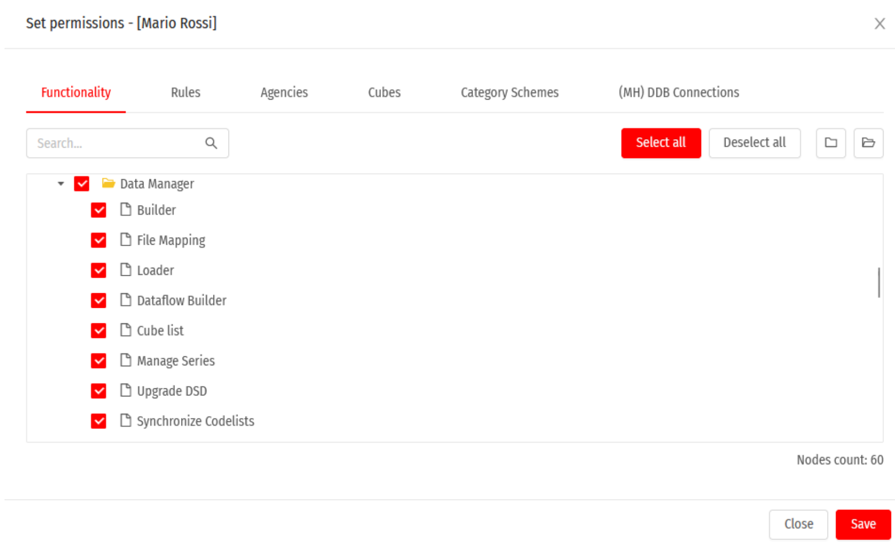
Once the *Login* has been carried out, the Administrator is able to create a *new user* and then profile it.
To profile the new user the Administrator selects the side menu item "Set Permissions". The following screen is displayed:



The Administrator must position the mouse on the user's line (e.g. "Mario Rossi") until it intercepts the "View/Edit" icon and press the left mouse button.
The System displays the user's details, in the mask there are buttons that allow the explosion or compression of the folder representation for the main applications (Data Manager, Meta Manager, etc.).



The Administrator who wants to profile the user as dedicated to loading data only will select the checkbuttons with the Data Manager functions as shown in the following figure:

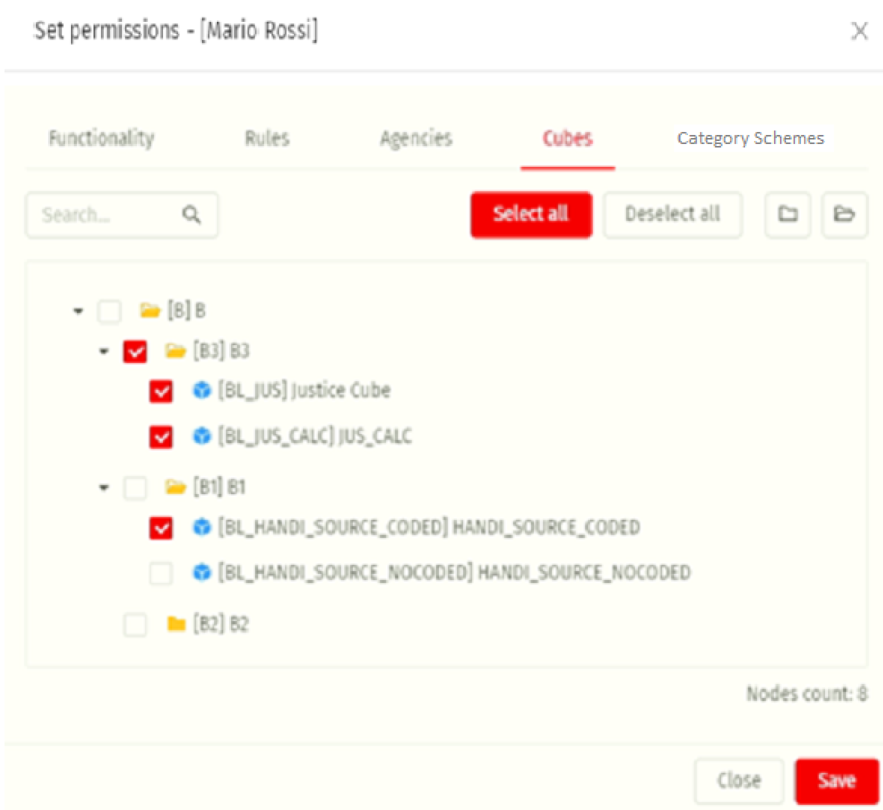


The “Rules” tab contains the list of actions that can be done on objects by the user.

Selecting the “Select All” button and unchecking “AdminRole” allows the user Mario Rossi to perform create, update, delete, import, download, view functions for the Meta Manager application without being a System Administrator.

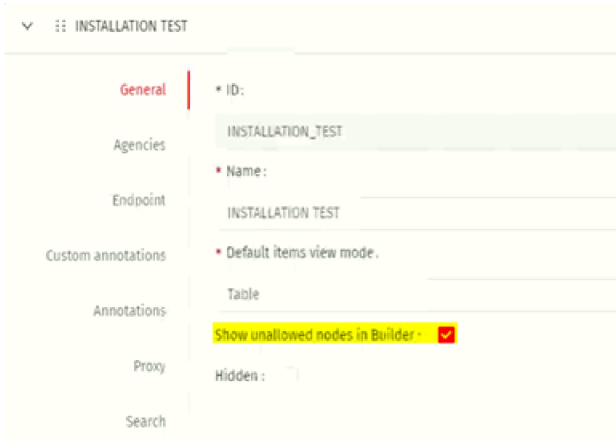
In the “Agencies” tab, the Administrator can select the Agencies that will be managed by the user when processing an Artifact.
The selection of some Agencies means that in the General Data management mask of an Artifact, only the Agencies assigned to the user are present.

Finally, in the “Cubes” tab, the Administrator can select the cubes or the categories of cubes on which the user can work.
Choosing a category will allow the user to work with all the cubes that are or will be part of that category.

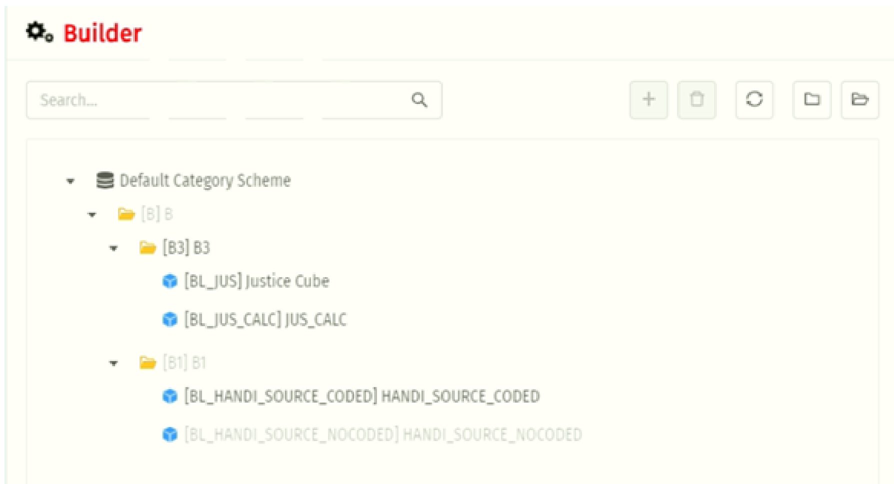


When displayed in the Builder, any folders required to reconstruct the top hierarchy of an object on which the user has permission or categories within which cubes on which the user has permission are categorised will be shown in grey.
By default, cubes that the user cannot work with and categories that the user is not allowed to work with (containing only cubes that the user must not work with) will not be visible.

This is a mode that can be changed in the General section of the node configuration.



By changing the default option then:

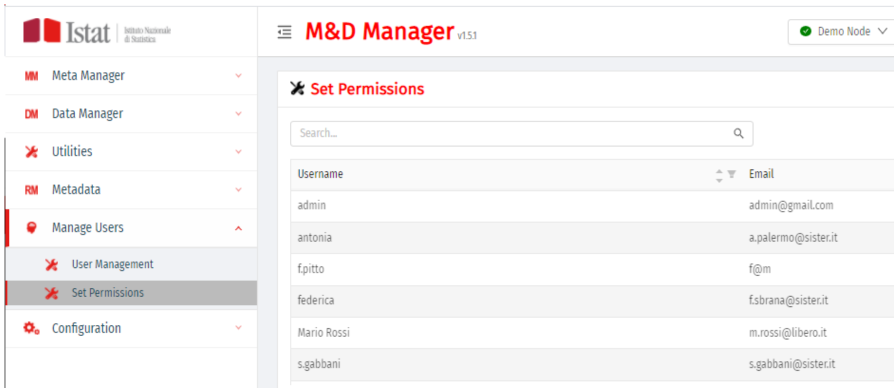


Note

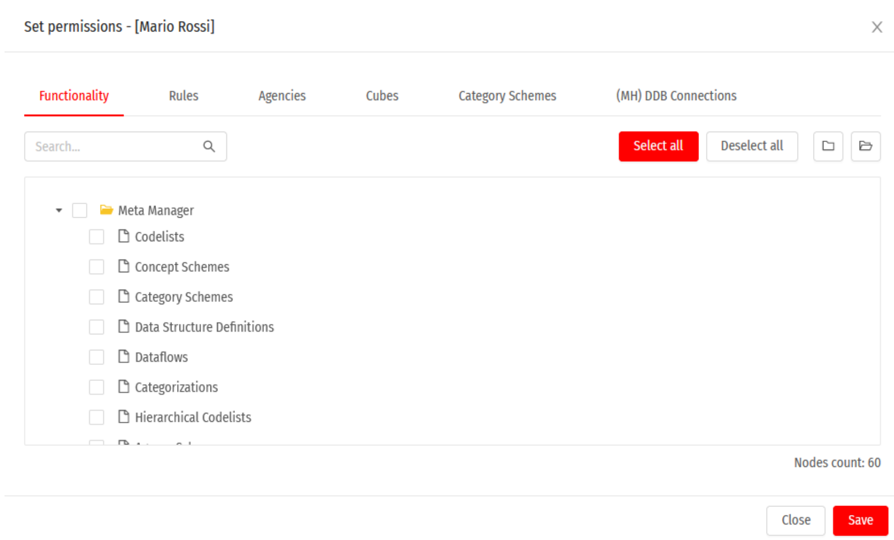
A user with AdminRole permission will not be able to see the category cubes on which he does not have permission but will always see the complete category hierarchy, with nothing greyed out.
This is to allow anyway to manage the Builder's Category Scheme by adding/removing child categories.
On categories on which the user with AdminRole has no permissions, the 'Create cube' option will be disabled.

6.3.6 Creating a user dedicated to Referential Metadata

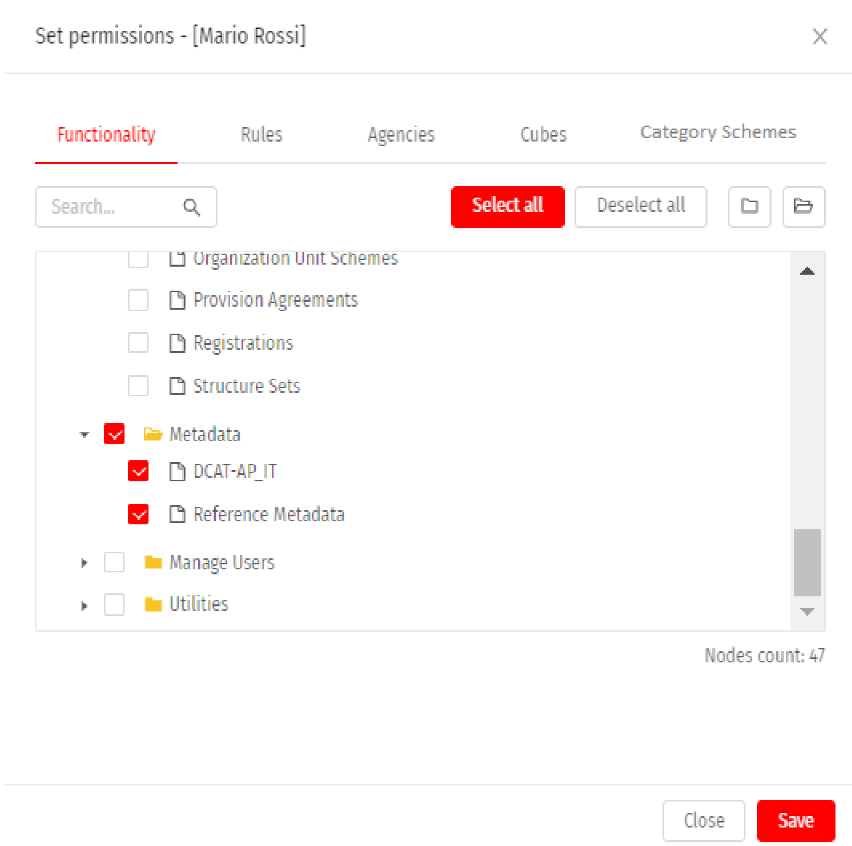
Once the *Login* is done, the Administrator is able to create a *new user* and then profile it.
To profile the new user the Administrator selects the side menu item "Set Permissions".
The following screen is displayed:



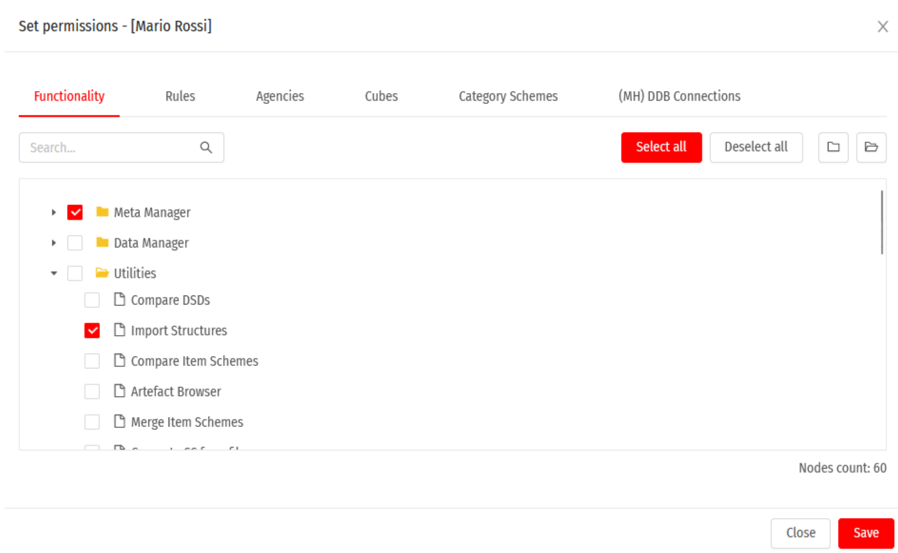
The Administrator has to position the mouse on the user's line (e.g. "Mario Rossi") until he intercepts the "View/Edit" icon and left click on it.
The System displays the user's details; in the mask there are buttons that allow the explosion or compression of the folder representation for the main applications (Data Manager, Meta Manager, etc.).



The Administrator who wants to profile the user as dedicated to Referential Metadata will only select the checkbuttons of some Meta Manager functionalities and all the Metadata functionalities as shown in the following figure:



and then will enable the user to use the 'Import Structures' tool:



The "Rules" tab contains the list of actions that can be done on objects by the user.

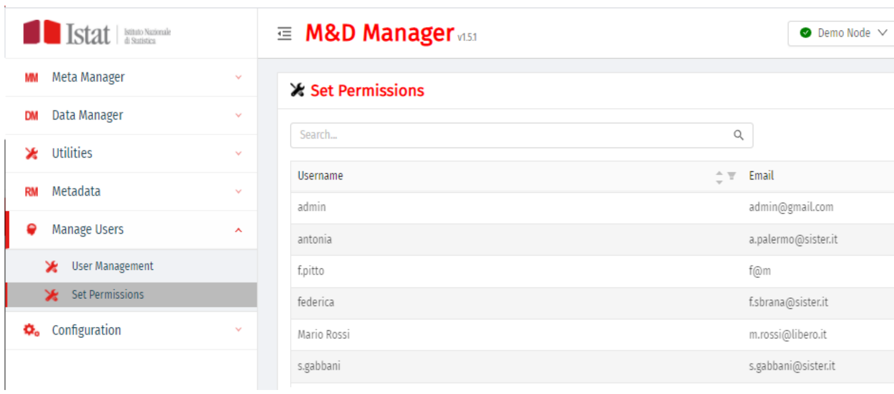
Selecting the "Select All" button and unchecking "AdminRole" allows the user Mario Rossi to perform create, update, delete, import, download, view functions for the Meta Manager application without being a System Administrator.

In the "Agencies" tab, the Administrator can select the Agencies that will be managed by the user when processing an Artifact.

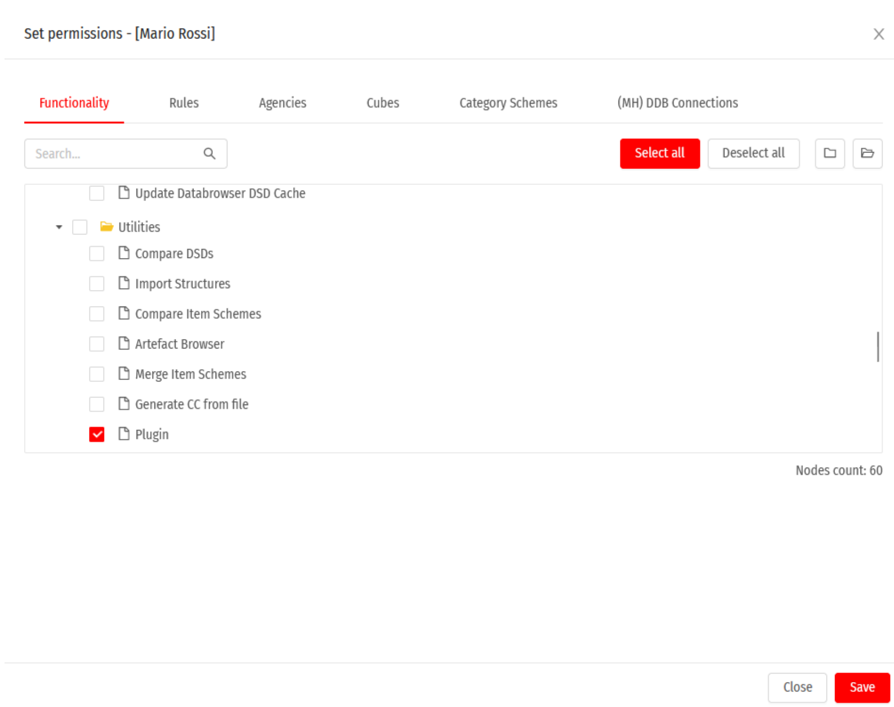
The selection of some Agencies means that in the General Data management mask of an Artifact, only the Agencies assigned to the user are present.

6.3.7 Creating a user that needs to use a plugin

Once the *Login* is done, the Administrator is able to create a *new user* and then profile it.
To profile the new user the Administrator selects the side menu item "Set Permissions".
The following screen is displayed:

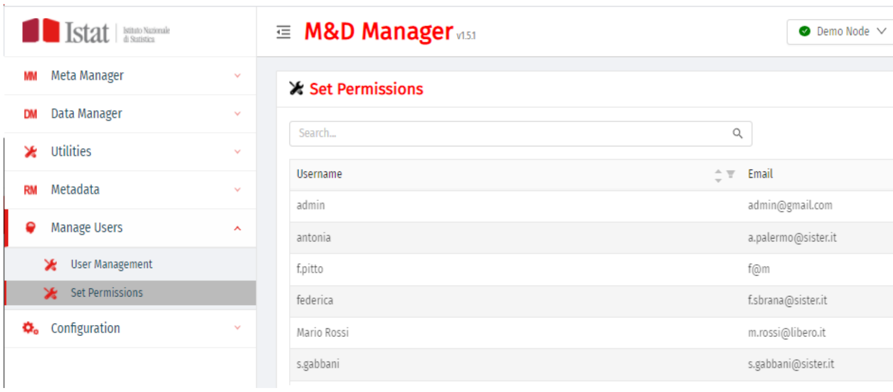


The admin should then look into the functionality tree and find the "plugin" item. If the admin enables it, the user will be able to see all the plugin in the sidebar under utilities (the ones configured to be visible in the sidebar with the "main menu" flag) and will be able to use the plugins in the "file mapping" and "loader" pages (the ones configured to be visible in those pages with the "mapping". flag).

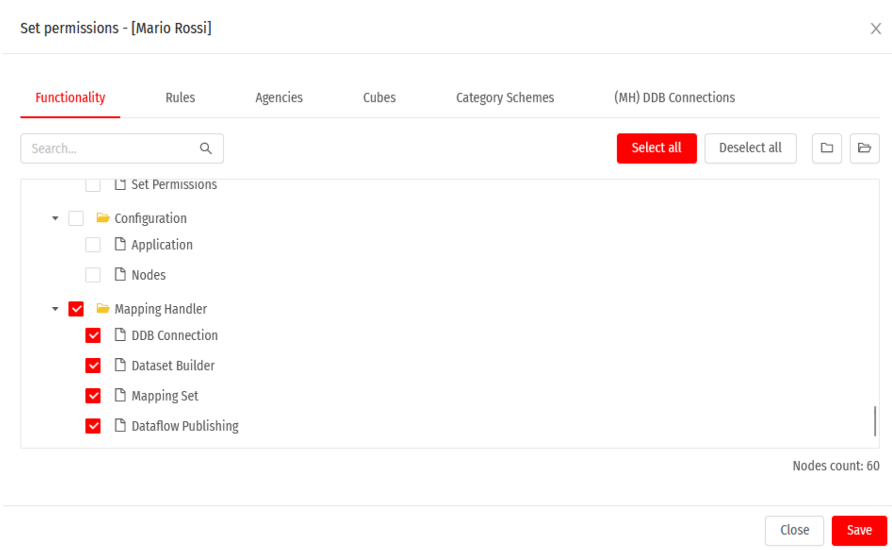


6.3.8 Creating a user dedicated to the Mapping Handler

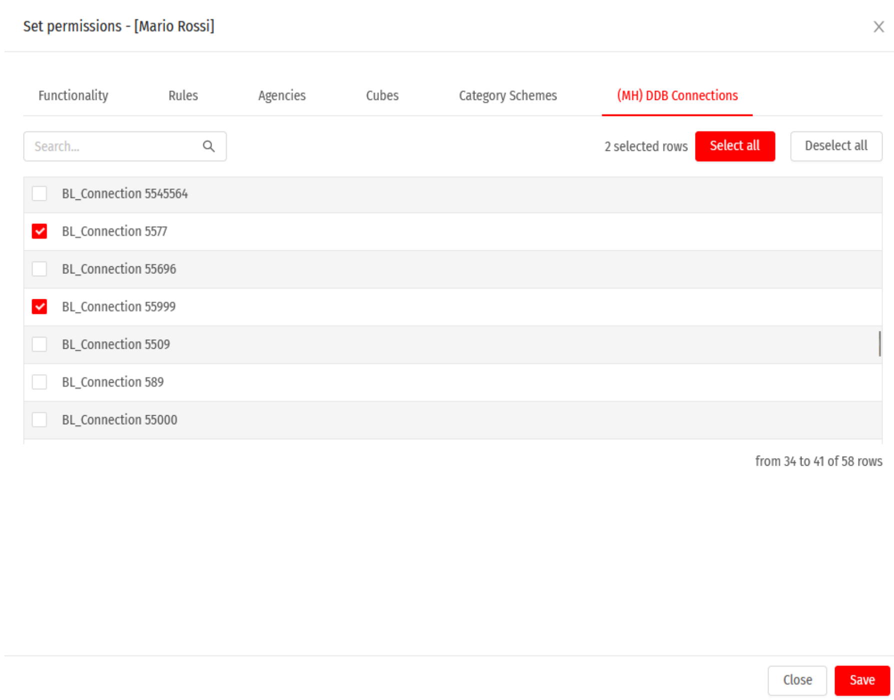
Once the *Login* is done, the Administrator is able to create a *new user* and then profile it.
To profile the new user the Administrator selects the side menu item "Set Permissions".
The following screen is displayed:



The admin should then look into the functionality tree and find the "Mapping Handler" item. If the admin enables it, the user will be able to see all 4 of the mapping handler pages (DDB Connection, Dataset Bilder, Mapping Set and Dataflow Publishing). The admin can choose to disable some of these pages or leave them all enabled



Once this is done, the admin can move to the DDB Connections tab to grant the user specific permissions on which connections the user can access.


Set permissions - [Mario Rossi]
Functionality
Rules
Agencies
Cubes
Category Schemes
(MH) DDB Connections
Search...
2 selected rows
Select all
Deselect all
BL_Connection 5545564
BL_Connection 5577
BL_Connection 55696
BL_Connection 55999
BL_Connection 5509
BL_Connection 589
BL_Connection 55000
from 34 to 41 of 58 rows
Close
Save

CHAPTER

SEVEN

METAMANAGER

Complete management module for SDMX structural metadata (Codelist, Concept Schemes, Category Scheme, Data Streams, DSD, etc.), i.e. how SDMX artefacts, depending on user's profiling, can be created, updated, archived, cloned and deleted.

7.1 Artefact Management

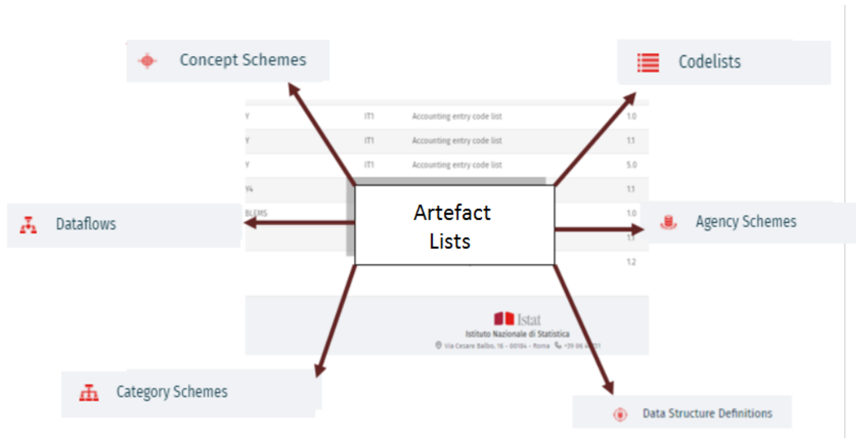
The Meta Manager application is aimed at browsing and managing Structural Metadata according to the SDMX standard.
The main functions are related to the creation and modification, search, navigation, display, import, download of Structural Metadata, called **Artefacts**, through connection to a specific Node.
The **Id**, **agency** and **version** are the three elements identifying an SDMX **Artefact**, independently from the type of artefact we are dealing with.
The **Id** is the identification code of the Artefact.
Agency is the name of the organization that is the creator and/or owner of the Artefact.
Version gives the version of the Artefact.

As they are defined and to favour the usability of the Suite, it has been chosen to standardise some functionalities common to several artefacts:

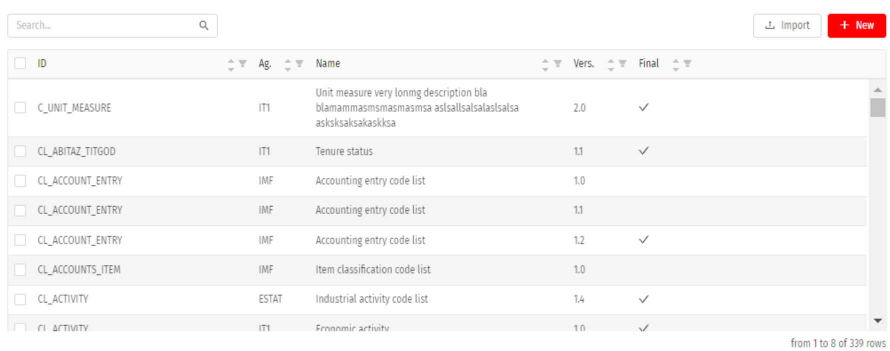
- *Artefact Lists*
- *Search an existing Artefact*
- *Creating a new Artefact*
- *Edit, duplicate and delete an existing Artefact*
- *Import an Artefact*
- *Item Management - Upload CSV file*
- *Download or export an Artefact*
- *Annotations*
- *Sorting Management*

7.1.1 Artefact Lists

From the MetaManager menu, you can select the lists of available artefacts:



The list accessed by selecting an artefact always has the same characteristics.
The lists of artefacts (one for each type of artefact) have a tabular (grid) representation that allows the sorting of elements, the single or multiple selection and the ability to activate elementary functions.

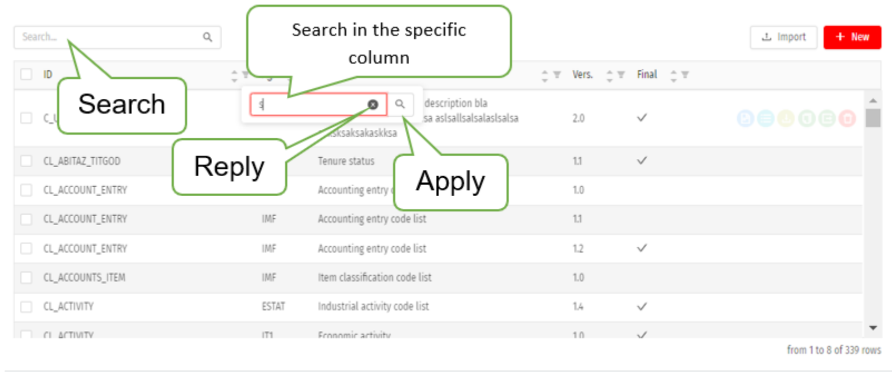


The rows of the list, in "grid" representation, show the list of artefacts while the columns show fields values identifiable through labels inserted in the header of the table.
For each record there are the columns identifying the artefact (ID, Agency, Version and Name) as well as the indication of the finalisation status of the artefact itself.
The system sets a default alphabetical sorting starting from the artefact ID with the possibility to reorder the table according to the sorting buttons provided at column level, the arrows placed next to the column name, in fact, they allow the user to change the sorting from ascending to descending and vice versa.

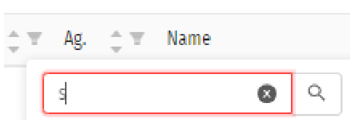
7.1.2 Search an existing Artefact

The search function is necessary for the management of Artefact lists with a high number of elements.

Once the desired *list of artefacts* has been chosen, the search for an artefact in the list can be done in two ways:

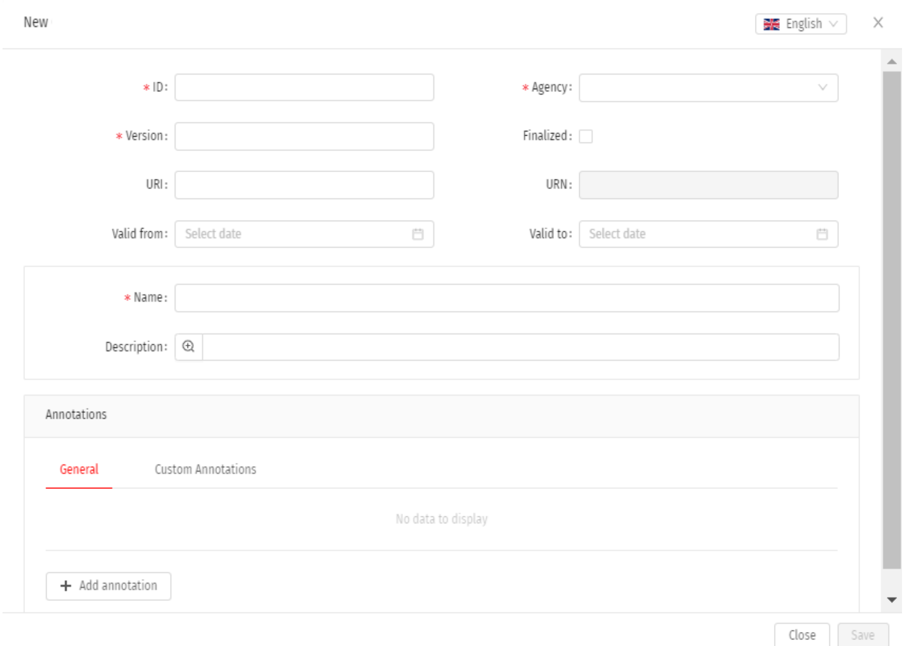


- using the centralised mode within the mask: the function is present in the masks of the Artefacts list management and is represented by an editable field that applies, already in the typing phase, in dynamic mode, the result of the search on the Artefacts list.
- at column level, activating the provided search filters.
  For the columns Agency, Version and Final, the filter shows the list of possible values.
  For the ID and Name columns the filter allows the user sto edit a subset of characters to apply the selection to the Artifacts.
  Once the filter value(s) has been chosen, where applicable, select the Apply function to make the selection valid.
  The coloured filter symbol indicates that a filter is active.
  To remove it, select the filter and then select Reset.



7.1.3 Creating a new Artefact

Once the *list of artefacts* desired has been chosen, to create a new Artefact, the user must select the **New** button in the top right corner.
Regardless of the chosen Artifact type, the management mask of the *General Data* of the Artifact will be opened.



Once the user has filled in the fields of the mask, the System verifies the presence of the compulsory data and the correctness of the data entered while setting them.
Only after completion of the minimum and mandatory information, the System activates the save function using the **Save** button.

7.1.3.1 Section "General"

Independently from the chosen artefact type, the "General" section of the artefact contains the following fields:

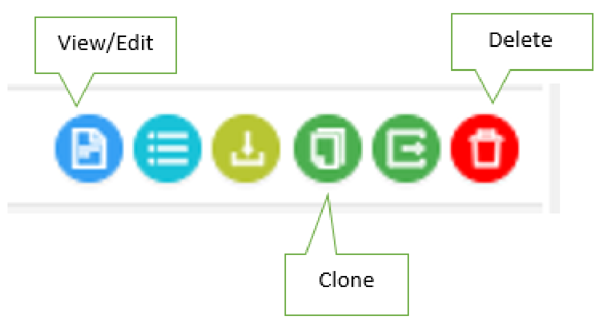
- **ID**: Mandatory alphanumeric field that follows the nomenclature constraints for some artefacts according to SDMX standard.
- **Agency**: Mandatory field which will contain the user's choice from the list containing all the Agencies.
- **Version**: Mandatory numeric field with a standard 9.9.9 format.
- **Finalized**: Represented by a checkbox that allows the user to set the artefact as finalized.
  Finalised artefacts are recognisable by the check mark in the 'Final' field. The behaviour of such artefacts depends on the version of the NsiWebService. If the System is connected to the NSI WS endpoint with a version prior to 6.12. modification is not allowed.
  In case of connection to web services with version higher than 6.12, it is possible also for the finalised artefacts to modify the general and detailed textual information (e.g. Name and description of the artefact, name and

description of the single Items, Annotations at artefact and/or Item level, etc.) and the other general information as provided for by the SDMX ver. 2.1 standard (e.g. Valid from, Valid at, etc.).
Moreover, in case of connection to NsiWebService with version 6.14 or higher, it is possible to add new Items to finalised artefacts but it is not possible to delete them.

- **URI**: Optional alphabetical editable field.
- **URN**: Non-editable field. Automatically set by the System, it is presented in read-only format.
- **Start validity date**: Represented with editing field and calendar for date selection (day, month, year).
- **Final Validity Date**: Represented with editing field and calendar for date selection (day, month, year).
- **Language**: Mandatory field, allows the user to select the language for the insertion of the Name and the Description of the artefact.
It is possible to enter a different Name and Description for each selected language.
At least for one language the name must be entered.
- **Name**: Mandatory multilingual field, identifies the name of the artefact.
- **Description**: Multilingual field for the description of the artefact in multi-line format.
- *Annotations*: Both general and custom, available for all artefacts and Items. This construct allows information to be added to the metadata.

7.1.4 Edit, duplicate and delete an existing Artefact

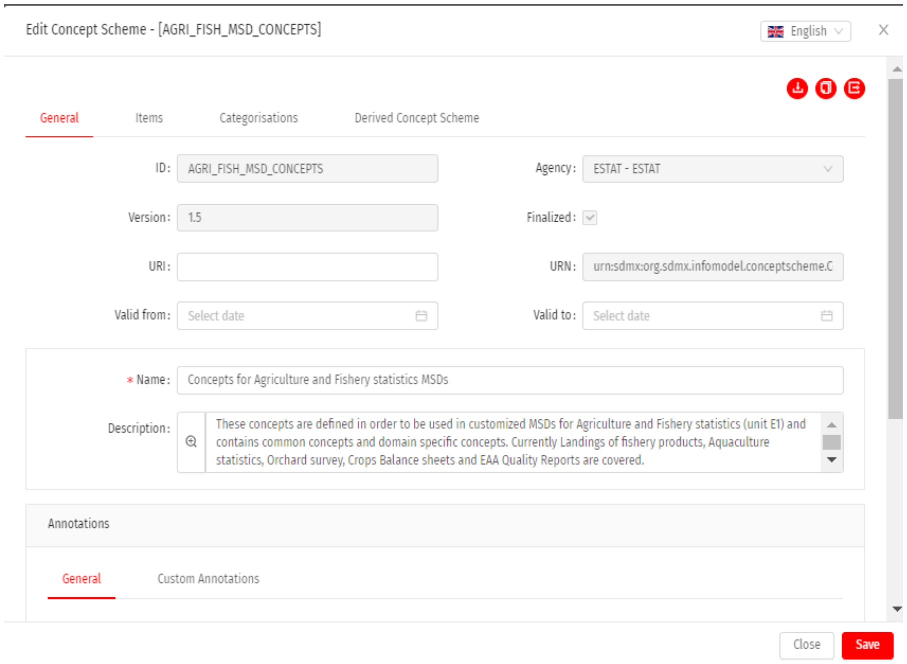
Once the *list of artefacts* has been chosen and the desired *search artefact* has been performed, the user has a series of functions at line level, according to his profiling.



If the user does not have permissions to perform a certain action, the icon is not displayed.

View/Edit
This button allows the user to view the Artifact Detail and update it if the user has write privileges.
Regardless of the type of artifact chosen, the data management mask *General* of the artifact will be opened.
In addition to the **General** tab, which is the same for all Artifacts, each Artifact will

show other sections that must be filled in/edited according to the peculiarity of the Artifact.



In the **General** tab, the identification information of the Artifact that should not be modified will be displayed in read-only format.

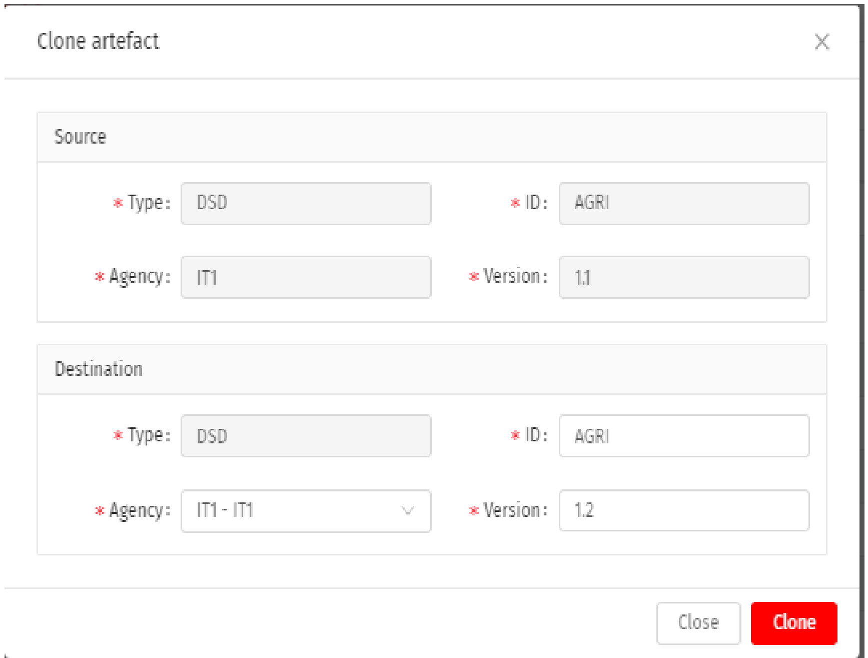
Duplicate

This button activates the function of duplication of an Artifact (it is present both at row level and in the various Artifact management masks).

The system opens a pop-up with the data of the source Artifact and of the destination Artifact:

- *Type* (mandatory, by default equal to the source type, not modifiable)
- *Id* (mandatory, the one of the Origin Artifact is proposed but can be modified)
- *Agency* (mandatory, the one of the Artifact of origin is proposed but can be modified)
- *Version* (mandatory, the version of the Artifact of origin is proposed increased but can be modified)

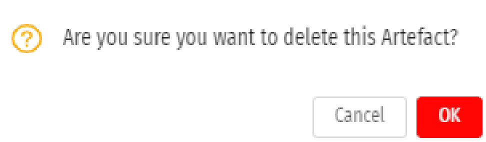
If the user considers the data correct, he can duplicate them using the Duplicate button



Duplicating a finalized Artifact generates an unfinalized Artifact.

Delete
This button activates a pop-up message for user confirmation.

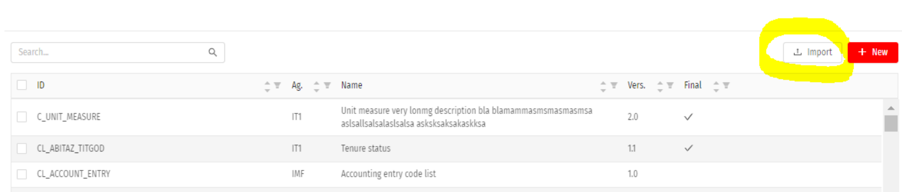
If the user wants to continue with item deletion he will select the confirmation button, otherwise he will select the cancel button.



Following confirmation the System will display the list of Artifacts updated and sorted alphabetically according to the default sorting or to the sorting carried out by the user.

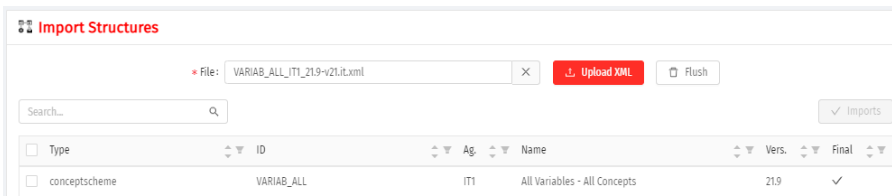
7.1.5 Import an Artefact

The **Import Structures** function is present in the left-hand side menu under "Tools" and also in the list of artefacts.

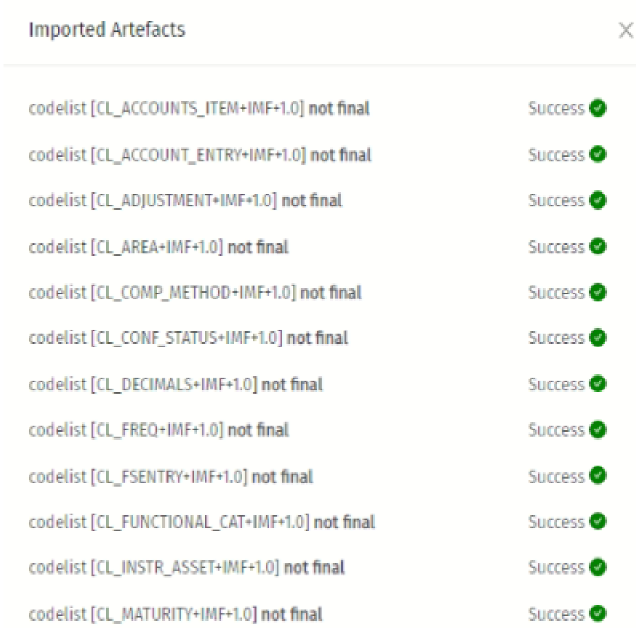


This function allows the user to import structures of SDMX-ML type into the system.

The user selects a file with data to load from *filesystem* .
During the upload function, activated by the selection of the **Load XML** button, the System checks the correct format of the file and the consistency of the information.
In case of anomaly a blocking message is displayed to the user that will not allow the file to be uploaded.
Otherwise the list of artefacts to be uploaded is shown on screen. Those already present in the system cannot be selected.



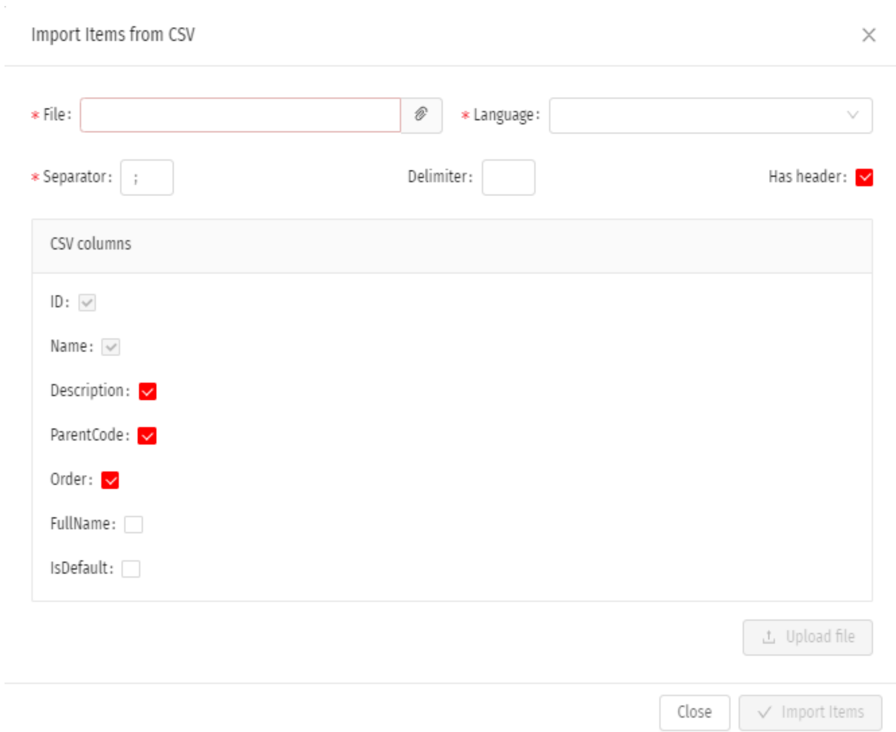
The user can choose which artefacts to import by selecting the line.
By pressing the import button, import ooperation is done and the list of imported artefacts is shown.



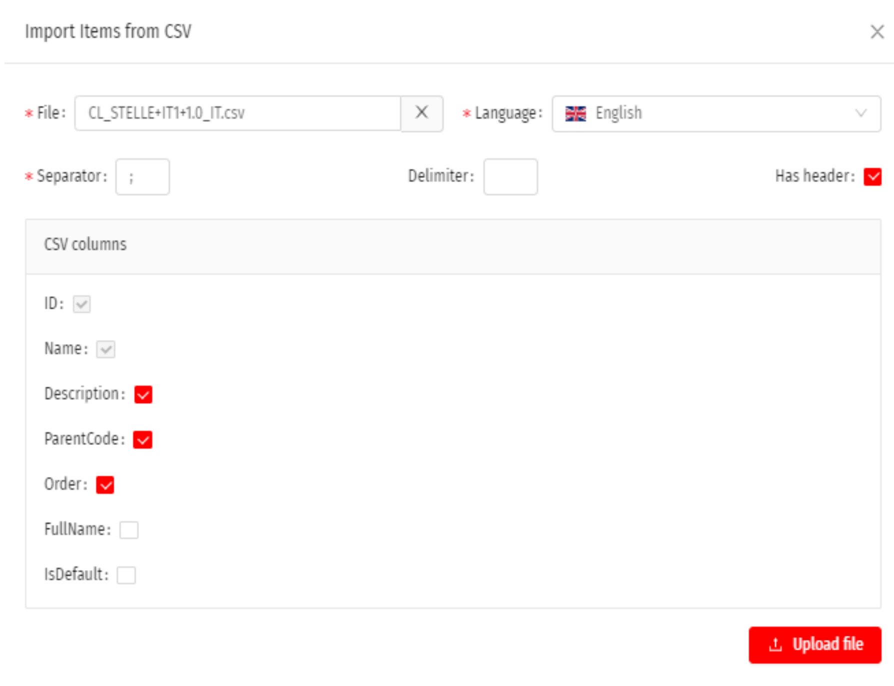
7.1.6 Item Management - Upload CSV file

For artefacts **Concept Schemes**, **Code Lists** and **Category Schemes** it is possible to import new Items from CSV files.
The following example concerns the loading of Items from a Code List.
The technique can also be extended to Concept Schemes and Category Schemes.
From the Item tab the user has to select the Load csv file button.

The system opens a pop-up dialogue window in which the user sets the data required to load the CSV file from the filesystem.
The upload mask contains checks to ensure that the file is imported correctly.



The user selects the csv file and completes the insertion of the required data, setting all the information that corresponds to the content of the file itself in order to avoid errors during import phase.
When the required data has been entered, the **Upload CSV** button becomes active.



Selecting the **Load CSV** button, the System checks the settings and if it does not find a match, it raises an application exception with an error message; otherwise, if it does find a match, the contents of the Code List is previewed in a synthetic table format with a subset of elements.

User can choose to preview the complete contents of the file by selecting the **Full Preview** button or he can select the **Import Items** button to perform the Items import. At the end of the operation, the system notifies the successful import, closes the pop-up dialogue window and updates the list of Items in the Items section of the artefact.

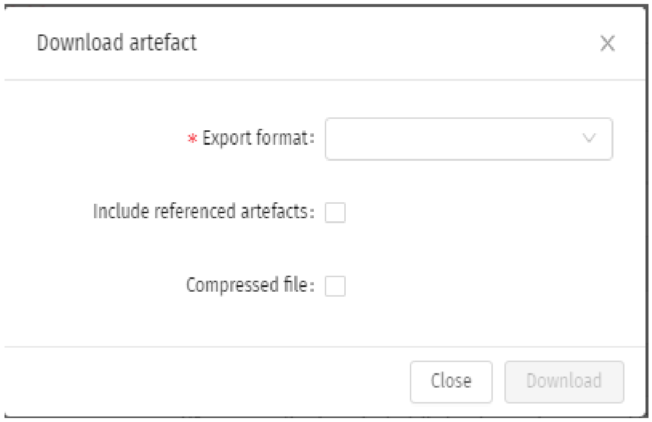
7.1.7 Download or export an Artefact

DOWNLOAD

The Download function is provided for all artefact types.

It is a centralised function that allows the user to download an artefact to file in a specific output format.

The function is activated through a button in the artefact List mask on each single row or from the single artefacts details.



The system opens a pop-up window where necessary data for downloading the selected artefact is set.
The user chooses the **export format** from a list.
The possible output formats are:

- SDMX-ML 2.1
- SDMX-ML 2.0
- CSV
- JSON
- RDF
- RTF

but depending on the type of artefact selected they may not all be available.

The user can download one or more artefacts in compressed format by selecting the **Compressed file** flag (the flag is preset not selected) and depending on the type of artefact it is possible to choose to include also the referenced artefacts by using the **Include referenced artefacts** flag (the flag is preset not selected).
If the user chooses the **CSV** export format, the System opens a form to request further information.
The fields required by the form are:

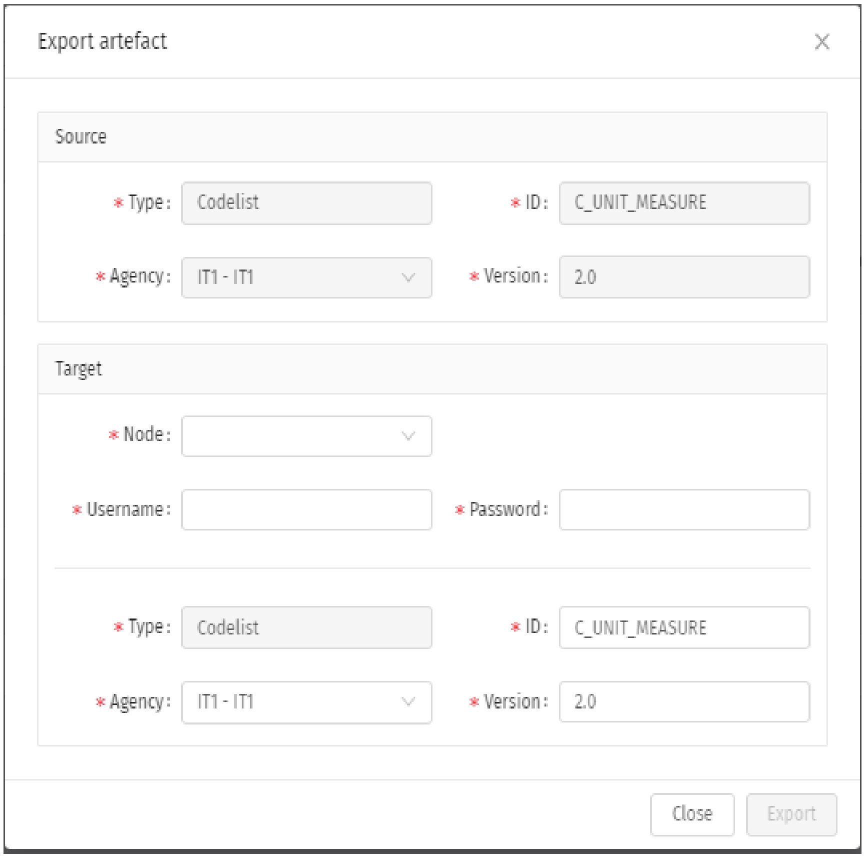
- Language (mandatory)
- Separator (mandatory, default is ';')
- Delimiter (optional)

The system checks that at least the mandatory information has been entered and activates the Download button.

EXPORT TO ANOTHER WS
The function is activated through the button present in the mask of artefacts list on each single line or from the detail of the single artefacts.
The system opens a pop-up window with the data of the source and destination artefacts:

- Node (mandatory)
- Username (required)
- Password (mandatory)
- Type (mandatory, by default the same as the source type, not modifiable)

- Id (mandatory, the one of the source Artefact is proposed but can be modified)
- Agency (mandatory, the one of the source artefact is proposed but can be modified)
- Version (mandatory, the version of the source artefact is proposed increased but can be modified)



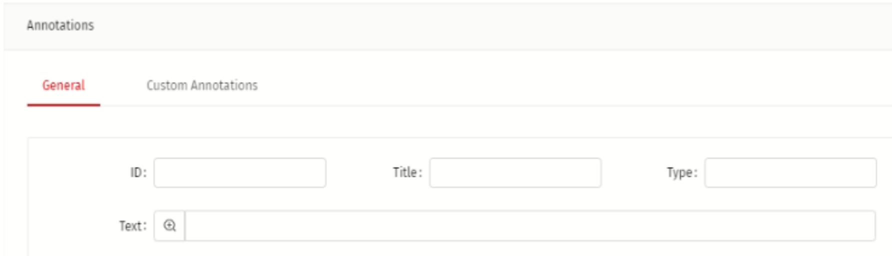
If the user considers that the data is correct, he can export it with the Export button.

7.1.8 Annotations

Annotations are available for all artefacts and items within artefacts.
This construct allows information to be added to metadata.
Individual organisations are free to use them in any way and with any combination as desired, using the field Type as "key" to the type of Annotation.
They can be divided into three major groups:

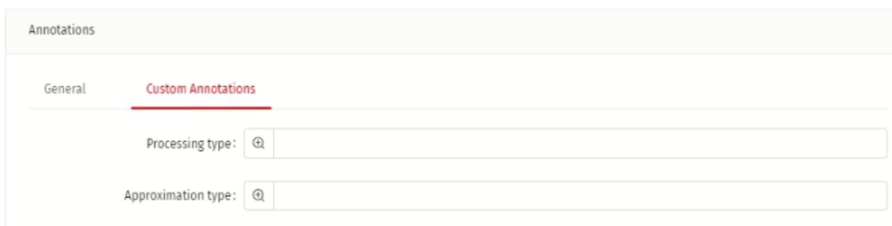
- **General**:
  They are not configured, their presence is not displayed in the List of artefacts and/or List of Items, but to see them and/or insert them the user must access the detail of the artefact or item. During insertion the user must fill in the following items to comply with the sdmx standard:
  - id (mandatory)
  - title (not mandatory)
  - type (mandatory)

- text (not mandatory)

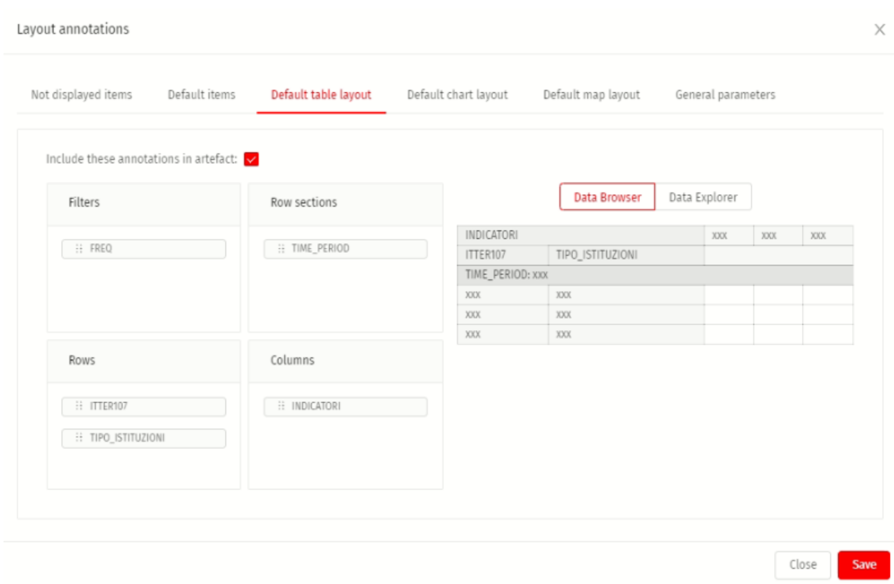


The insertion of these annotations presupposes the existence, outside the Suite, of an application (Data Viewer) capable of interpreting these annotations, with their specific TYPE, according to the needs of the user who inserted them.

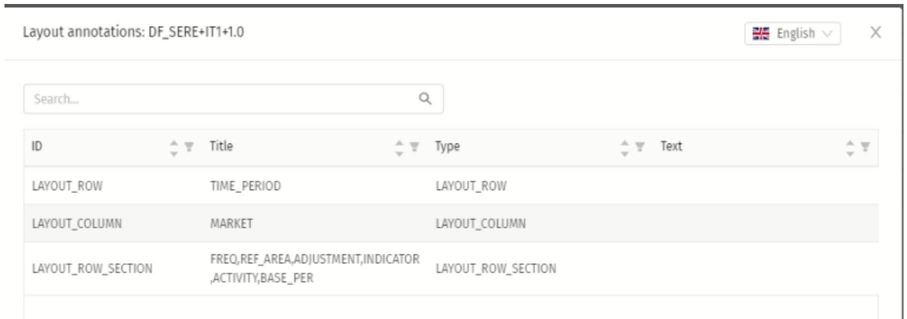
- **Custom**:
  These Annotations are configured at Node level and allow the operator to acquire a collection of different Annotations in a more complex form.
  This type of Annotation, for example, can be used to group together annotations already defined for some specific purpose in order to guide the user in their compilation.



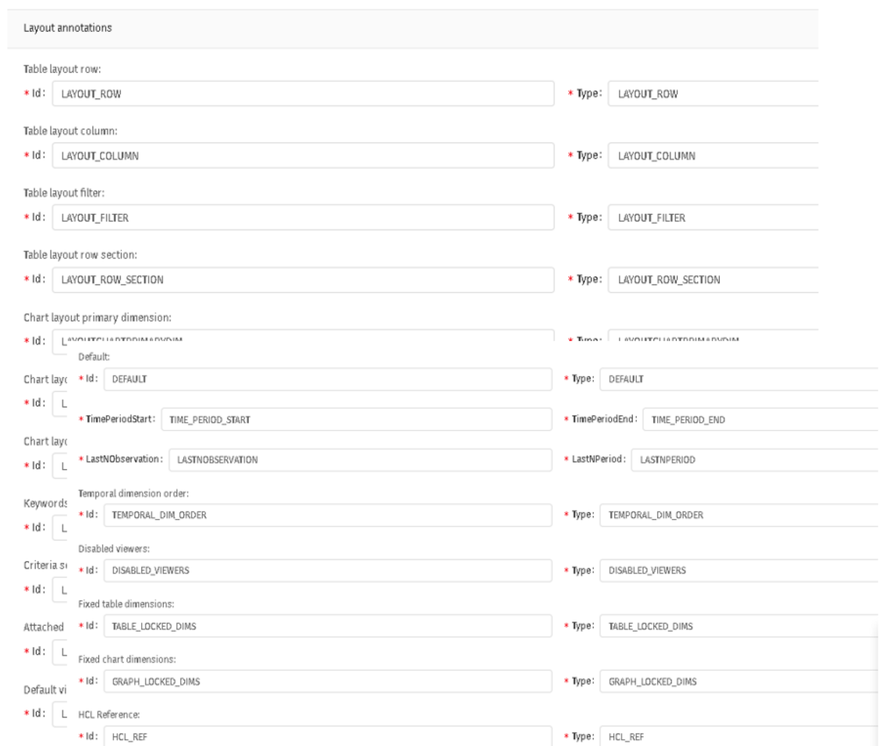
- **Working**:
  This type of annotations are used internally by the system, such as those that define the order of items (e.g. in a codelist).
  Others, e.g. some of those defined by OECD, are used to give indications on the presentation layout of the tabular display of data in an external viewer.



As shown in the above example figure we are not explicitly giving annotation types, it is the System that interprets the (layout) choices made by the operator and adds them to the artefact involved:



the correct annotations among those predefined by the SuperUser at the level of this Node during configuration.



NOTE

Unlike **General** annotations which are not visible in the artefact lists, **Custom** and **Work** annotations defined for an artefact are shown in the artefact list management mask at row level, with white symbol for Custom Annotation and orange symbol for OECD type annotation:

.

7.1.9 Sorting management

The ORDER annotation (which can be properly configured) is used for managing the sorting of Code Lists, Concept Schemes and Category Schemes, and is multilingual. However, to optimise performance, the software behaves as follows:

- for final and non-final artefacts, the ORDER annotation will always be present when created. Nevertheless when an artefact is imported without ORDER annotation, this will not be added automatically.
- The ORDER annotation in a given language will only be added when the order in that language is changed.
- When the ORDER annotation is added for a language it will be added for all items in the related itemscheme.

When an artefact is cloned, the ORDER annotation (if present) is inherited as well.

7.2 Artefacts in detail

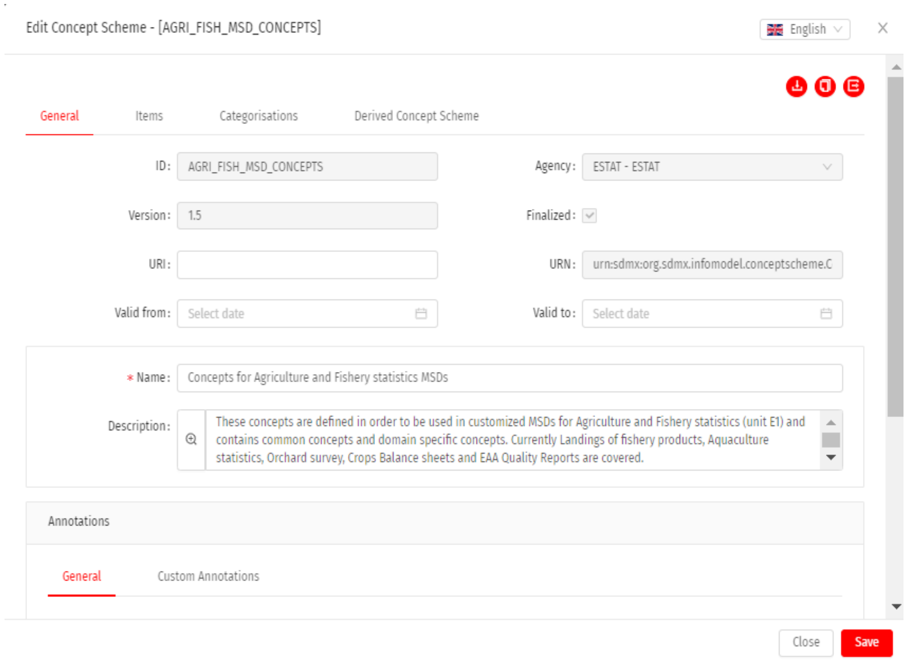
Each artefact has its own peculiarities according to its type, let's see which ones.

- *Concepts Schema*
- *Code List*
- *Data Structure Definitions*
- *Dataflows*
- *Category Schema*
- *Categorizations*
- *Hierarchical Code Lists*
- *Agency Schema*
- *Data Provier Schemes*
- *Data Consumer Schemes*
- *Organization Unit Schemes*
- *Content Constraints*

- *Structure Sets*
- *Provision Agreements*
- *Registrations*
- *Category Sets and Dataflows*
- *Metadata Structure Definitions*
- *Metadataflows*

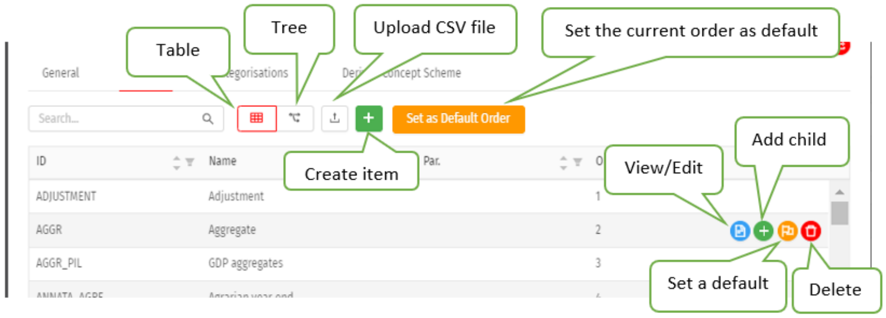
7.2.1 Concept Schemes

Selecting **Concept Schemes** from the "Meta Manager" module, the application displays the list of Concept Schemes of the Node similarly to the lists of all artefacts present, as explained in *artefact list*.
Selecting an item in the list with the mouse and pressing the View/Edit button, the system displays the Detail mask.



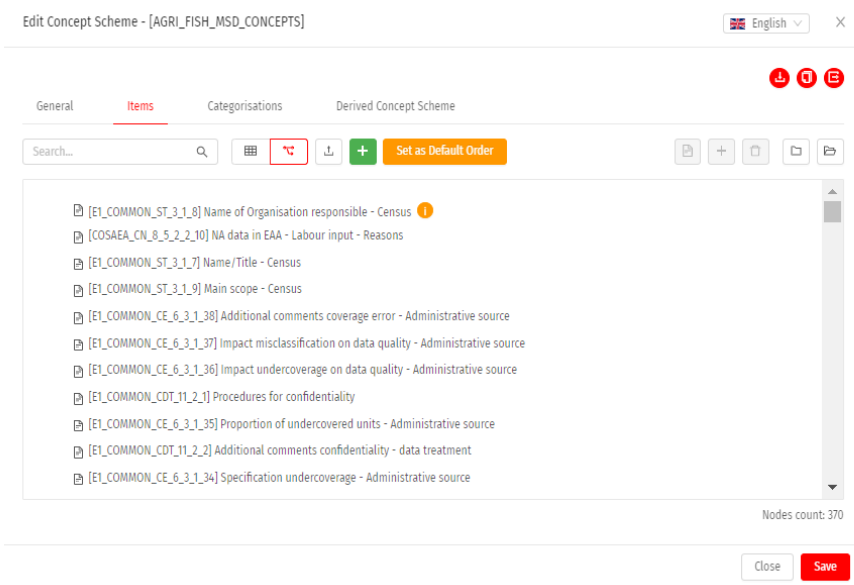
The main section of "Concept Schemes" detail mask contains the following tabs:

- *General*
- **Items:** Items can be displayed in grid or tree mode. By default the grid view is proposed.

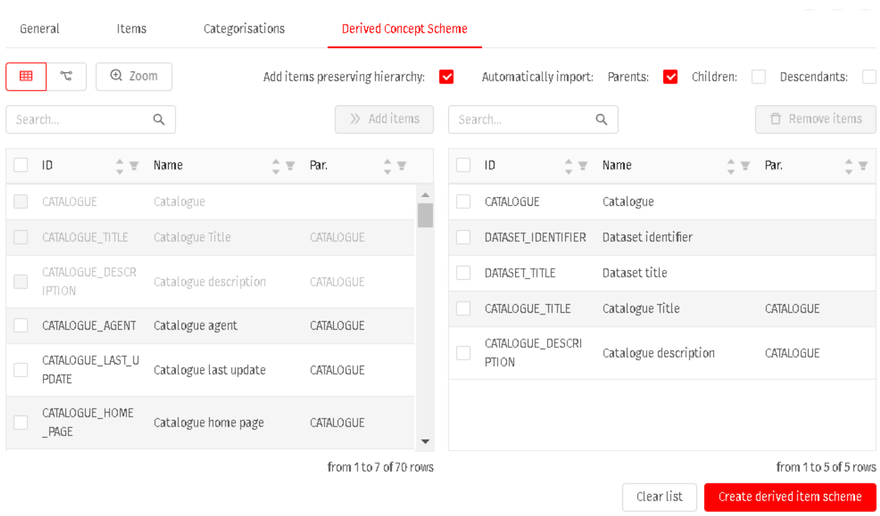


As an alternative to the insertion of the single items, it is possible to *import a CSV file* in order to insert them in bulk, otherwise the management functions of the single items are the following:

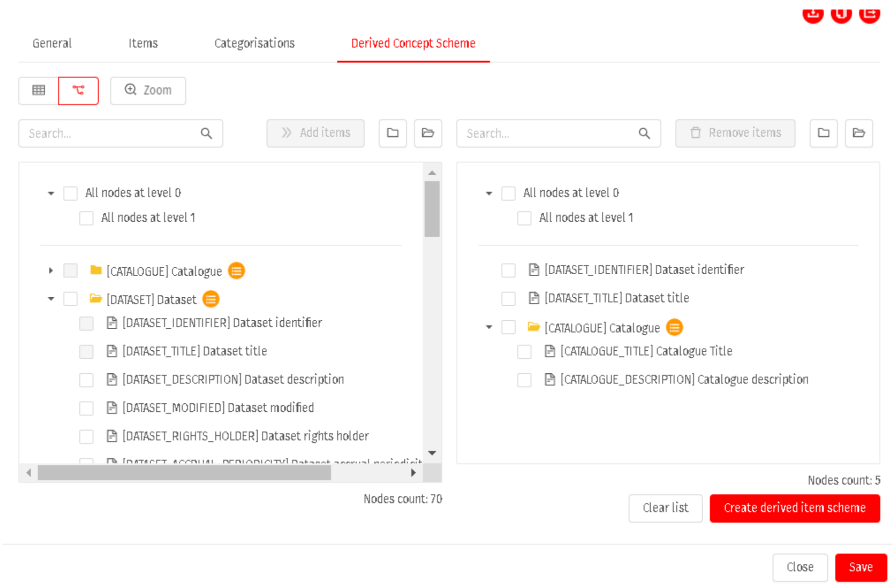
- **Create Item**
  To create a new Item, the user must use the *Create Item* button at the top.
  The System presents the data acquisition mask.
  The fields to be populated are the following:
  – **Identifier Item** (mandatory - alphanumeric)
  – **Language** (mandatory - alphanumeric)
  – **Item name** (required - alphanumeric)
  – **Description** (optional - alphanumeric)
  – **Fullname** (optional - alphanumeric)
  – **Order** (optional - numeric)
  – **Parent** (optional - from a list of items already present)
  – *Annotations* (optional - alphanumeric)
  The new Item is displayed as a record of the grid table and the possible information related to the "Parent Code" is valorized in a column of the grid view.
- **Add Child**
  The *Add Child* button is activated on the "Parent" row and opens a mask with the fields indicated in **Create Item** where obviously the *Parent* field will be preset with the value of the selected parent.
  The new Item is inserted with indentation under the parent code in the "folder" representation.
  The mask contains the buttons for system acquisition: **Save:** to save and close the mask and **Close:** to close the mask without saving.
  In the folder view, the order and the parentship of an item can be determined by drag-and-drop operations of the item itself.
  The user can switch to the folder view by clicking on the "Tree" button.



- **Categorization**
  (not yet implemented)
- **Derived Concept Scheme**
  This tab allows the creation of a new Concept Scheme from a subset of items of the starting scheme, hence the adjective "derived".
  In choosing the elements of interest in table view mode, the user is helped by tools that allow him to take into account or not the hierarchies between parent elements, child elements, etc.

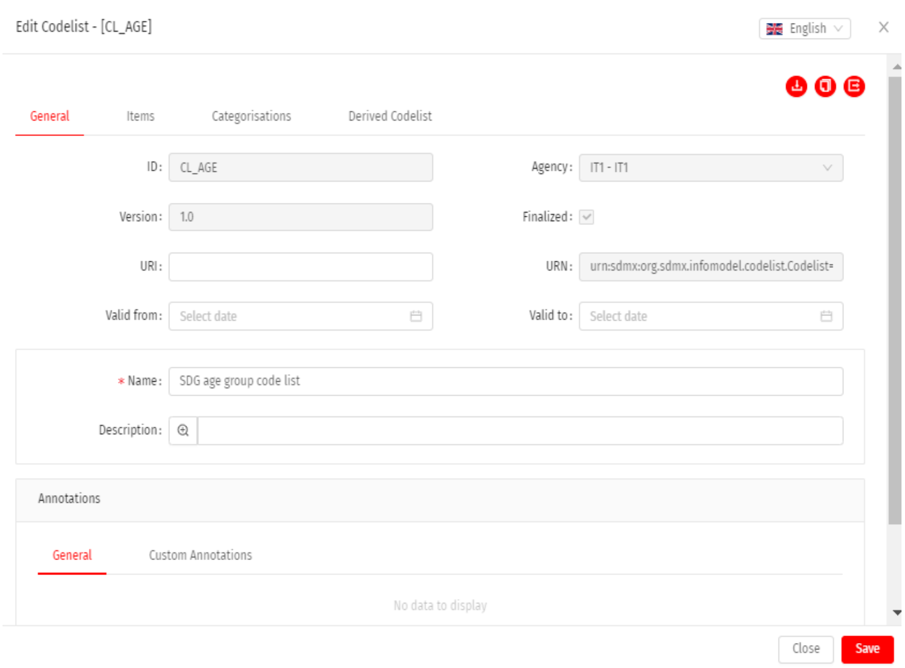


If tree mode is used it will be possible to select:
– a specific node
– a specific sub-level
– all children of a specific node
in addition, the zoom functionality will display the two artefacts in a full screen popup (left and right of the screen) allowing selection.



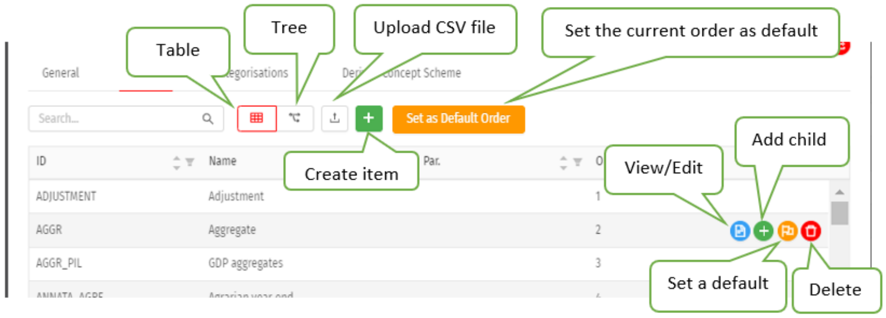
7.2.2 Code Lists

Selecting **Code Lists** from the "Meta Manager" form, the application displays the list of Node Code Lists similar to the lists of all artefacts present, as explained in *artefact list*.
Selecting an item in the list with the mouse and pressing the View/Edit button, the system displays the Detail mask.



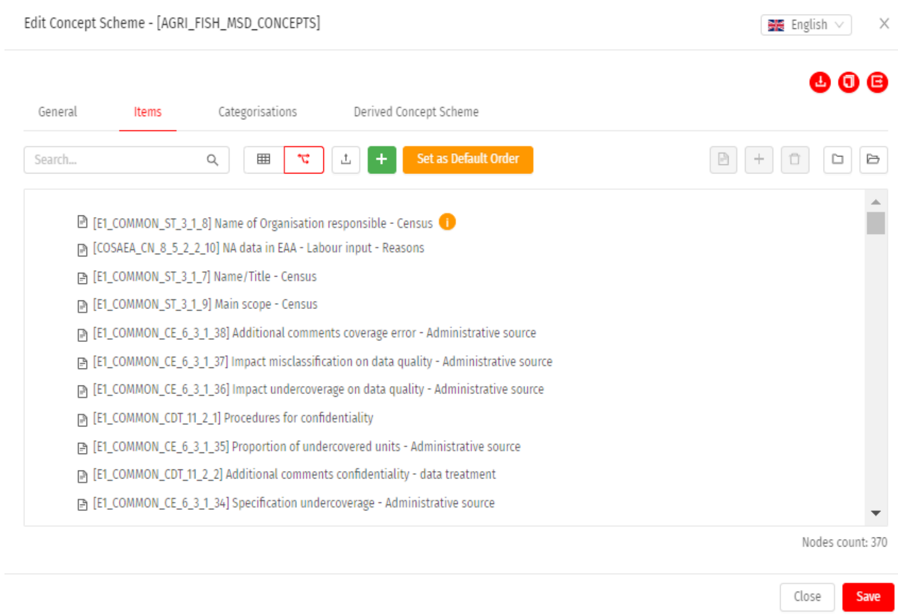
The main section of the Code List detail mask contains the following tabs:

- *General*
- **Items**
  Items can be displayed in grid or tree mode. The default is the grid view.



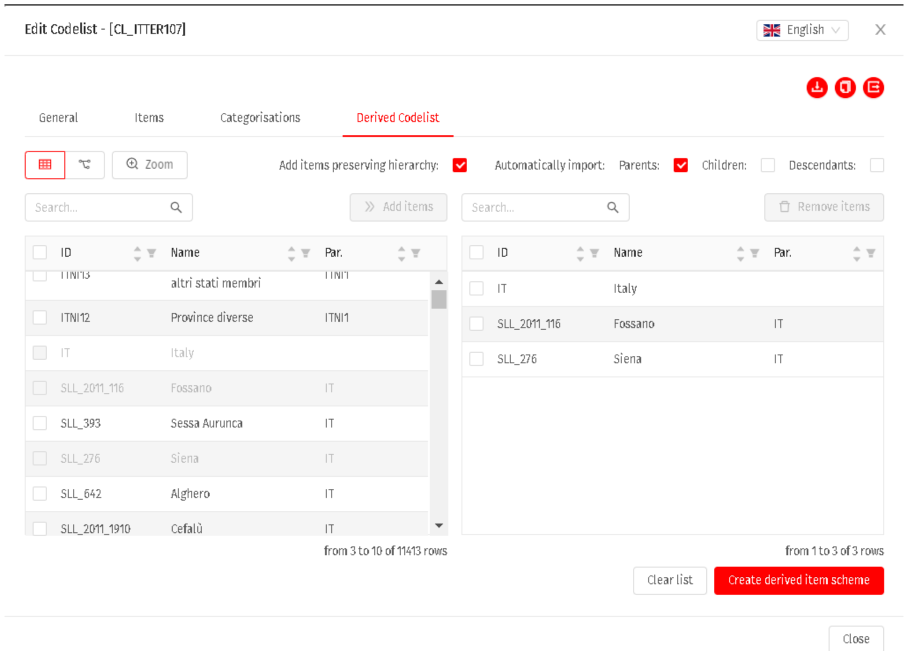
As an alternative to the insertion of individual items, it is possible to *import a CSV file* in order to insert them in bulk, otherwise the management functions of the individual items are the following:

- **Create Item**
  To create a new Item, the user must use the *Create Item* button at the top.
  The System presents the data acquisition mask.
  The fields to be populated are:
  – **Identifier Item** (mandatory - alphanumeric)
  – **Language** (mandatory - alphanumeric)
  – **Item name** (required - alphanumeric)
  – **Description** (optional - alphanumeric)
  – **Fullname** (optional - alphanumeric)
  – **Order** (optional - numeric)
  – **Parent** (optional - from a list of items already present)
  – *Annotations* (optional - alphanumeric)
  The new Item is displayed as a record of the grid table and the possible information related to the "Parent Code" is valorized in a column of the grid view.
- **Add Child**
  The *Add Child* button is activated on the "Parent" row and opens a mask with the fields indicated in **Create Item** where obviously the *Parent* field will be preset with the value of the selected parent.
  The new Item is inserted with indentation under the Parent code in the "folder" representation.
  The mask contains the buttons for system acquisition: **Save:** to save and close the mask and **Close:** to close the mask without saving.
  In the folder view, the order and the parentship of an item can be determined by drag-and-drop operations of the item itself.
  The user can switch to the folder view by clicking on the **Tree** button.

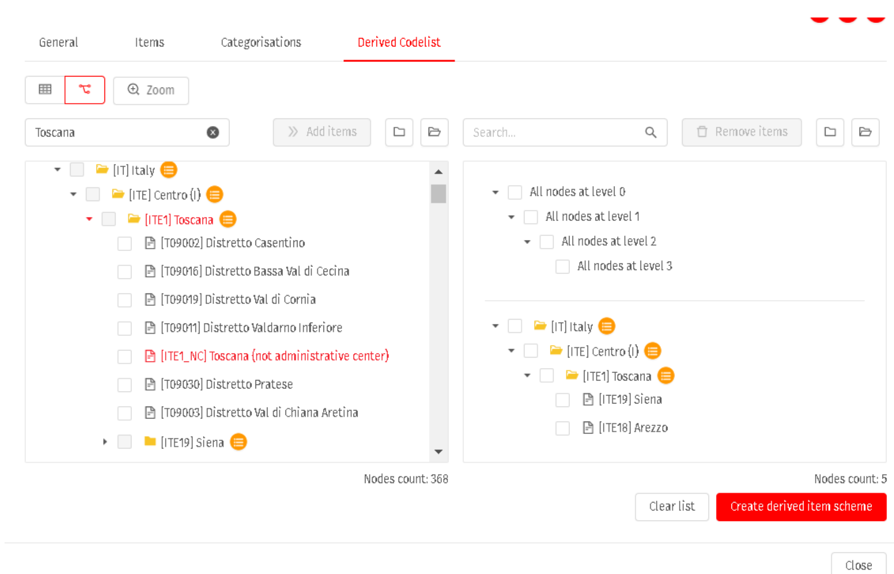


Note:
During the Capture or Modification phase of a Code List Item, the System is set with **Automatic Save**.
The user can in any case set the saving as manual through an ON/OFF button.
If the user disables the automatic saving, he has to save the inserted or modified Items with the Save button otherwise he loses the modifications made.

- **Categorization**
  (not yet implemented)
- **Derived Codelist**
  This tab allows the creation of a new Code List from a subset of items of the starting list, hence the adjective "derived".
  In choosing the elements of interest in table view mode, the user is helped by tools that allow him to take into account or not the hierarchies between parent elements, child elements, etc.



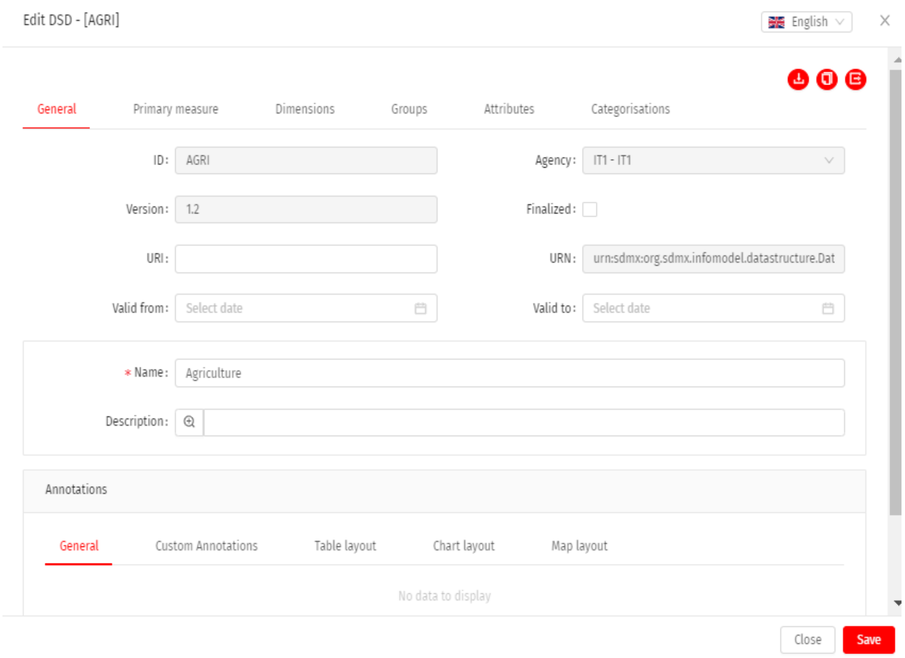
If tree mode is used it will be possible to select:
– a specific node
– a specific sub-level
– all children of a specific node
in addition, the zoom functionality will display the two artefacts in a full screen popup (left and right of the screen) allowing selection.



7.2.3 Data Structure Definitions

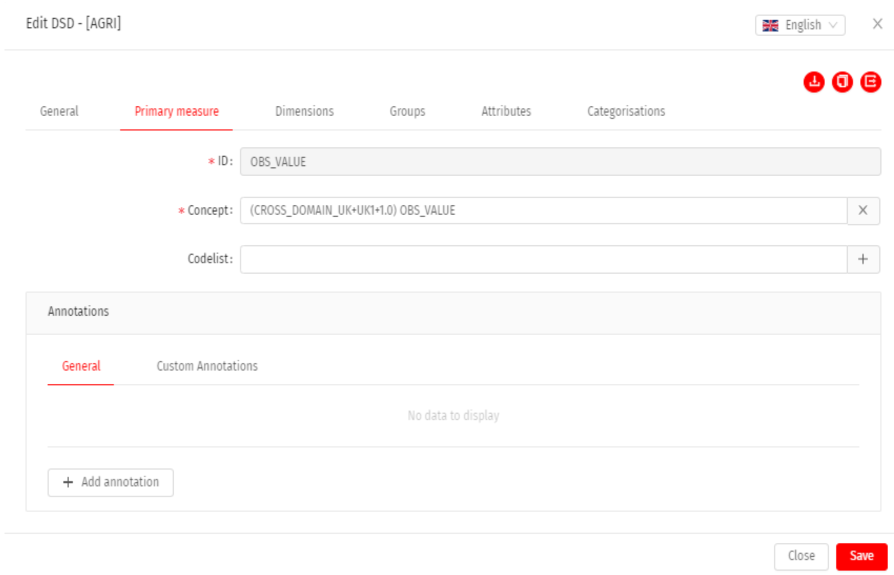
Selecting Data Structures Definitions from the "Meta Manager" module the application will show the list of DSDs present in the system similarly to the lists of all artefacts present, as explained in *artefact list*.

The View/Edit function allows the user to view the detail of the DSD and update it if the user has write privileges.



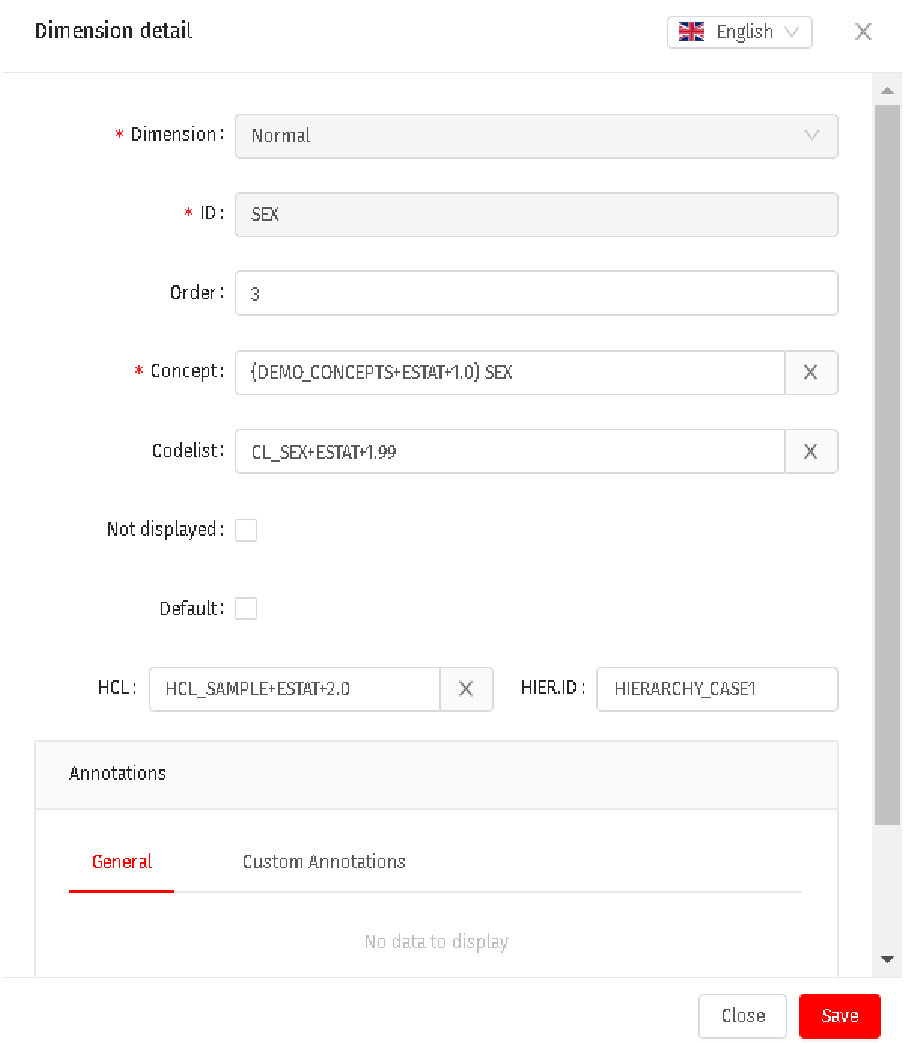
The main section of the DSD detail mask contains the following tabs :

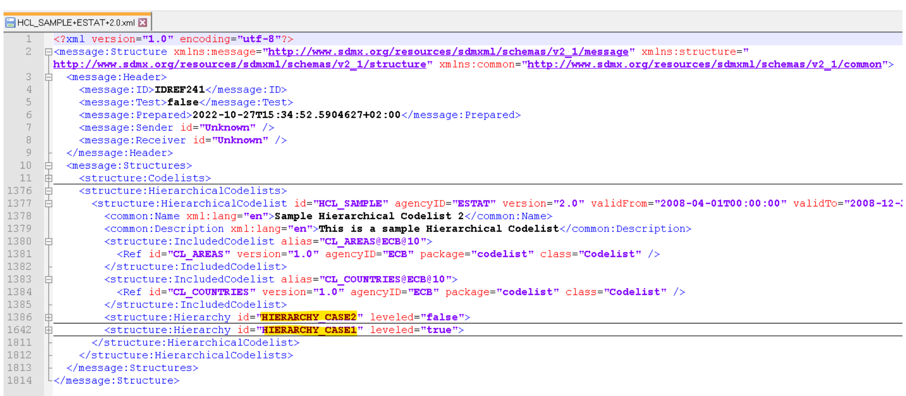
- *General*
  In the "General" window user can configure annotations, among them it is possible to edit annotations for the "Table layout", the "Chart layout" and the "Map layout". Other annotations at dimension level (e.g. not displayed) can be edited directly in the dimension definition window. For the meaning of these annotations please refer to *Other node configurations*.
- **Primary measure**
  The fields for both editing and insertion are:
  – *ID* (mandatory alphanumeric): OBS_VALUE (not modifiable)
  – *Concept* (mandatory alphanumeric)
  The System allows the selection of a Concept Scheme by means of an interface that allows selection among the concepts present.
  – *Code List* (alphanumeric not mandatory).
  The System allows the selection of a List of Codes by means of an interface that allows selection from the list of codes present.
  – *Annotations*



Note:
the definition of a unique measure complies with the SDMX standard, if it is necessary to display indicators, the measure will remain unique, what needs to be done is to define a dimension that expresses the name of the indicators.
Filtering on this dimension, the value of the chosen indicator will be obtained in obs_value.

- **Dimensions**
  The fields in insertion and modification are:
  – *Type Size*: The System allows the choice between the following Size types: Normal, Frequency, Time, Measure
  – *ID* : (alphanumeric mandatory)
  – *Order*: (numeric optional)
  – *Concept*:(alphanumeric mandatory)
  The System allows the selection of a Concept Scheme by means of an interface that allows selection among the concepts present.
  – *Code List* (alphanumeric not mandatory).
  The System allows the selection of a List of Codes by means of an interface that allows selection from the list of codes present.
  – *HCL* (alphanumeric not mandatory).
  The System allows the selection of a List of Hierarchical Codelist by means of an interface that allows selection from the list of HCL present.
  After the user has chosen the HCL list, a text field will appear in which the user must specify the desired hierarchy from those available in the chosen HCL.





These fields will be valorizable only after the 'Codelist' field is filled and will be automatically emptied if the codelist reference is removed.

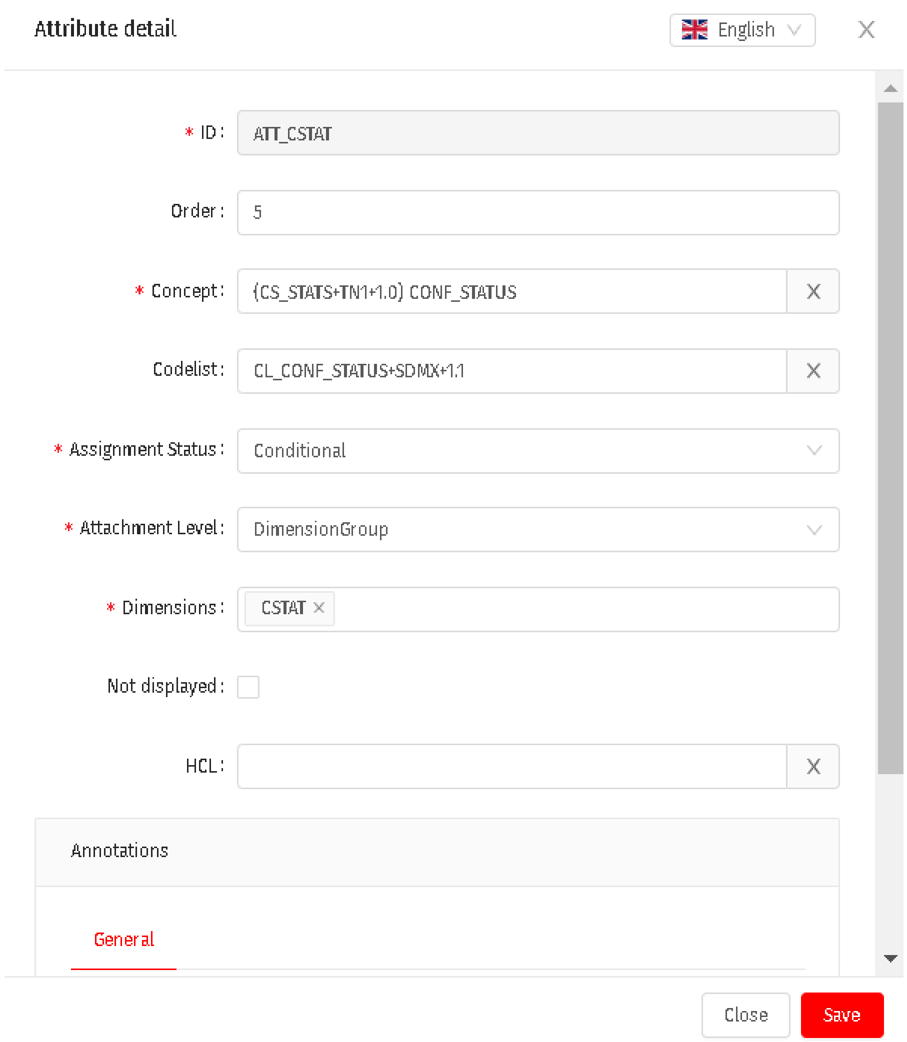
– *Annotations*

Note:
What are "Normal" dimensions? They are all non-special classifications such as: gender, age, economic sector etc…
A "Measure" dimension is a special dimension that contains the names of several in-

dicators so it will certainly be necessary to filter this dimension to show the values correctly.
A "Frequency" dimension is a special dimension that in some way is linked, for example, to update time of data and therefore contains items such as: annual, semi-annual etc..
A "Time" dimension is a special dimension that expresses the historical series so if the frequency is, for example, annual it will contain years.

- **Groups**
  The fields in insertion and modification are:
  – *Group ID*: (alphanumeric mandatory)
  – *Dimensions*: The list of dimensions defining the group is built on the basis of the user's choice of one or more DSD Dimensions.
  At least one dimension must be selected for the definition of a group.
  – *Annotations*: (alphanumeric not mandatory)
  Note:
  Groups are nothing more than groupings of dimensions to which an ID is associated in order to refer to those dimensions in a simpler way.

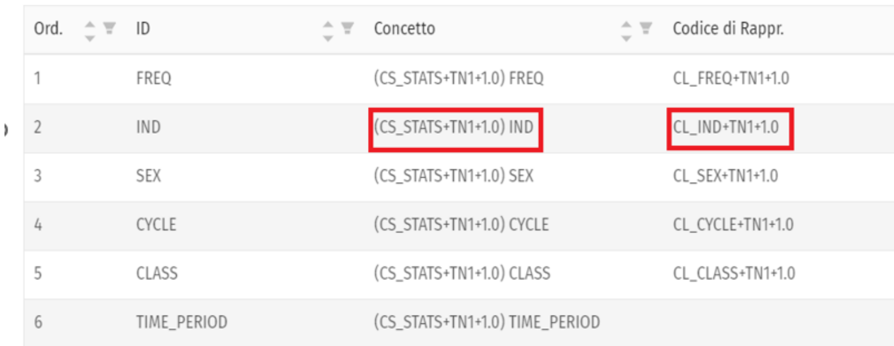
- **Attributes**
  The fields in insertion and modification are:
  – *ID*: (mandatory alphanumeric)
  – *Order*: (numeric optional)
  – *Concept*: (alphanumeric mandatory)
  The System allows the selection of a Concept Scheme by means of an interface that allows selection among the concepts present.
  – *Code List*: (alphanumeric not mandatory)
  The System allows the selection of a List of Codes by means of an interface that allows selection from the list of codes present.
  – *Assignment Status*: (alphanumeric mandatory)
  Possible values are: Mandatory, Optional
  – *Attachement Level*: (alphanumeric mandatory)
  Possible values are: Observation, Dataset, Size Group, Group
  – *HCL* (alphanumeric not mandatory).
  The System allows the selection of a List of Hierarchical Codelist by means of an interface that allows selection from the list of HCL present.
  As already seen when describing the dimensions, the user after choosing the HCL list, he must specify the desired hierarchy from those available in the chosen HCL.
  This field will be valorizable only after the 'Codelist' field is filled and will be automatically emptied if the codelist reference is removed.
  – *Annotations*: (alphanumeric not mandatory)



Note:
Attributes are additional information that, although not modelled as dimensions, are equally fundamental to understand and use data that will later be associated with the DSD.
It is easy to understand, for example, that if the measure expresses payments, it is necessary to know: the currency, the number of decimal places or whether it is expressed in thousands or millions etc..
For all this information it is necessary to define an attribute and where it must be applied, if to the whole dataset, if to groups of dimensions or to single observations.

- **Categorisations**
  Not yet implemented.

General Note:
The user can edit codelists and conceptscheme used in a dsd in the dsd management window, without having to close the window and access the respective features from the left menu.



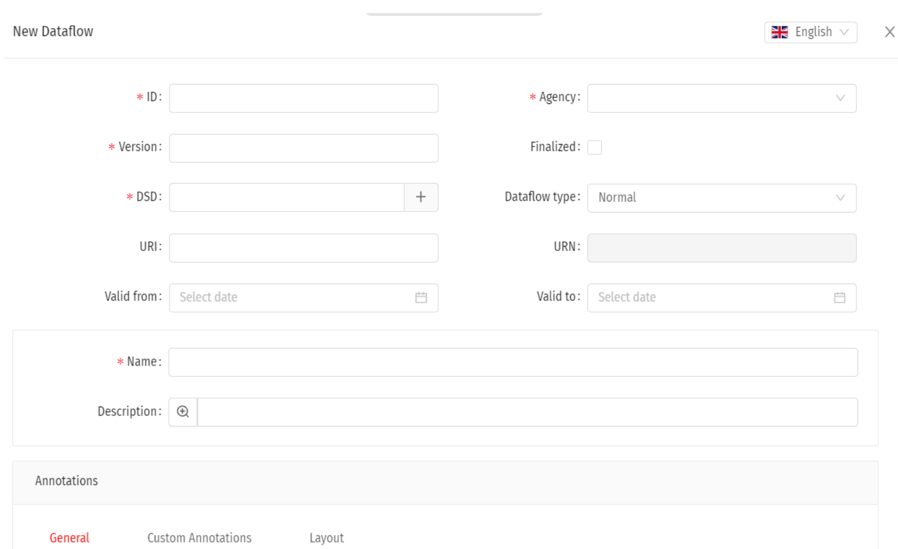
Ord.	ID	Concetto	Codice di Rappr.
1	FREQ	(CS_STATS+TN1+1.0) FREQ	CL_FREQ+TN1+1.0
2	IND	(CS_STATS+TN1+1.0) IND	CL_IND+TN1+1.0
3	SEX	(CS_STATS+TN1+1.0) SEX	CL_SEX+TN1+1.0
4	CYCLE	(CS_STATS+TN1+1.0) CYCLE	CL_CYCLE+TN1+1.0
5	CLASS	(CS_STATS+TN1+1.0) CLASS	CL_CLASS+TN1+1.0
6	TIME_PERIOD	(CS_STATS+TN1+1.0) TIME_PERIOD	

The artifacts mentioned above can be edited from the PrimaryMeasure/Dimensions/Attributes windows where the codelist/conceptscheme used by each component is indicated (in which case the artifacts will surely already be finalized).
When creating a component in an unfinalized dsd and selecting artifacts from the list of all available ones, it is also possible to finalize "unfinalized" codelists or conceptscheme so that these artifacts can be used in the dsd.

7.2.4 Dataflows

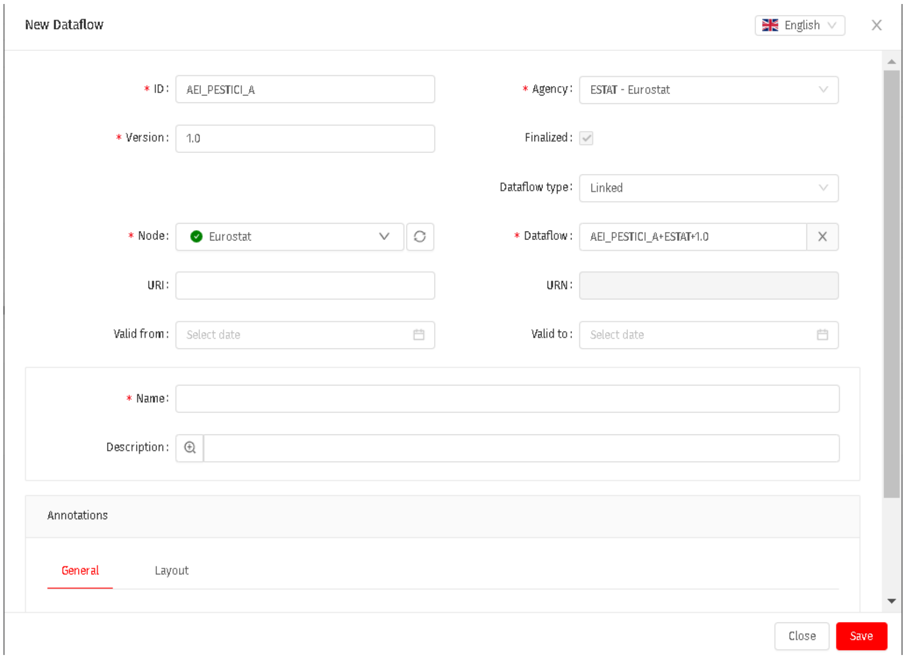
Selecting **Dataflows** from the "Meta Manager" form the application will show the list of the Node's Dataflows similar to the lists of all artefacts present, as explained in *list of artefacts*.

- **Creation of a Dataflow**
  From MetaManager it is possible to create 3 types of Dataflow:
- Normal: Standard Dataflow



- Linked: Linked Dataflow, i.e. Dataflow that are present in other nodes on which only read access is granted and that are "referenced" by the work node.
  It is possible to create multiple linked dataflows from the same physical dataflow present on a remote node.
  If the user enters ID, Agency and Version of a dataflow in the "General" window before choosing the "Linked" type, these fields cannot be changed.

On the contrary if the user first chooses the "Linked" type then the ID+AgencyID+Version fields will be suggested but the user will still be able to change them later.
A linked Dataflow is not physically present in the work node, moreover there is a specific annotation for them.

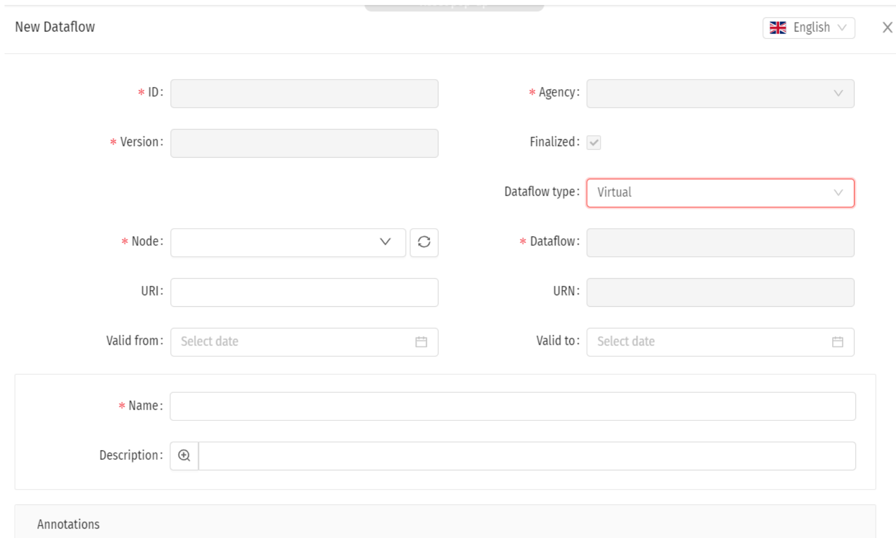


The information about the "source" dataflow on the remote node will be found in the Text field of the LinkedDataflowNode annotation of the English language in this way:

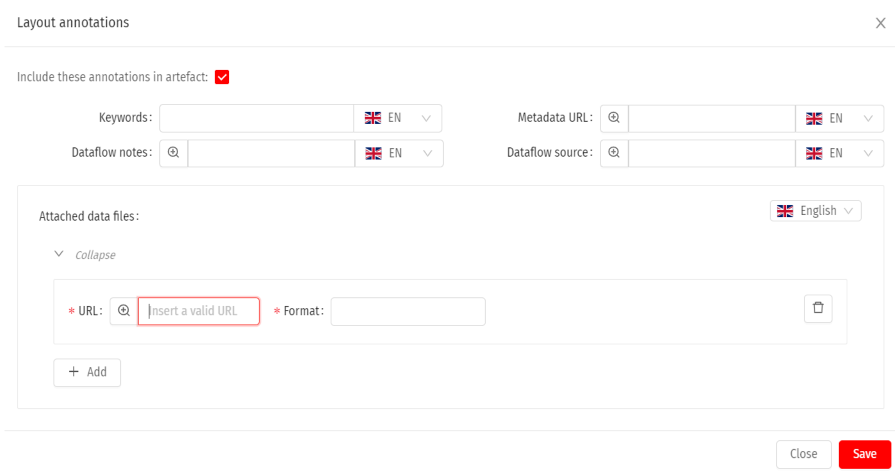
```
<common:Annotation id="LINKEDDATAFLOWNODE">
<common:AnnotationTitle>IT1</common:AnnotationTitle>
<common:AnnotationType>LINKEDDATAFLOWNODE</common:AnnotationType>
<common:AnnotationText xml:lang="en">AEI_PESTICI_A+ESTAT+1.0</common:AnnotationText>
</common:Annotation>
```

In addition, for Annotations of type Layout, it will be possible to define all types of annotations currently provided for the Normal dataflow type, which can be initialized based on the sdmx artifacts used by the pointed dataflow and present in the remote node.

- Virtual: Dataflow for which there are only one (or more) files that can be directly downloaded

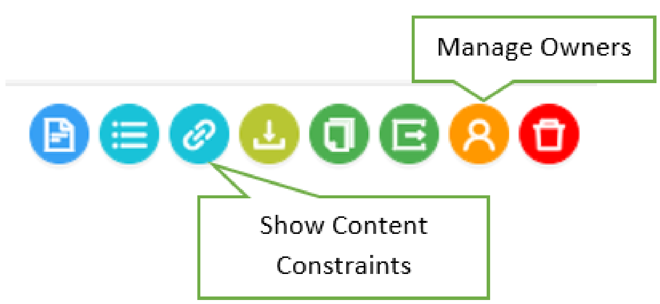


and can be inserted from the Layout section:



In the Dataflow creation window it is possible to edit the annotations; there is a section to insert the *Layout Annotations* among which there are annotations to not display dimensions (or dimension items) or annotations that set the graphic structure of the table, the chart or the map.
Other annotations can also be set (such as decimal separator, value for empty table cells, etc.).
For their meaning please refer to the paragraph *Other node configurations*.

Compared to the other artefacts on each Dataflows element, there are extra functions:

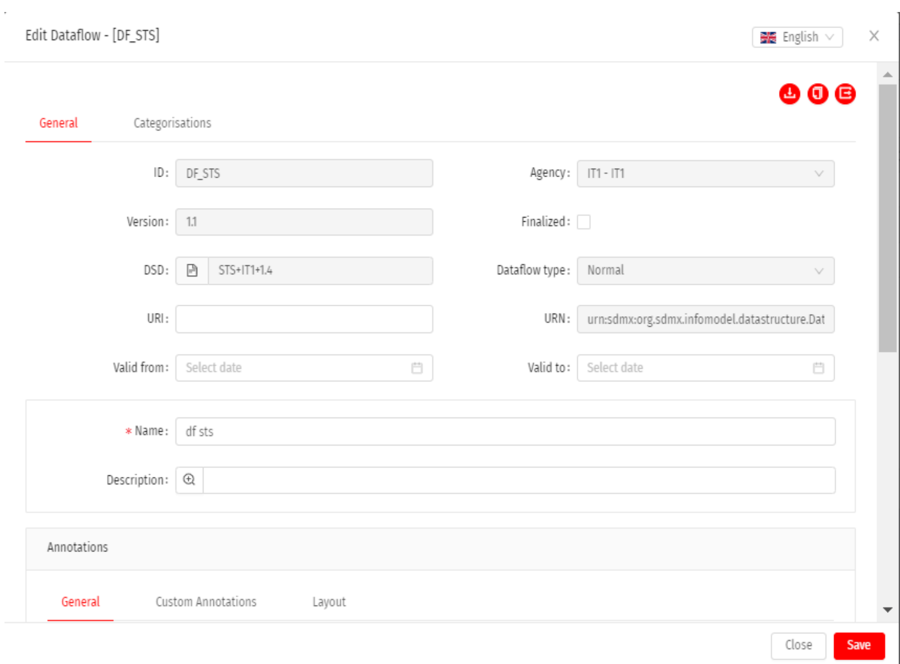


- **Show Content Constraints**
  By selecting a Dataflow from the list with the mouse and pressing the *Show Content Constraints* button, the System shows the list of Content Constraints related to the selected Dataflow.

- **Manage owners**
  A user owning a Dataflow can assign the rights on this Dataflow to another user using the *Manage owners* button.
  The user owner of the Dataflow chooses from the list of users present in the node

and clicking on the Save button gives the selected user rights on the Dataflow.

By selecting an item from the list with the mouse and pressing the View/Edit button, the system displays the Detail mask.



The main section of the Dataflows detail mask contains the following tabs:

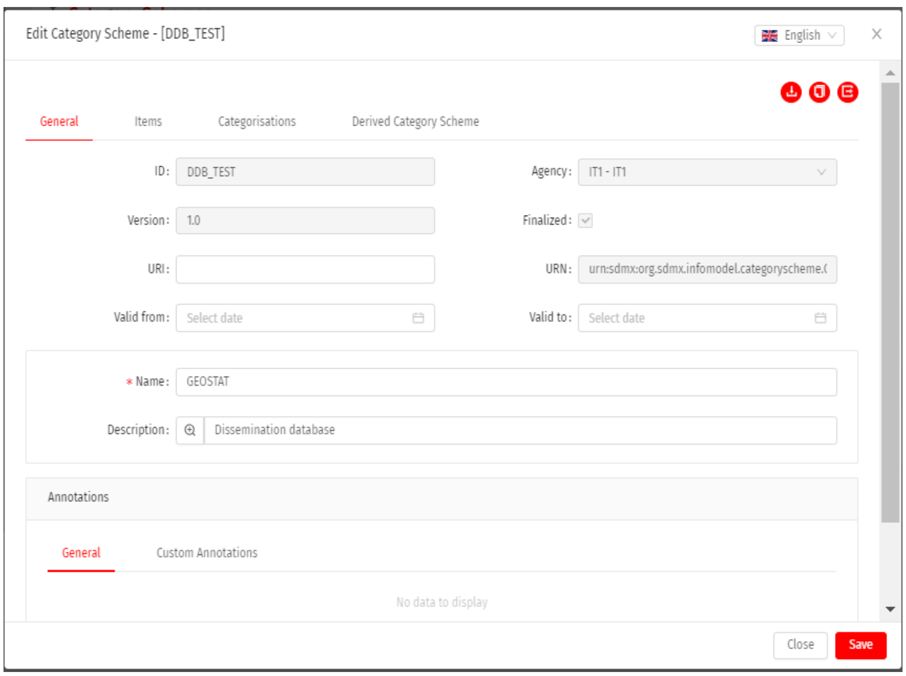
- *General*

NOTE:
in addition to the entries present in common with all the other artefacts, in the General section, we also find the DSD field in which the reference to an existing DSD in the System must be inserted.
Many Dataflows can be created referring to the same DSD, for example reflecting different time intervals or geographical areas.

- **Categorisations**
  Not yet implemented.

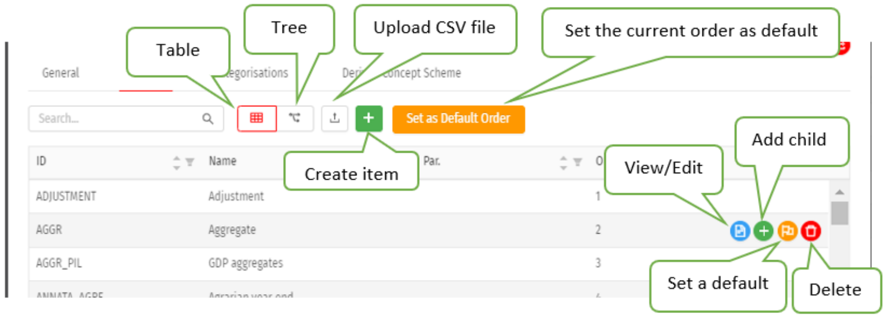
7.2.5 Category Schemes

Selecting **Category Schemes** from the "Meta Manager" module the application displays the list of the Node's Category Schemes similar to the lists of all artefacts present, as explained in *list of artefacts*.
By selecting an item from the list with the mouse and pressing the View/Edit button, the system displays the Detail mask.



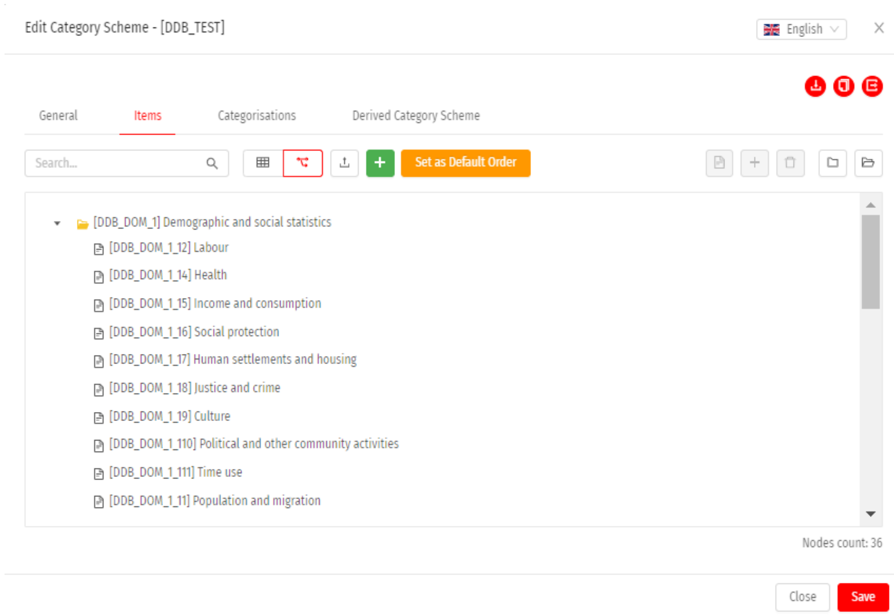
The main section of the Category Scheme detail mask contains the following tabs:

- *General*
- **Items**
  Items can be displayed in grid or tree mode. By default the grid view is proposed.

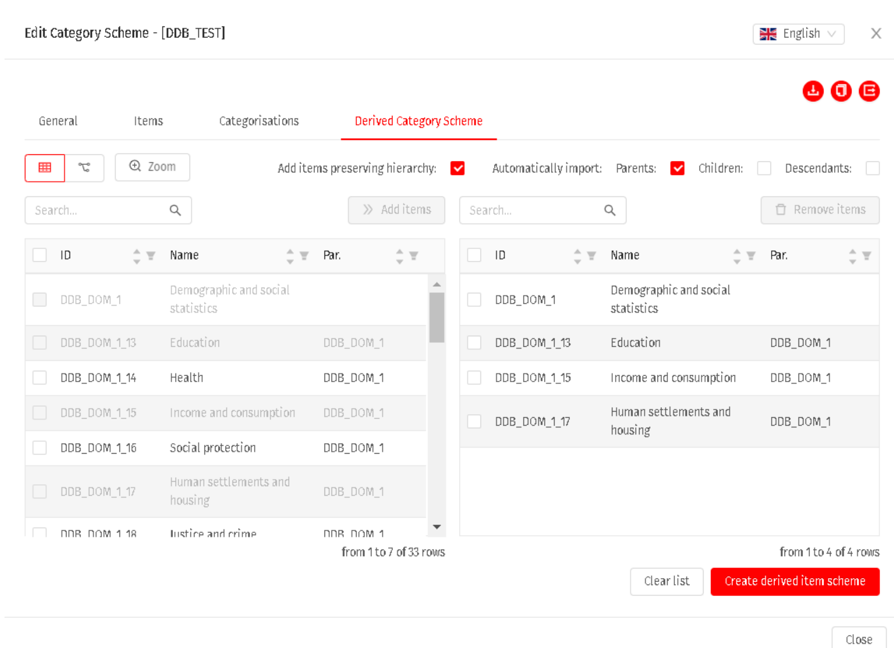


As an alternative to the insertion of the single items, it is possible to *import a CSV file* in order to insert them in bulk, otherwise the management functions of the single items are the following:

- **Create Item**
  To create a new Item, the user must use the *Create Item* button at the top.
  The System presents the data acquisition mask.
  The fields to be populated are the following:
  – **Identifier Item** (mandatory - alphanumeric)
  – **Language** (mandatory - alphanumeric)
  – **Item name** (required - alphanumeric)
  – **Description** (optional - alphanumeric)
  – **Fullname** (optional - alphanumeric)
  – **Order** (optional - numeric)
  – **Parent** (optional - from a list of items already present)
  – *Annotations* (optional - alphanumeric)
  The new Item is displayed as a record of the grid table and the possible information related to the "Parent Code" is valorized in a column of the grid view.
- **Add Child**
  The *Add Child* button is activated on the "Parent" row and opens a mask with the fields indicated in **Create Item** where obviously the *Parent* field will be preset with the value of the selected parent.
  The new Item is inserted with indentation under the Parent code in the "folder" representation.
  The mask contains the buttons for system acquisition: **Save:** to save and close the mask and **Close:** to close the mask without saving.
  In the folder view, the order and the parentship of an item can be determined by drag-and-drop operations of the item itself.
  The user can switch to the folder view by clicking on the "Tree" button.

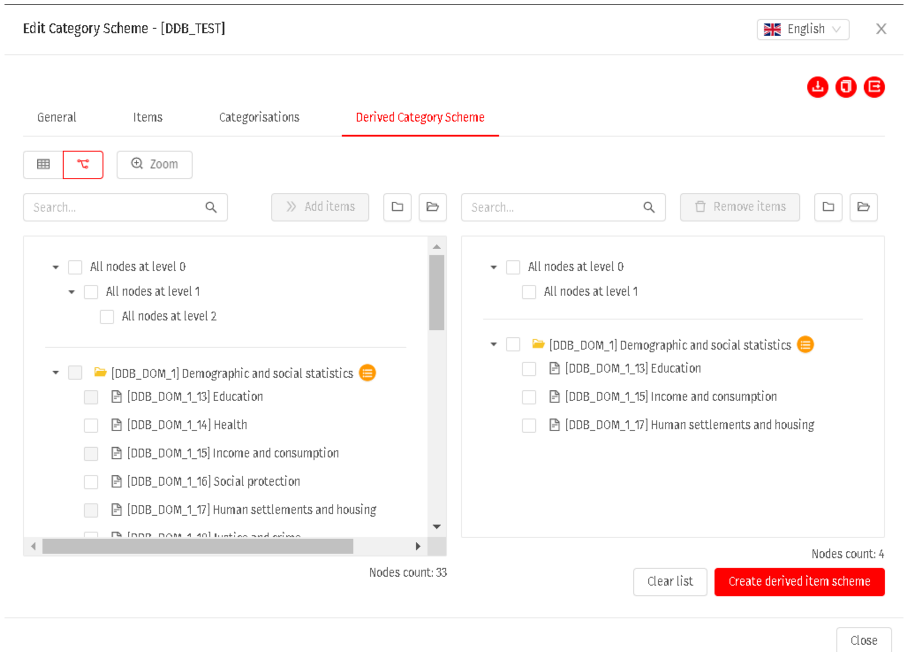


- **Categorization**
  (not yet implemented)
- **Derived Category Schema**
  This tab allows the creation of a new Category Scheme from a subset of items of the starting scheme, hence the adjective "derived".
  In choosing the elements of interest in table view mode, the user is helped by tools that allow him to take into account or not the hierarchies between parent elements, child elements, etc.



If tree mode is used it will be possible to select:
– a specific node
– a specific sub-level
– all children of a specific node

in addition, the zoom functionality will display the two artefacts in a full screen popup (left and right of the screen) allowing selection.



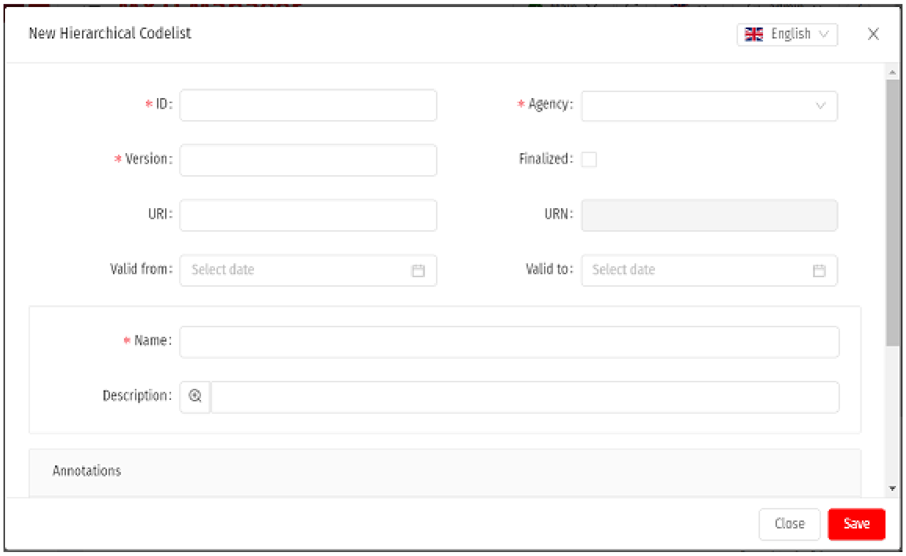
The administrator user can hide specific Category Schemes to other users from the "Set Permission" configuration. If so, the user will not be able to insert dataflows under such categories or view them in the "Category Schemes and Dataflows" in the Meta Manager section.

7.2.6 Categorisations

….. coming soon …..

7.2.7 Hierarchical Code Lists

Hierarchical code lists are used to model non-exclusive hierarchies (where a child can have more than one parent).
It is possible to import or create a new hierarchical codelist.
Once the user decides to create a new HCL, a new window with two tabs appears. Only HCL with valued based hierarchies can be created.
In the General Tab, the user speficies the mandatory infomation which identify the artifact itself (ID, Agency, Version).



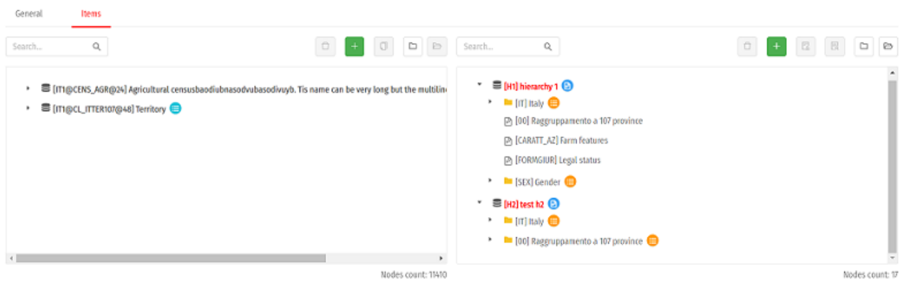
After filling-in the main information and clicking "Save", the Item Tab is showed which is organized in two panels.
In the left panel the user can:

- select the codelists referenced by the HCL from the final ones in the MSDB;
- display in tree format the items of a referenced codelist;
- remove a referenced codelist via appropriate button;
- collapse a single codelist;
- select via appropriate button all the items of a codelist and then possibly copy them to a hierarchy on the right;
- select via appropriate button all children of a given node;
- copy one or more referenced codelist items to a specific location in a hierarchy already created in the right panel by right-clicking and then clicking on the specific context button or by selecting them and clicking on the appropriate button in the top button pad.

In the right panel the user can:

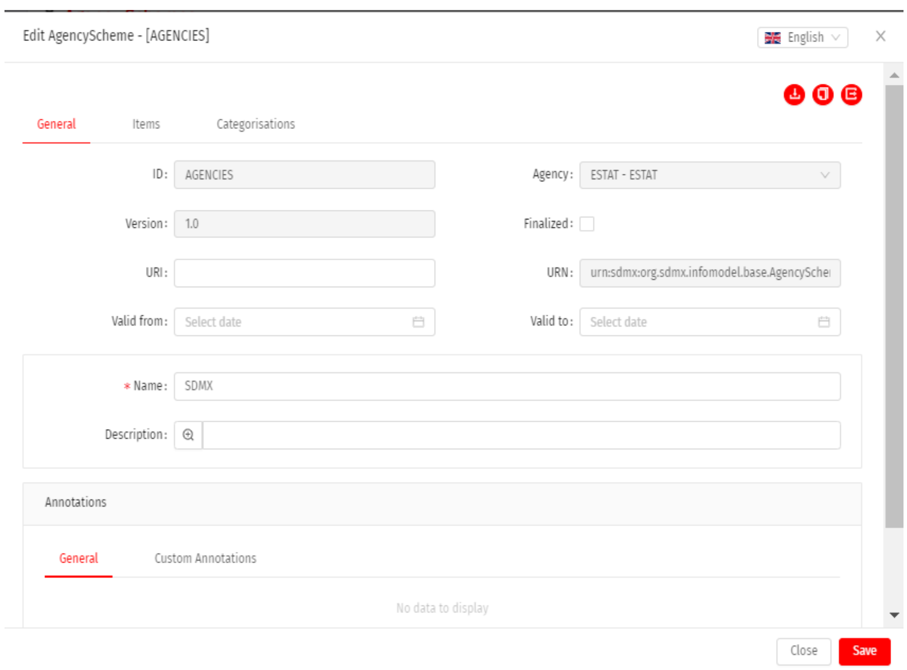
- create a new Hierarchy (value-based type) for the HCL. Each HCL can contain 0 or more Hierarchies. For each Hierarchy you will be able to define the general information (Id, Name (multilingual), Description (multilingual), Annotations (multilingual)) from a dedicated popup. The hierarchy will be created by default in the queue of all existing ones: in case one and only one hierarchy is selected instead, the new hierarchy will be created immediately after the currently selected one;
- view the general info of a specific hierarchy;
- delete a hierarchy via appropriate button;
- paste as child or sibling one or more referenced codelist items previously copied from the left panel. This action can be completed via contextual button viewable by right-click or, alternatively, via the button in the top pane;
- collapse a single hierarchy;
- select via appropriate button all the children of a given node;

- change via drag & drop the relative order of the hierarchies;
- rearrange via drag & drop the hierarchical structure of a specific hierarchy and the relative order of items at the same hierarchical level;
- select an item in a Hierarchy and delete it via appropriate button.



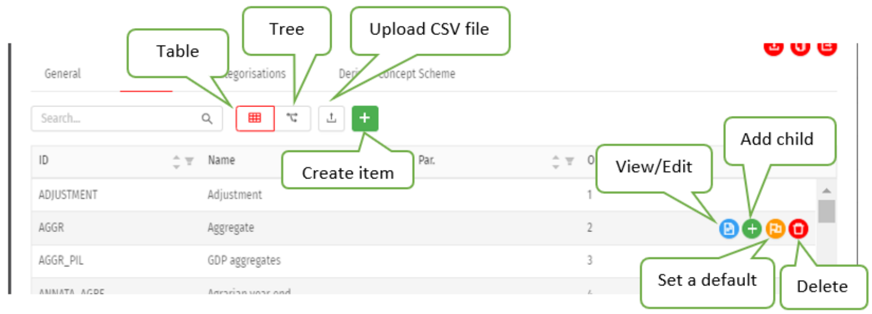
7.2.8 Agency Schemes

Selecting **Agency Schemes** from the "Meta Manager" module, the application displays the list of the Node's Agency Schemes similar to the lists of all artefacts present, as explained in *list of artefacts*.
Selecting an element of the list with the mouse and pressing the View/Edit button, the System shows the Detail mask.



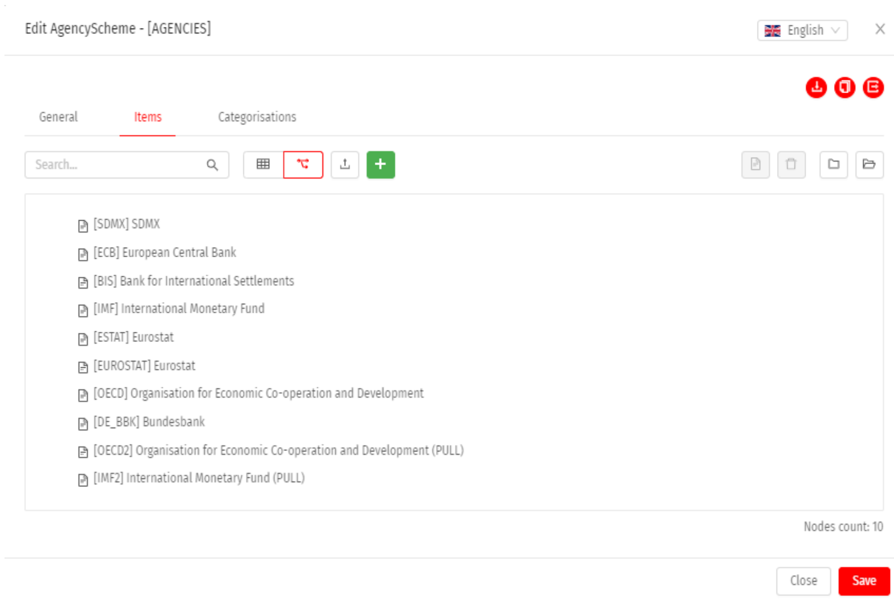
The main section of the Agencies Scheme detail mask contains the following tabs:

- *General*
- **Items**
  The Items can be displayed in grid or tree mode. By default the grid view is proposed.



As an alternative to the insertion of the single items, it is possible to *import a CSV file* in order to insert them in bulk, otherwise the management functions of the single items are the following:

- **Create Item**
  To create a new Item, the user must use the *Create Item* button at the top.
  The System presents the data acquisition mask.
  The fields to be populated are the following:
  – **Identifier Item** (mandatory - alphanumeric)
  – **Language** (mandatory - alphanumeric)
  – **Item name** (required - alphanumeric)
  – **Description** (optional - alphanumeric)
  – **Fullname** (optional - alphanumeric)
  – **Order** (optional - numeric)
  – **Parent** (optional - from a list of items already present)
  – *Annotations* (optional - alphanumeric)
  The new Item is displayed as a record of the grid table and the possible information related to the "Parent Code" is valorized in a column of the grid view.
- **Add Child**
  The *Add Child* button is activated on the "Parent" row and opens a mask with the fields indicated in **Create Item** where obviously the *Parent* field will be preset with the value of the selected parent.
  The new Item is inserted with indentation under the Parent code in the "folder" representation.
  The mask contains the buttons for system acquisition: **Save:** to save and close the mask and **Close:** to close the mask without saving.
  In the folder view, the order and the parentship of an item can be determined by drag-and-drop operations of the item itself.
  The user can switch to the folder view by clicking on the "Tree" button.



- **Categorization**
  (not yet implemented)

7.2.9 Data Provider Schemes

….. coming soon …..

7.2.10 Data Consumer Schemes

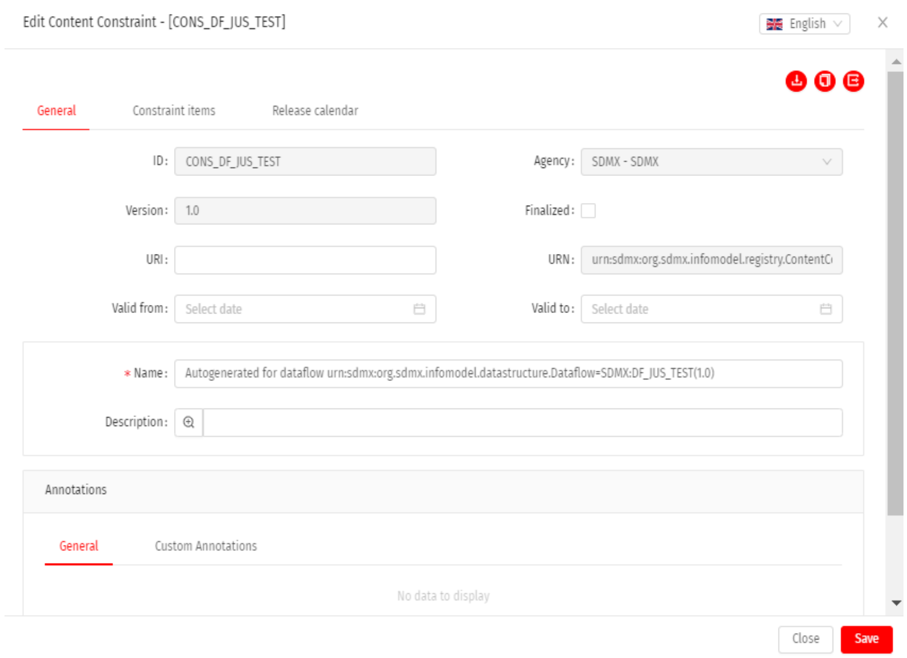
….. coming soon …..

7.2.11 Organization Unit Schemes

….. coming soon …..

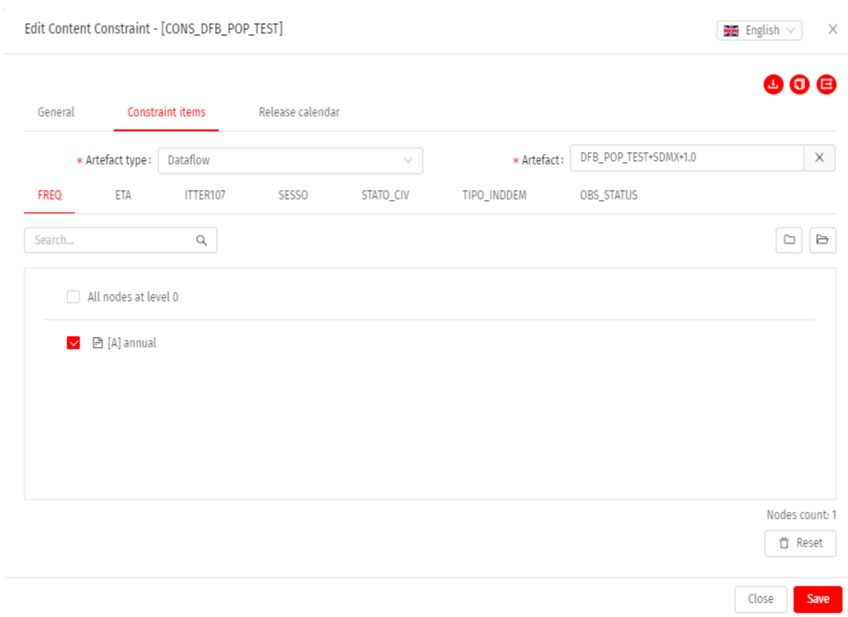
7.2.12 Content Constraints

Selecting **Content Constraints** from the "Meta Manager" module, the application displays the list of Content Constraints of the Node similarly to the lists of all artefacts present, as explained in *list of artefacts*.
Selecting an item in the list with the mouse and pressing the View/Edit button, the system displays the Detail mask.



The main section of the Content Constraints detail mask contains the following tabs:

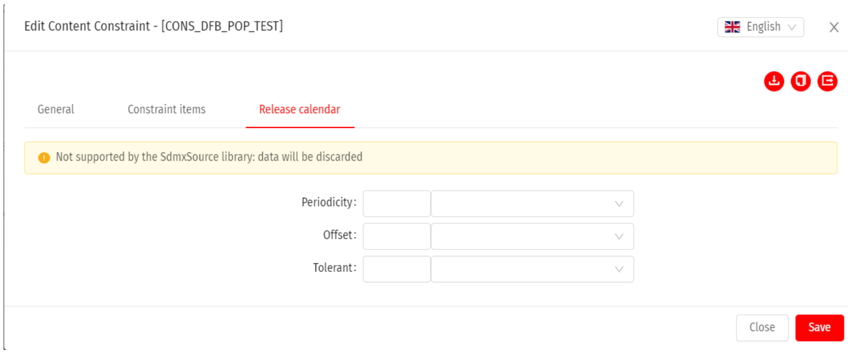
- *General*
- **Constraint Elements**
  In this tab it is possible to insert/edit the type of artefact to which the CC is associated (DSD or Dataflow) and to choose the desired artefact from a dynamic list that will show all those available for use in the System.



Once the artefact to be associated with the CC has been determined, the System will display a series of tabs below (one for each dimension or attribute present in the selected artefact).
In each tab from left to right the user will find the possible values of the dimension or attribute linked to that tab and will be able to add filters according to his needs.

Keep in mind that the choices made in one tab affect the items shown in the following tabs, so for example it may happen that, according to the choices made in the previous tabs, only one item can be chosen for the Sex dimension (e.g. Male). This implementation choice has been made to avoid the creation of inconsistent filter combinations.

- **Release Calendar**



7.2.13 Structure Sets

….. coming soon …..

7.2.14 Provision Agreements

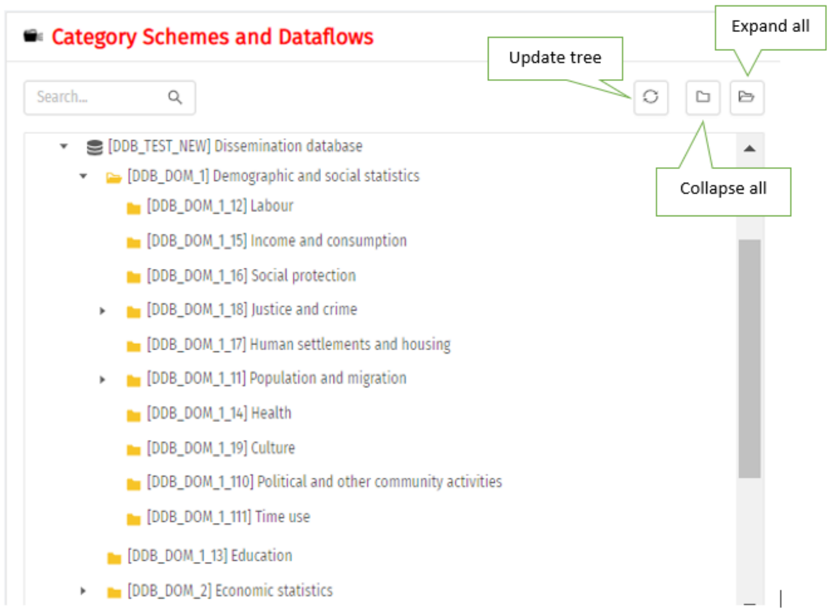
….. coming soon …..

7.2.15 Registrations

….. coming soon …..

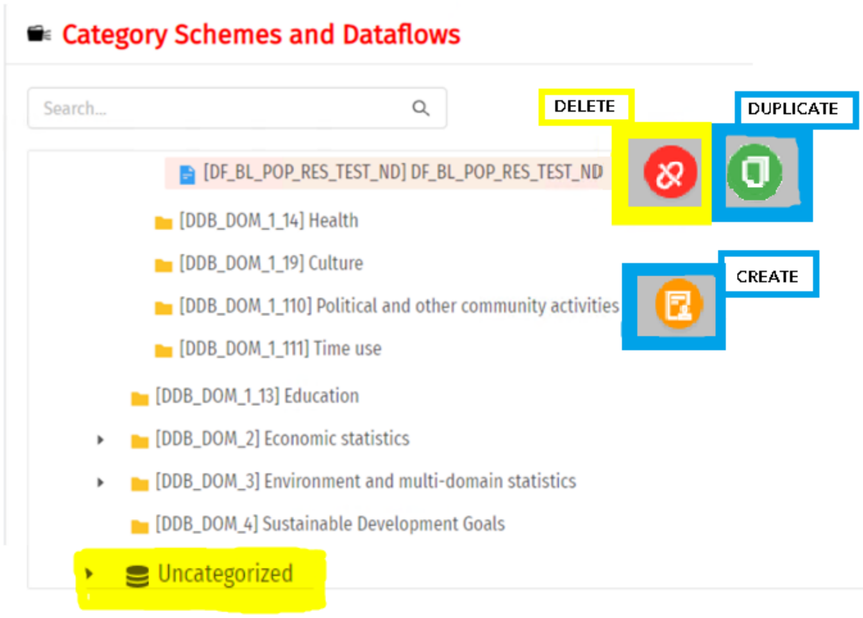
7.2.16 Category and Dataflow Schemes

The Category and Dataflow Schemes management function is present in the left side menu starting from the "Meta Manager" item.



The screen shows Categorisations and Dataflows in a tree view.
The user can expand a Category Schema by clicking on the arrow next to its name.

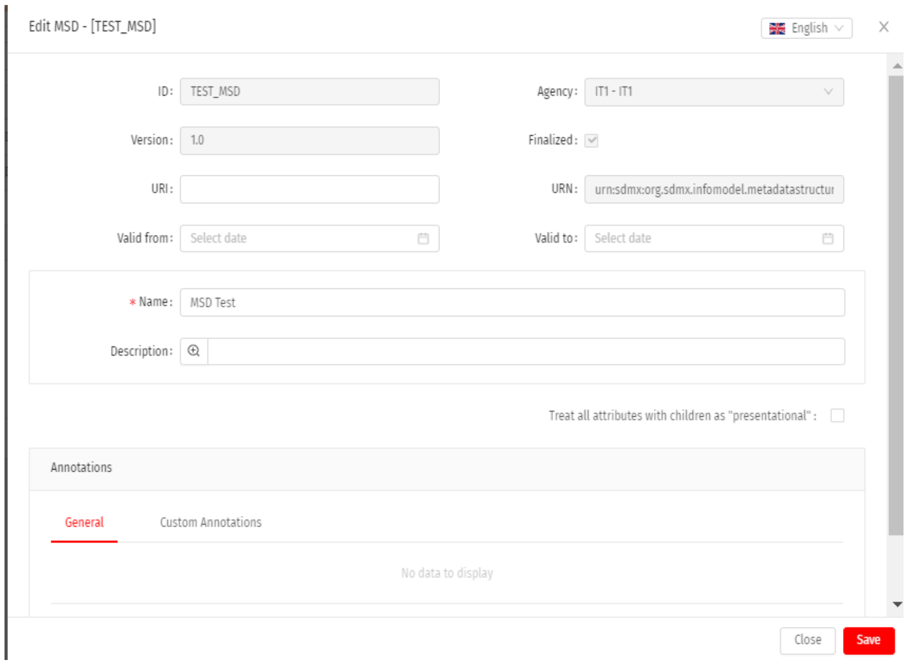
In the upper right part of the screen there is an **Update category tree** button.
This function allows the user to update the tree by reloading the list that may contain new Categories and/or Dataflow components created by other users within the same System and Node connection.
Further functions available to the user are those related to the exploded view of Categories Schemes tree or the reduced view.
By selecting a Dataflow and right clicking the mouse button it is possible to activate the **Delete Categorization** button, and the Dataflow removed from the categorization is moved to the **Uncategorized** folder.



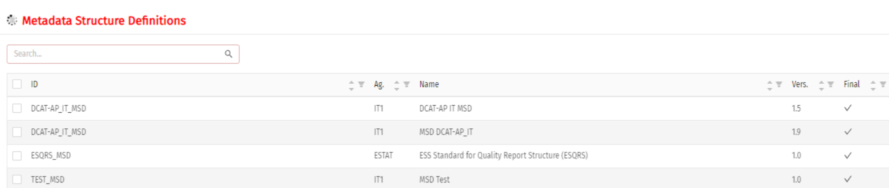
Alternatively (again by right-clicking), the user can activate the **Duplicate Categorization** button and, after right-clicking on a category, a “Create” button will appear allowing the user to create a new categorization under the selected category.
Uncategorized dataflow must first be classified into a category and, only after this initial categorization, the user can duplicate it and place it in another category.
It is possible to change Dataflow categorisation with the drag&drop tool or during the “Categorisation” phase of the Dataflow Builder.

7.2.17 Metadata Structure Definitions

By selecting **Metadata Structure Definitions** from the “Meta Manager” module the application displays the list of Metadata Structure Definitions of the Node similar to the lists of all artefacts present, as explained in *list of artefacts*.
By selecting an item in the list with the mouse and pressing the View/Edit button, the system displays the Detail mask.



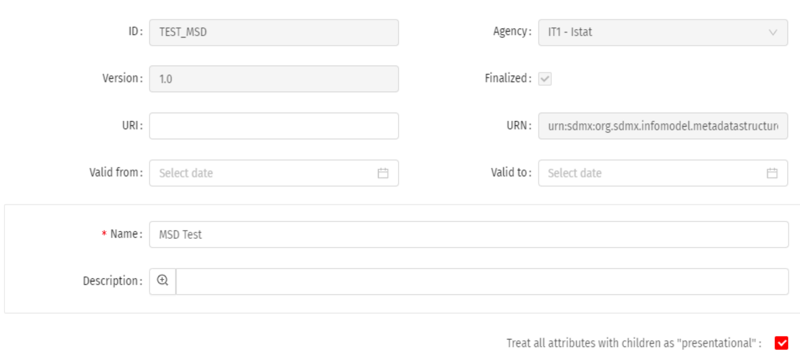
This is a particular SDMX artefact and represents a template for reporting and dissemination of reference metadata.
The Suite does not provide the creation of new artefacts of this type, so user can only *import an external MSD* through the import button in the top right corner of the MSD List.



7.2.17.1 Attribute "Presentational"

Title section attributes - Presentational

Within the Metadata Structure Definition, the user can configure the attributes that show the title of a section in the report, by checking the ***presentational*** button, as shown below:

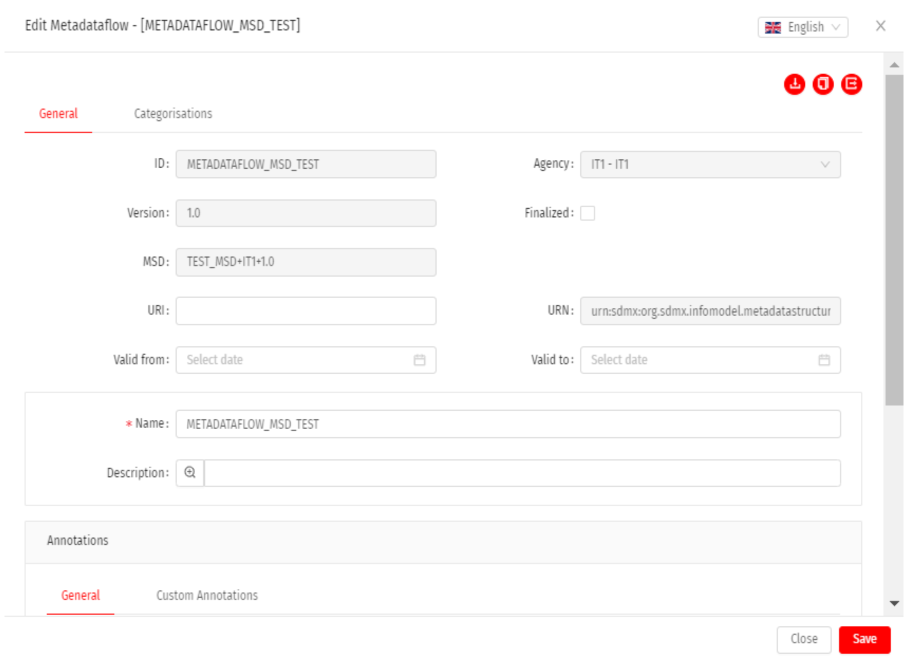


Or by adding the CUSTOM_IS_PRESENTATIONAL annotation in the SDMX file as follows:

```
<str:MetadataAttribute id="DCAT_AP_IT_CATALOGUE_TITLE" urn="urn:sdmx:org.sdmx.in
    <com:Annotations>
        <com:Annotation>
            <com:AnnotationTitle>True</com:AnnotationTitle>
            <com:AnnotationType>SDMX21_IsMultiLingual</com:AnnotationType>
            <com:AnnotationURL />
        </com:Annotation>
        <com:Annotation id="CUSTOM_IS_PRESENTATIONAL">
            <com:AnnotationTitle>CUSTOM_IS_PRESENTATIONAL</com:AnnotationTitle>
            <com:AnnotationType>CUSTOM_IS_PRESENTATIONAL</com:AnnotationType>
        </com:Annotation>
    </com:Annotations>
```

7.2.18 Metadataflows

Selecting **Metadataflows** from the "Meta Manager" form will show the list of Metadataflows of the Node similarly to the lists of all artefacts present, as explained in *list of artefacts*.
By selecting an item from the list with the mouse and pressing the View/Edit button, the system displays the Detail mask.



The main section of the Metadataflows detail mask contains the following tabs:

- *General*

NOTE:
in addition to the entries common to all other artefacts, in the General section, we also find the MSD field in which the reference to an existing MSD in the System must be entered.
Several Metadataflows can be created referring to the same MSD.

- **Categorizations**
  Not yet implemented.

CHAPTER

EIGHT

DATA MANAGER

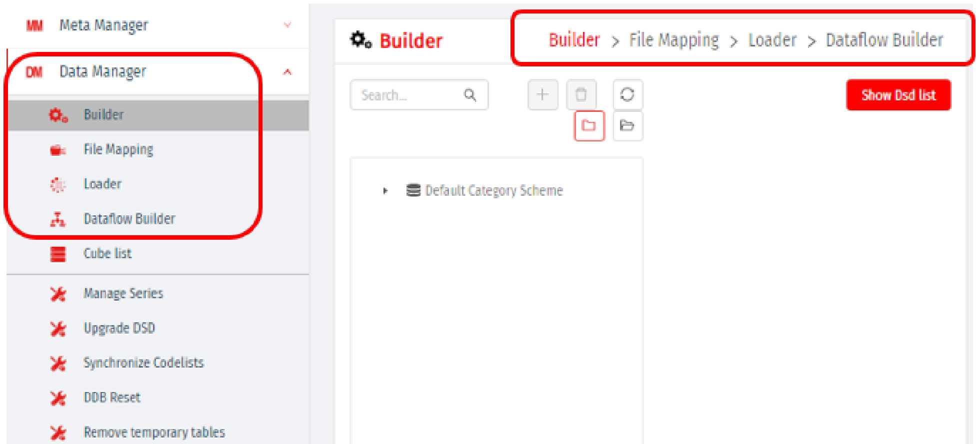
This section of the guide explains the steps needed to create a Dataflow and all the functions available to manage it.

8.1 How to load a new Dataflow

The loading of a new Dataflow is done in 4 sequential steps:

- **Builder**: creation of a Cube associated with a DSD
- **File Mapping**: creation of a mapping between Cube and data
- **Loader**: loading data
- **Dataflow Builder**: Dataflow creation and publication.

can be activated from the menu on the left or from the list in the upper right of the masks used.



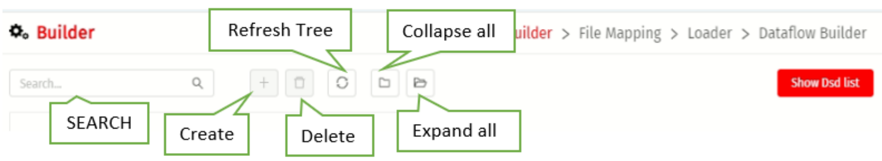
- *Builder*
- *File Mapping*
- *Loader*
- *Dataflow Builder*

8.1.1 Builder

The Builder window displays the Category Work Scheme with a tree structure.

The leaves of the tree are the registered cubes. To manage the list of cubes the following functions are provided:

- **Create Cube \ Create Category**
- **Delete Cube \Delete Category**.
- **Update Tree**
- **Collect all**
- **Expand all**
- **Search**

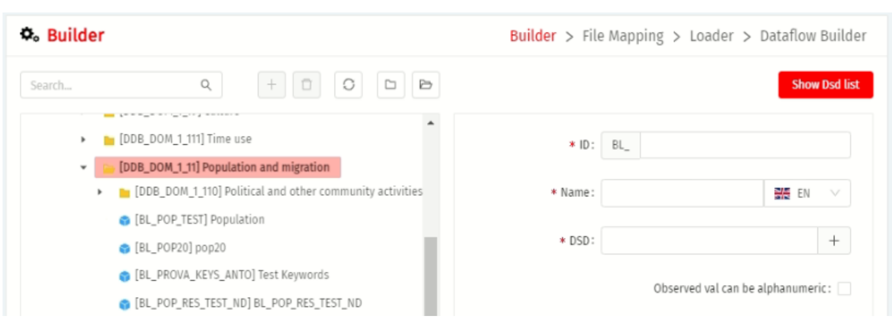


Some functions are active or not depending on the selected element (e.g. the "Create Cube" function is active if an element of the Category Diagram is selected, "Delete Cube" is active if a Cube is selected).

Create Cube

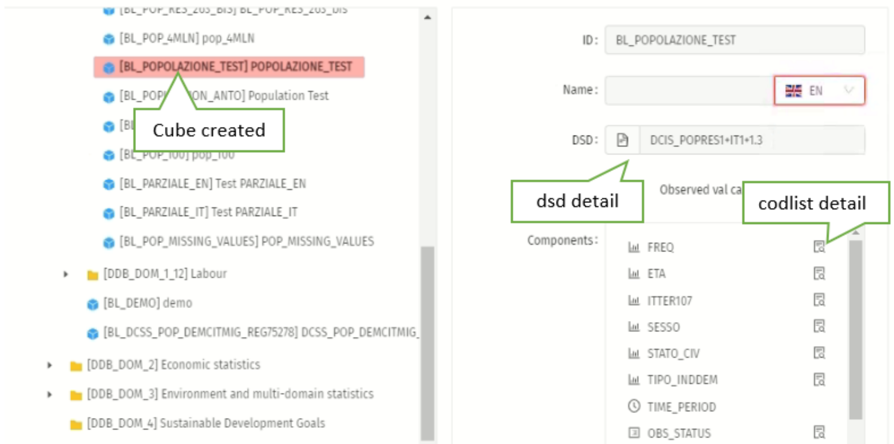
By clicking on the icon, a window appears allowing the user to choose whether to create a new cube or to create a new sub-category with reference to the one that has been selected.
Clicking on create cube opens the mask for entering the necessary fields:

- *Cube ID* (alphabetical type, mandatory)
- *Name of the Cube* (alphabetic type, mandatory)
- *Language*
- *DSD identifier* (set by selection function, mandatory)
- *Flag "Observed value may be alphanumeric "* (optional)

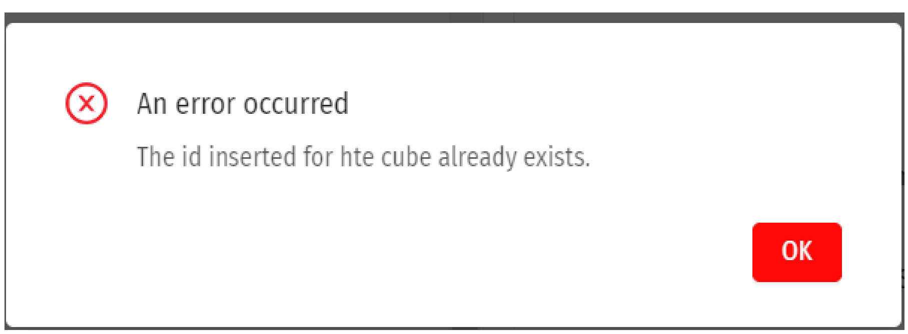


The non-editable prefix of the ID field is set during configuration.
In the "Name" field the user enters the name of the Cube for each language available in the Node.
Selecting the "+" next to the DSD field opens a pop-up window that allows the user to view the DSDs present in the System and to choose the one to be used to create the

Cube.
The "List DSD" button, in the upper right corner, shows the same list of DSD present in the System.
Once selected the DSD the mask shows the list of Dimensions and Attributes: by default will be selected and not deselectable the Dimensions and Attributes that in the DSD are defined as mandatory, while the optional attributes by default will not be selected, but can be chosen at this stage.



Among the selectable attributes there is the TID (table identifier) that is used to uniquely identify the Dataflows built from the same Cube (displayed in the image). from the same Cube (displayed in the tree on the left).
In this way it is also possible, for example, to load two attributes at dataset level for two dataflows built from the same cube: to do this it is necessary that the user adds the TID in the cube and when loading a data file (see paragraph *Loader*) or an attribute file, he loads different attribute values for different TID values.
The user has the possibility to check the structure of the DSD and its components by clicking on the icons present next to the artefacts.
During the creation of the Cube the System checks the correctness of the data entered. If the user, for instance, tries to create a Cube with an already existing ID, saving is blocked and the user is notified with an error message.

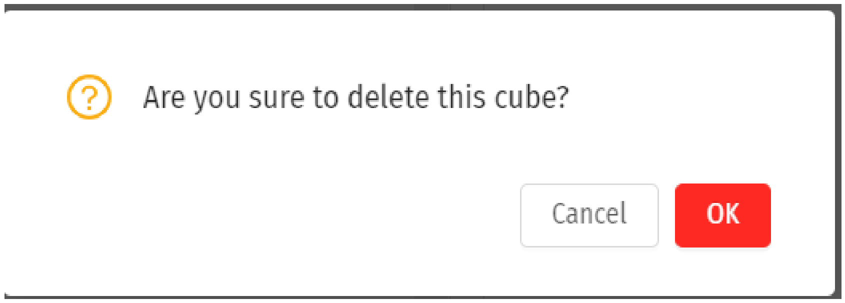


Once the mandatory fields have been filled in, the "Save" button becomes active and the cube can be saved.
The window with all the Cube's properties is displayed whenever the Cube is selected.

Delete Cube

To delete a Cube, the user must select it with the mouse; in this way the "Delete Cube" button is activated.

By clicking on the button, a mask appears asking the user to confirm the operation. If yes, the deletion is carried out.



Following deletion, the System automatically updates the tree of the Category Schemes.
If a Dataflow is associated with the Cube, deletion is not allowed.

Create Category

Clicking on the "+" icon the user can create a new Category, a window is opened where the user has to type the Category ID, the Name and, optionally, the parent of the Category among those already defined. Categories can be moved under other categories with drag&drop functionality.

Delete Category

To delete a Category, the user must select it with the mouse; the "Delete Category" button will only be active if the Category does not have any associated cubes. When the button is clicked, a mask appears asking the user to confirm the operation. If so, the deletion is carried out.

Update tree

This function allows the user to update the Category Diagrams tree to show changes following the creation and deletion of cubes and Category Diagrams.

Collect All

This function displays the Category Tree in a reduced form.

Expand all

This function displays the Category Tree in an exploded view.

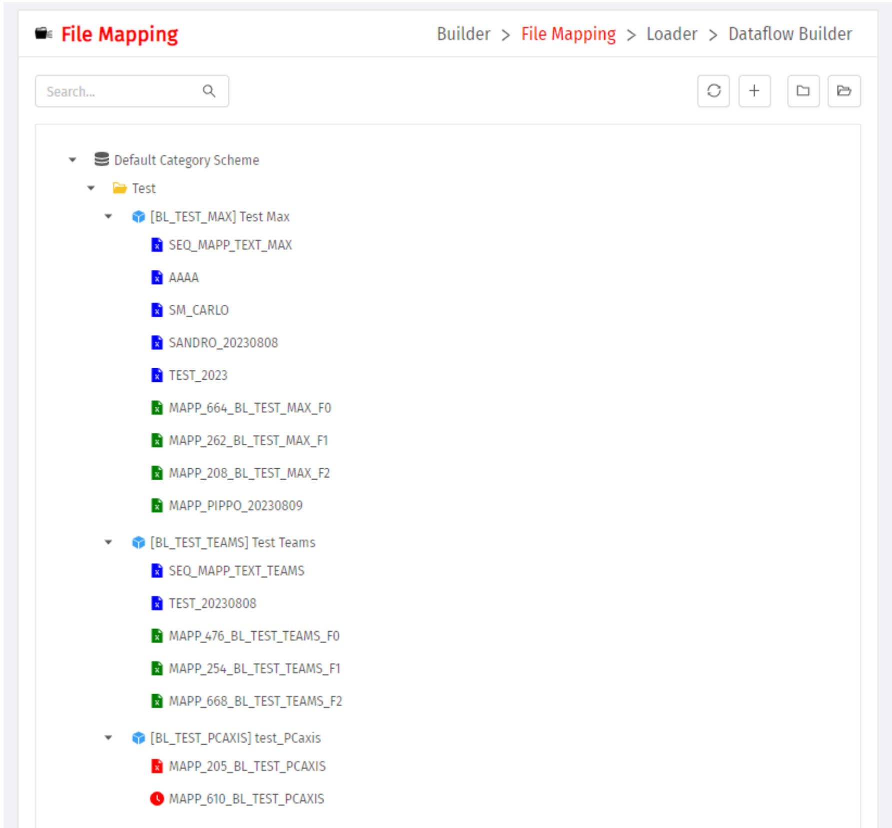
Search

The search is carried out by means of an editable field in the upper left-hand corner of the screen, as described in *Search in the list of artefacts*.

8.1.2 File Mapping

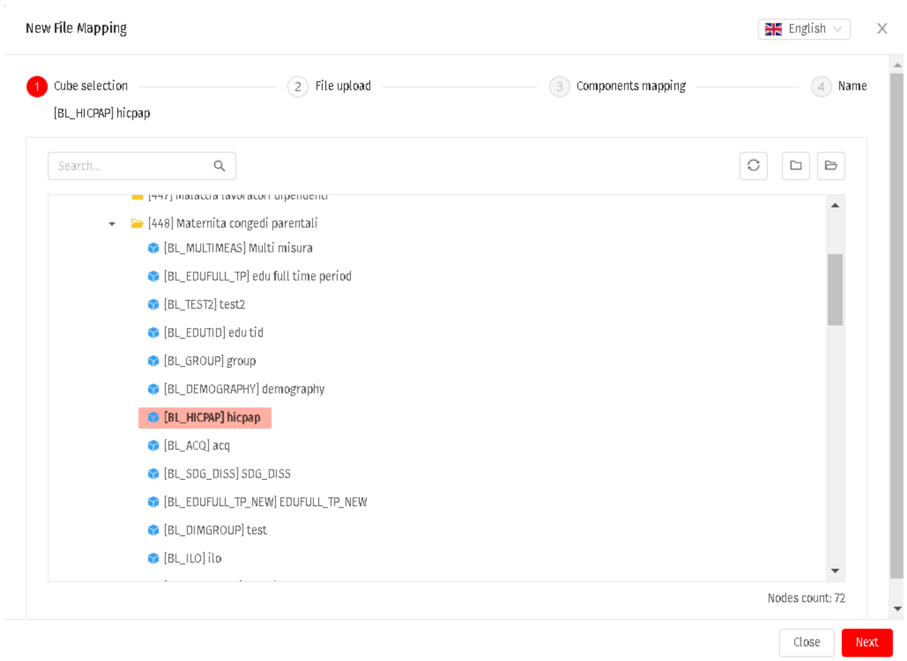
The File Mapping window displays all mappings in tree format. It is possible to:

- **Create a new mapping**
- **View or delete an existing mapping**.



Create a new mapping

The selection of the "New" button, in the upper right corner, opens a pop-up window for the setting of the information necessary for the creation of the Mapping.
The system provides for successive steps displayed in the top bar: it is possible to move to the next step with the "Next" button or selecting the step directly from the bar.
The user can go back to the previous steps to make any changes or to check the settings.



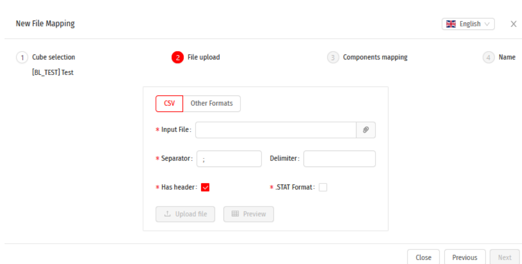
Cube Selection

The first step is the identification of the Cube to be mapped among those present in the tree of Category Schemes.

Once the Cube has been selected the "Next" button becomes active to pass to the next step.

Upload File (CSV)

The system displays a form for uploading the CSV file with which to perform the mapping with the selected Cube.



The form allows searching the CSV file from filesystem; once the CSV file has been selected, its name will be shown in the "File" field.

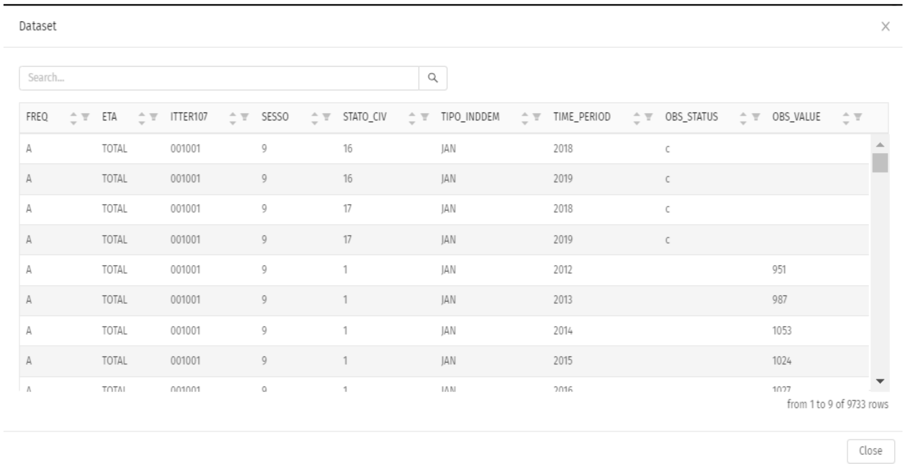
The user must check that the "Separator" and "Header" settings, which are set by default, correspond to what is present in the chosen CSV file and change them if necessary.

The form allows the user to set a "Delimiter" (optional field) and the ".STAT Format" flag (activate the flag if the CSV file has the .STAT format).

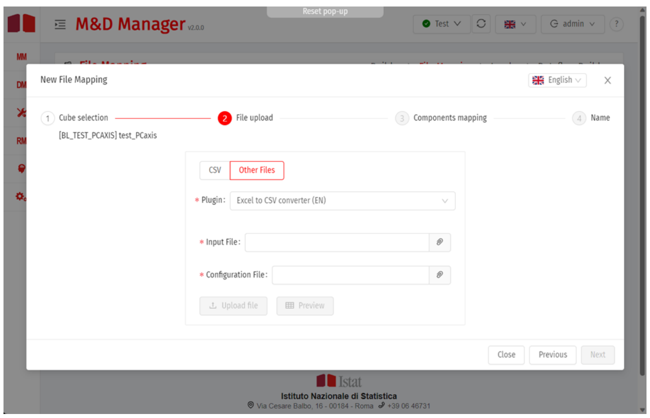
The frame allows the user to delete the file upload, to allow the selection of a different file.

Once the compulsory information has been filled in, the button "Upload File" will become active, with the click, the System will upload the file and activate the "Next" button.

After uploading the user can view the content of the file with the "Preview" button.



Upload File (Other formats)
The system allows a new mapping to be created and a new dataset to be loaded from a plugin, an input file and any .



In order to perform the conversion an input file is needed, whose type depends on the configuration of the selected plugin in the current Node.

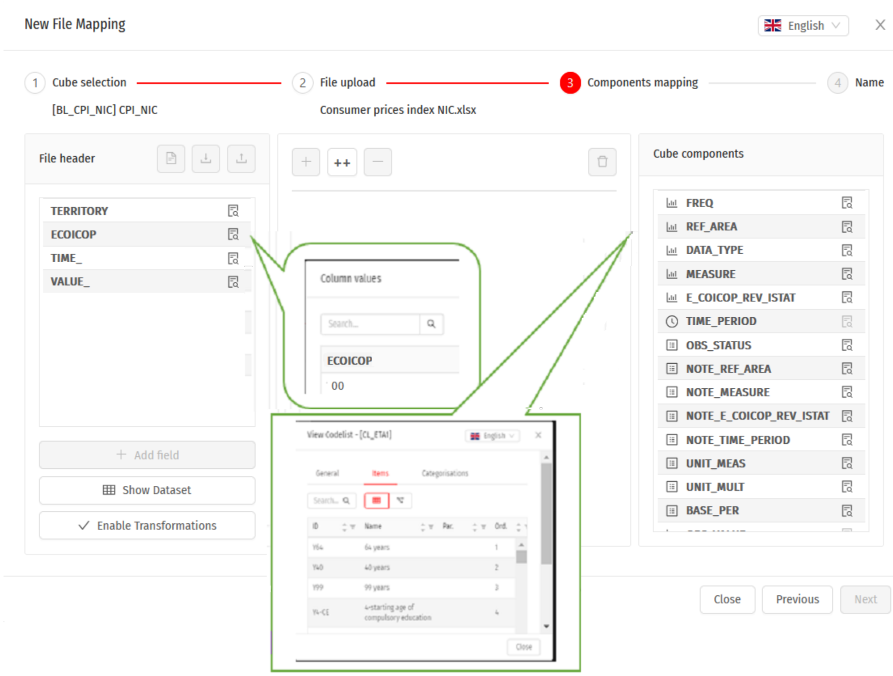
For the Excel 2 CSV Converter the user will have to select a .xls or .xlsx file as the plugin input and a configuration file as an .xml one. The PcAxis converter requires only an input file, which must be a .px file. For any other plugin, the input file is always required. The user will have to select a configuration and a process file if the plugin needs it (can be set in the node configuration, under the plugin tab.

After the Upload button has been clicked, a message indicating that the operation has been completed successfully is displayed and it will be possible to get a Preview of the transformed file by clicking on the "Preview" button.

After the upload process has finishedthe "Next" button can be clicked to proceed with the mapping step.

Component mapping
This step allows the mapping between the components of the Cube and those of the upload file.
The functions available are:
"+" to add a mapping after the selection of a Cube Component and the Header
"++" to add in the mapping all the correspondences between the Cube Component and the Header in automatic mode
"-" to remove a match from the mapping

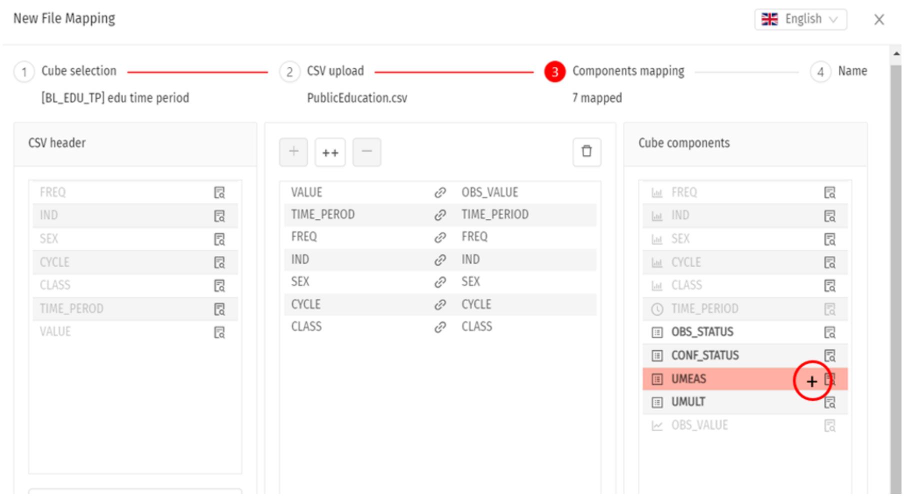
The “Show Dataset” button at the bottom activates a pop-up window displaying the file in table format.
The user can visualize the values contained in the components by clicking on the icon placed next to each element.
On the left side the elements and the values present in the loaded file are shown, on the right side the Codelists and the items related to the DSD of the selected Cube are also shown. The mapping is used to check the correspondence between the contents of the csv header and the components of the Cube.



Optional components are greyed out and can be selected in this step.
Component Mapping not only allows the user to map individual dimensions or attributes of a cube by associating it with a column in the CSV file, as seen in the previous section, but also allows the user to associate a fixed or conditional value. In particular, the following cases are supported:

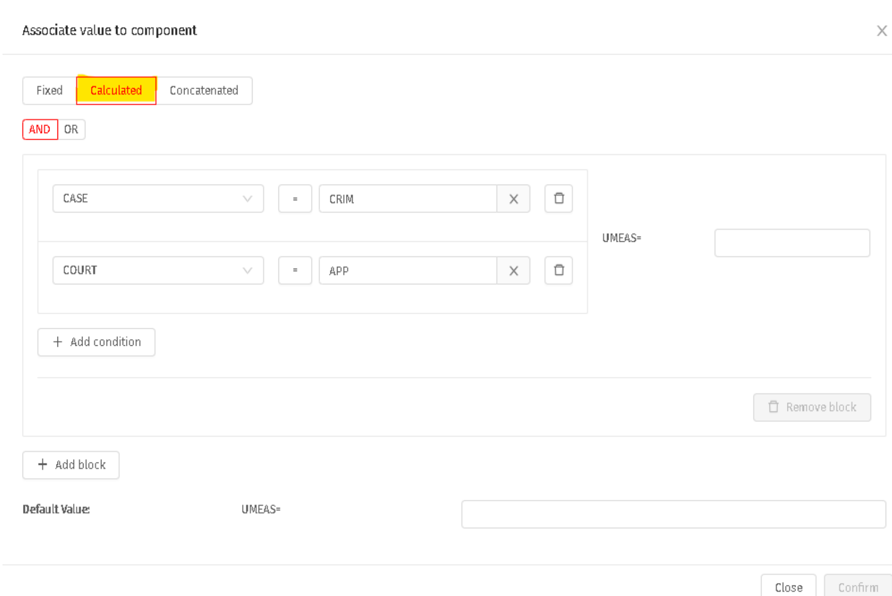
- **Fixed:** a fixed value is assigned for the component (dimensions or attributes) for all records (e.g., ‘A’)
- **Concat:** the value of the component will be defined as a concatenation of other components and/or fixed values (e.g. ‘START_’ @ [FREQ] @ ‘_FINE’)
- **CalculatedExpression:** the value of the component will be defined based on the value of other components (e.g. CASE WHEN [CASE] = ‘CRIM’ THEN ‘A1’ WHEN [CASE] = ‘IMPC’ AND [COURT] = ‘APP’ THEN ‘A2’ ELSE ‘A3’ END)

To create Component Mapping of any of these types to a component, the user must select the desired cube element (dimension or attribute) in order to activate the button with the “+” symbol.
Note: The “+” symbol is visible only when the element is selected.

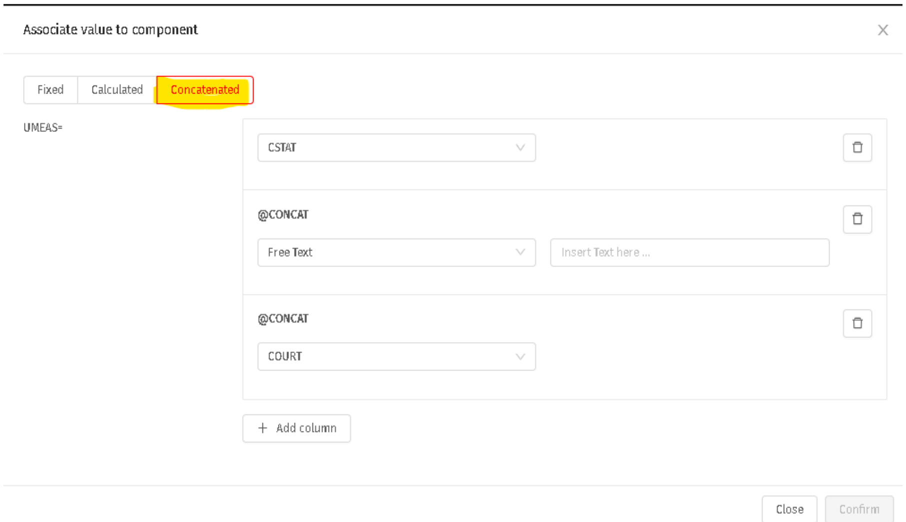


When the user clicks on the '+' symbol, it activates the option of choosing, for example, to enter a fixed value; if the component is encoded, the available values are displayed, otherwise a field for entering free text is displayed.

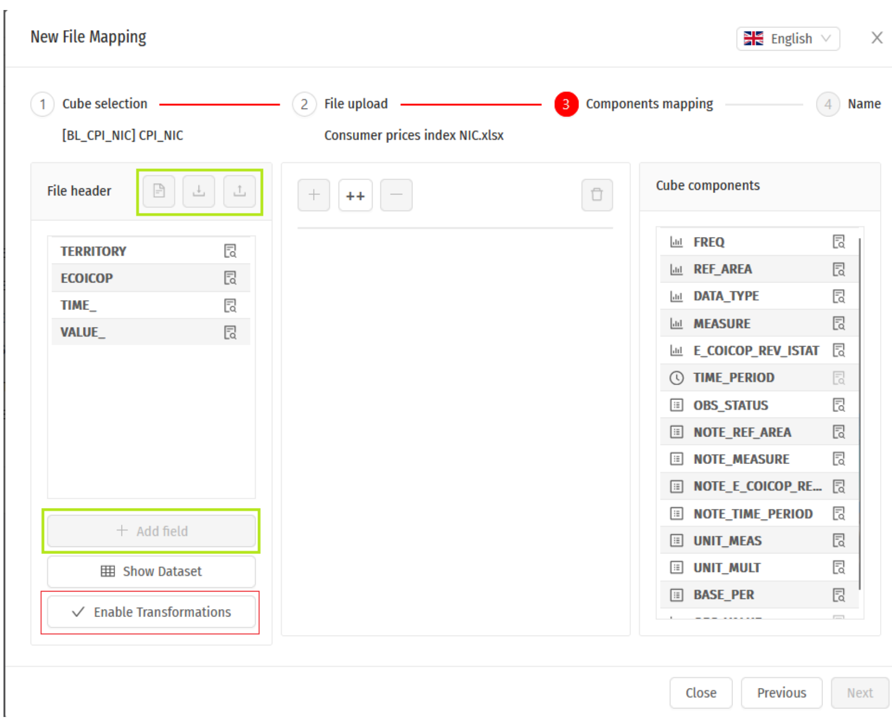
Another option is to 'Calculated' the value to put in our component in relation to the combination of values of other dimensions.
For component mappings of type 'Calculated' it will be necessary to enter one or more conditions, keeping in mind that they may be either all in AND or all in OR.
In addition, the DefaultValue field will be mandatory if the component is mandatory and optional otherwise.



The last option available is 'Concatenated' whereby it is possible, starting with 'Default' type components, i.e., those mapped directly from file, to concatenate them with each other and with free text to form a single expression to be mapped into the target component of the cube.

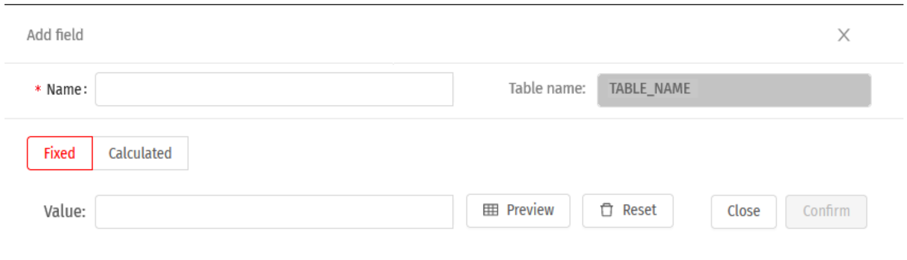


It is possible to create new calculated fields in addition to the existing ones in the CSV file. The "Enable Transformations" button activates the "Add field" button and the three option buttons above the fields list.

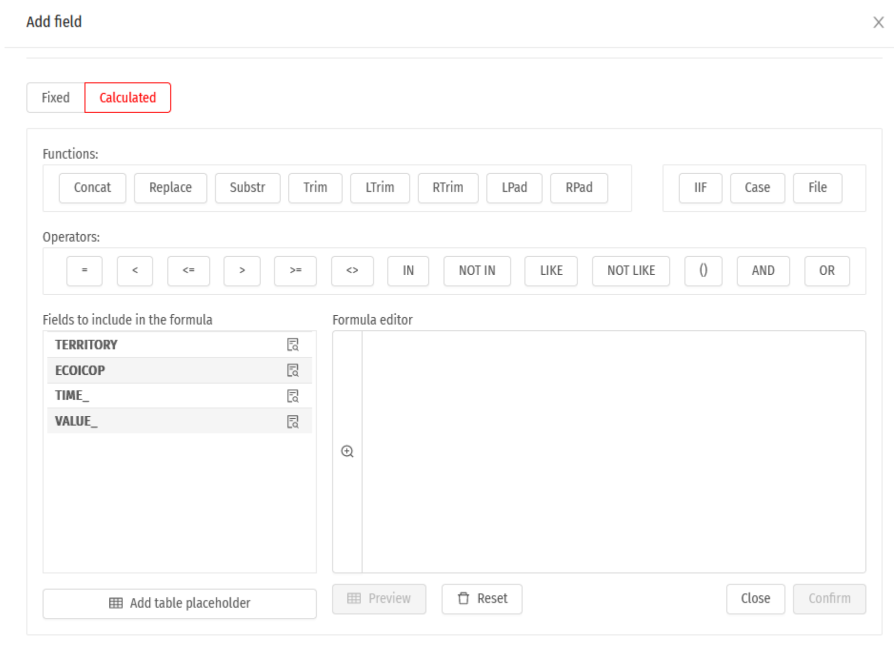


The selection of the "Add field" button opens a pop-up window in which the user can create a new field, assigning it a name and a value. The following two cases are supported:

- **Fixed:** a fixed value is assigned to the field for all records

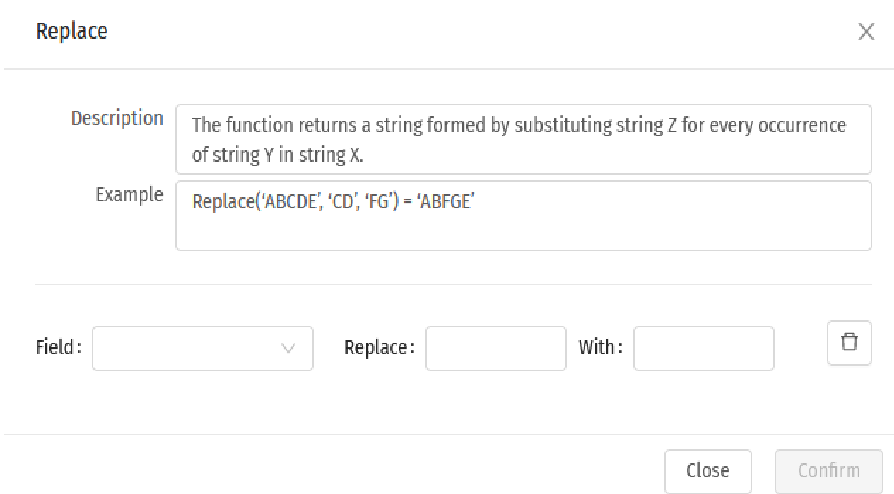


- **Calculated:** the value of the field can be defined by an sql expression including operators and string manipulation functions (e.g. REPLACE([FREQ], 'EU', 'IT'))



In both cases, the "Preview" button allows to verify the result, the 'Reset' button cleans the fixed value or the expressione entered in the "Formula editor". "Table name" shows the temporary table name used to verify the result of the created field.

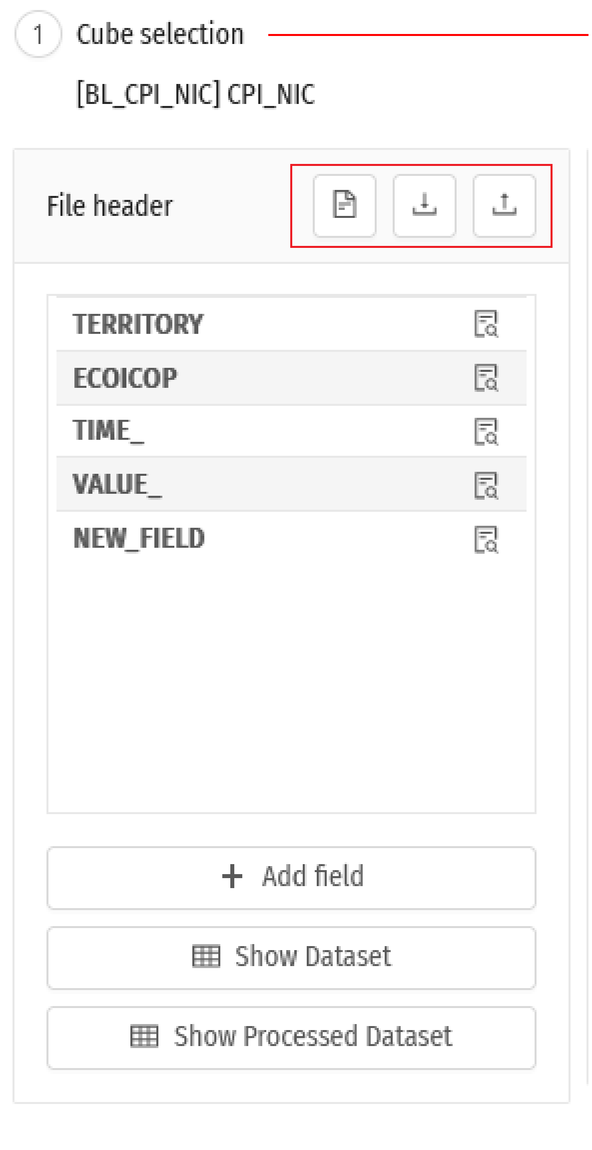
In the **Calculated tab**, each function button opens a pop-up in which the user can select the field and enter the parameters required by the function.
e.g.: Replace function



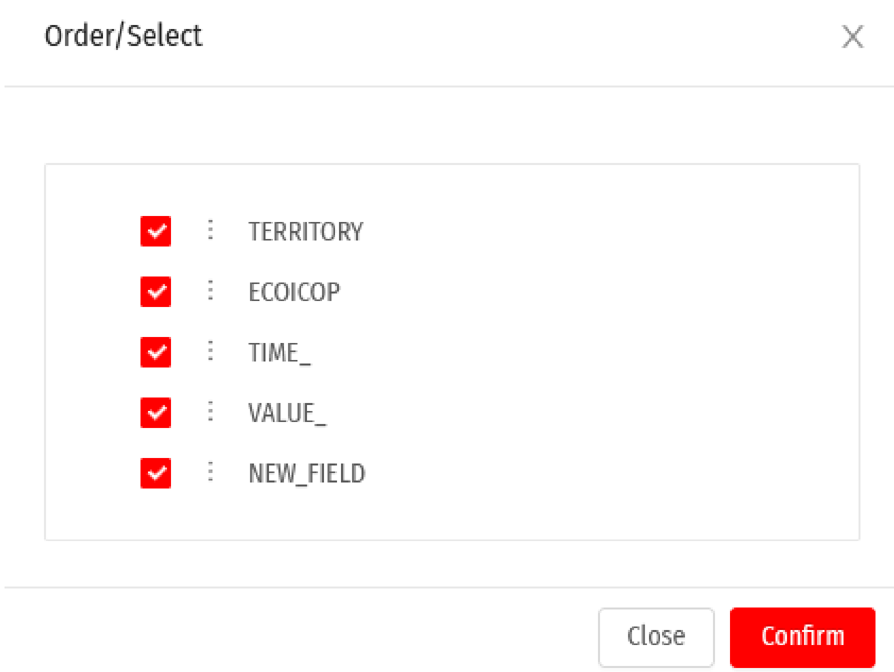
The pop-up window shows the description and an example of the function.
After selecting the field and entering the parameters, the "Confirm" button allows the user to insert and modify the function in the "Formula editor".
It is also possible to enter the function template into the editor with a right click on the function button.
The function or its template can be modified in the "Formula editor" using the "Operators" buttons and the fields on the left side. If any text has been selected in the editor, it will be replaced by the operator or field chosen by the user.
Note: the SQL "Select" statement does not have to be included in order to evaluate the new field; it can only be used within the expression, in this case the name of the table will be replaced by the placeholder "[CSVINPUTTABLE]"
The following statement is wrong: **SELECT** IIF([TIME_PERIOD] = (Select max([TIME_PERIOD]) from [CSVINPUTTABLE]), '0', '1') **FROM [CSVINPUTTABLE]**
The right one is: IIF([TIME_PERIOD] = (Select max([TIME_PERIOD]) from [CSVINPUTTABLE]), '0', '1')

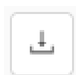
"Confirm" button creates the new field and adds it to the fields list.

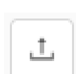
Option Buttons

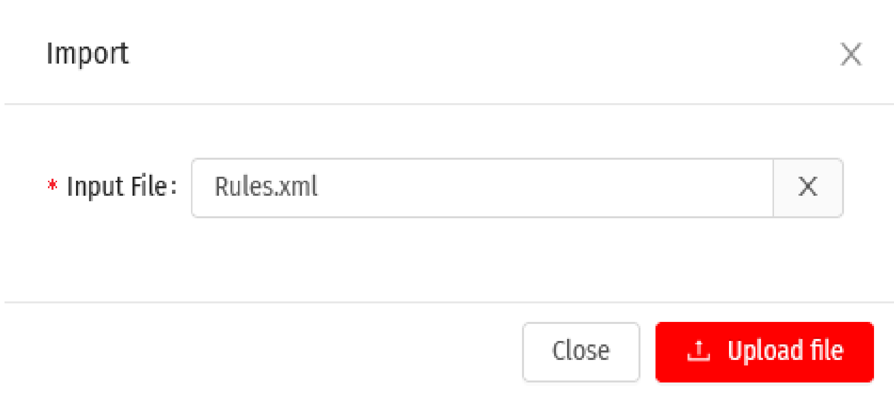


-  **Order/Select button:** it allows to modify the order and selection of the fields for preview and download

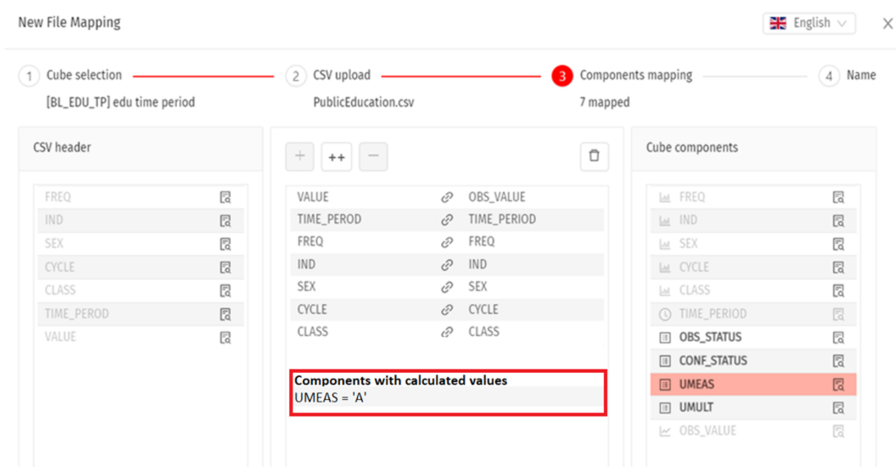


-  **Download rules:** it allows to save the state of the processed fields list, including new fields, order and selection

-  **Import rules:** it allows to load the last saved state of the processed fields list, in order to continue the transformation of the CSV fields

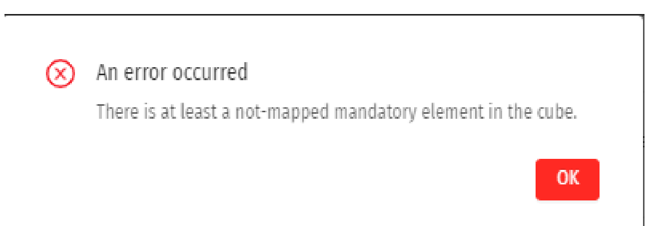


Once the component is mapped with the value, the mapping will be shown in the appropriate section of the center column with the chosen value.
It can still be removed by clicking the '-' button as for other types of component mapping.



Note For import from SDMX-ML file, it is not possible to define components with fixed or calculated values, also it is not possible to define calculated or conditionated values for OBS_VALUE.

In order to continue with the creation, it is necessary to map all the mandatory components of the Cube, otherwise, the System allows the user to continue but prevents from saving with an error message.

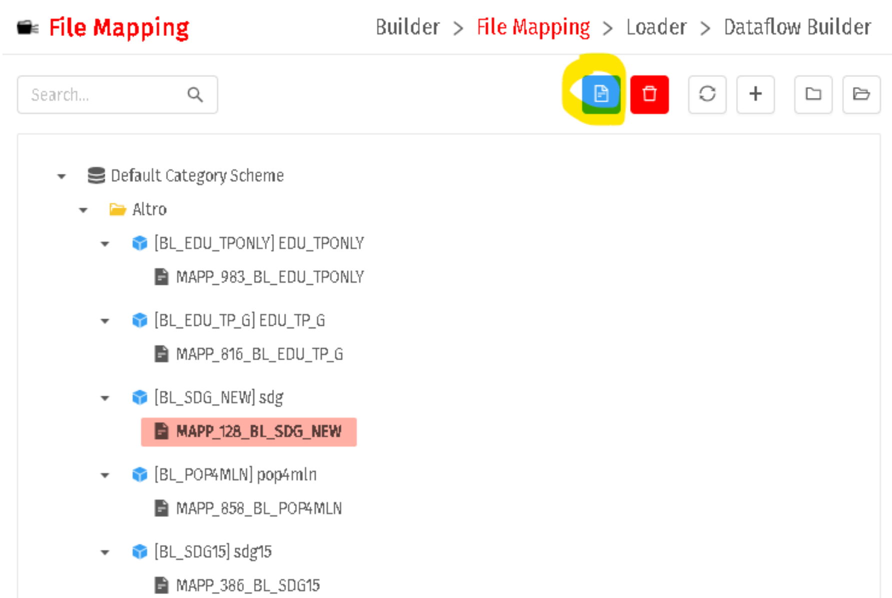


Name
The save mask contains the fields:

- *Mapping Name* (mandatory, automatically created but editable)
- *Mapping Description* (optional)

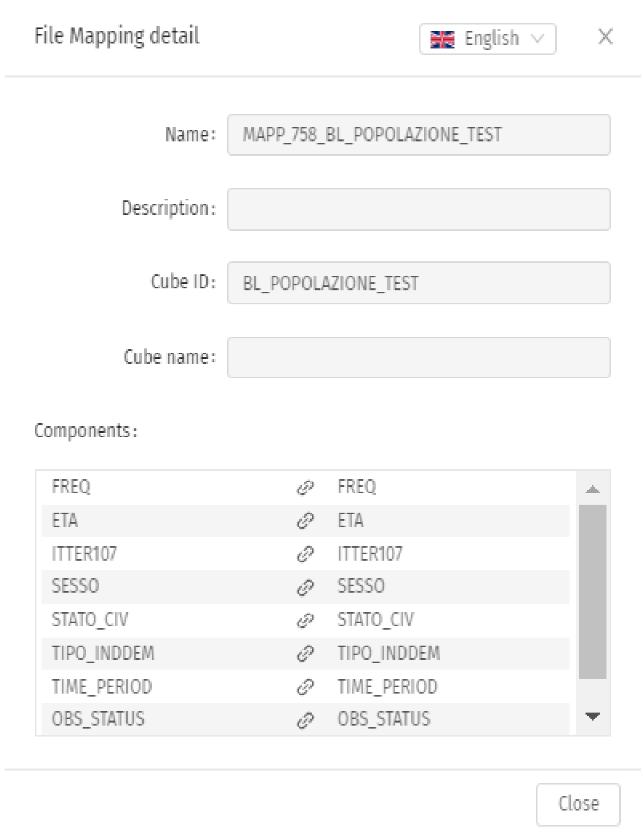
In case the OBS_VALUE component has not been mapped to any column in the csv file (this is allowed), if all other mandatory components of the cube are correctly mapped, a warning message will be shown to the user in the last step of the wizard for creating a mapping: "Be careful! Observation value has not been mapped".
Once the values in the fields have been set, the user selects the "Save" button.
The system notifies the user that the mapping has been created by closing the creation pop-up window and displaying the updated mapping list with the new item.

Viewing or deleting an existing mapping
The "Show Details" button is selectable from the tree-element of the Mapping list.

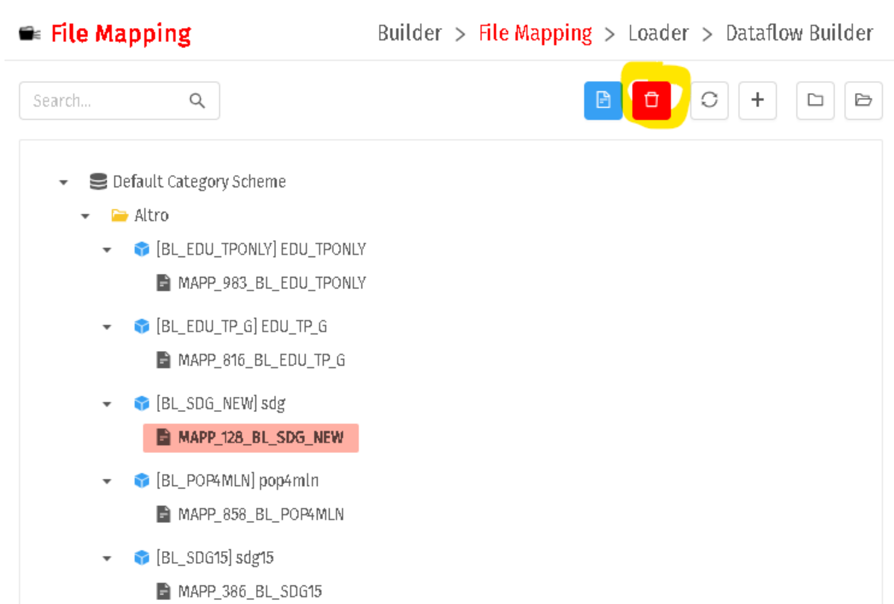


The System displays a pop-up window showing the following information of the Mapping in read-only mode:

- *Mapping Name*
- *Mapping Description*
- *Cube Identifier*
- *Cube Name*
- *Component List*



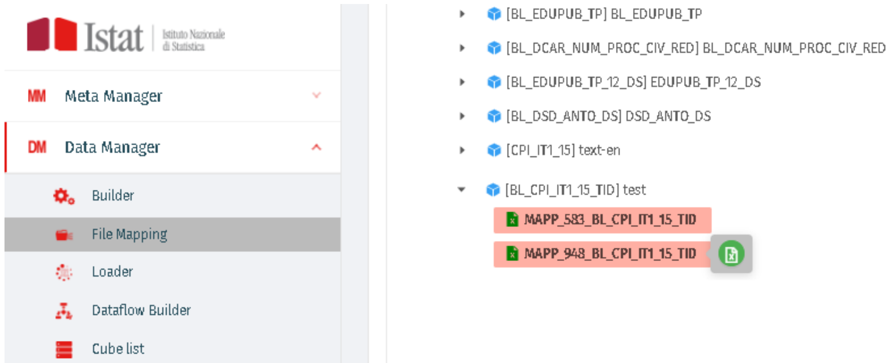
Deleting a mapping is possible from the "Delete Mapping" icon, which can be selected from the tree branch in the Mapping list or from the "Delete selected Mapping" button at the top of the screen.



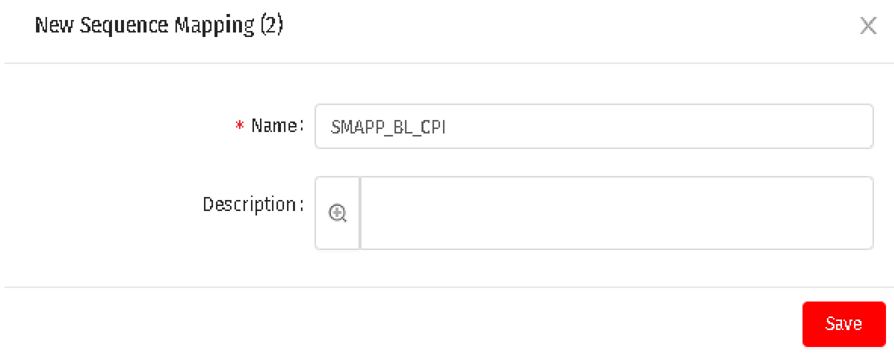
Creating a Sequence Mapping

The user has the option of defining a new type of mapping called Sequence Mapping consisting of a set of Excel mappings created on the same cube (in the manner seen above) and corresponding to a set of sheets in an Excel file.

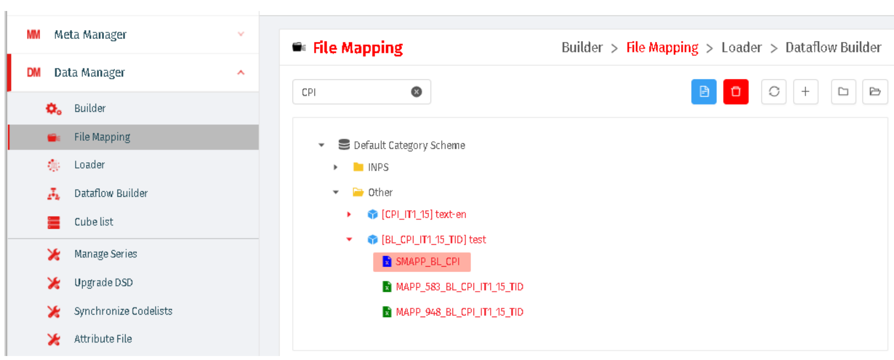
First the user must choose the excel mappings already constructed (of a cube) with which to create a Sequence Mapping subsequently right-clicking activates the green “Create Sequence Mapping” button.



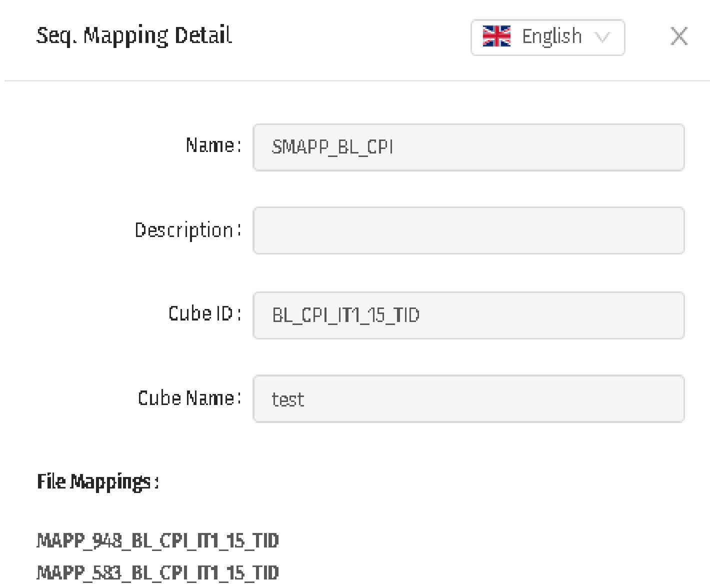
At this point a small window opens in which the user can give a name and description to the defined sequence.



Once saved, the Sequence will always appear below the chosen cube, at the same level as the other mappings, with its own identifying icon.
If the user chooses it, the Delete and View (of details) buttons will be activated:



In the details window there will be information about the cube and the mapping sequence used.



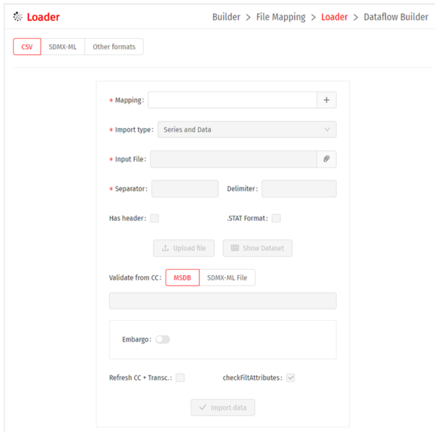
It is not possible to edit a Sequence Mapping but only to create or delete it.

8.1.3 Loader

The system displays the mask containing the possible file formats from which data is acquired. Choosable formats are CSV, SDMX-ML, other formats (uses the plugins to convert different types of files into CSV).

According to the choice, fields in the window may slightly change. The required fields for loading from CSV file are:

- *Mapping* (mandatory, to be chosen from a list)
- *Import type* (mandatory, to be chosen from a list)
- *File* (mandatory)
- *Separator* (mandatory, it is pre-set with ";")
- *Delimiter* (optional)
- *Flag "Has header "* (optional, the default is ON)
- *Flag "Format .Stat "* (optional, the default is OFF)



The user can select the mapping by opening the list of mappings and by clicking on the "+" in the list box, which will be populated with the chosen mapping.
In the field "Import type" the user sets the type of loading between the values:

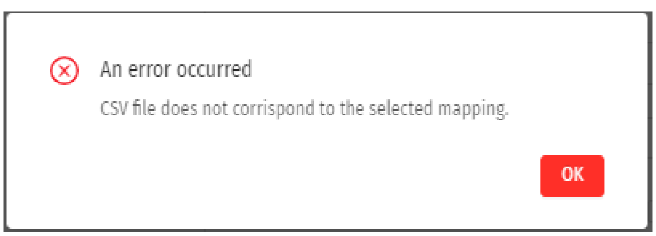
- Series and Data
- data
- Series

Generally, the loading includes both the series and the data, but the user may choose to load the series first (e.g. for control) and then the data.

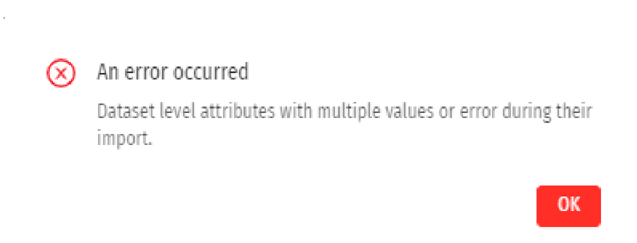
If only the data is loaded, the series must already be present, which is useful when updating the data.

The other fields for loading the CSV are similar to those provided in the "File Mapping".
Once all mandatory fields have been filled in the "Load CSV" button will be activated. After loading the CSV it is possible to view the data ("Show Dataset" button) , moreover the "Import Data" button is activated to complete the loading.

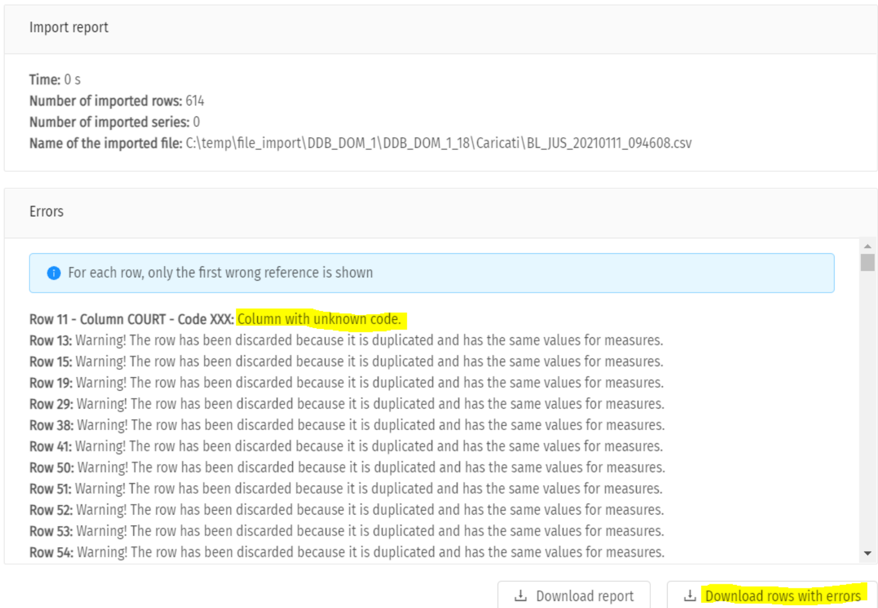
The system, during the importation process, checks for errors in consistency, formatting and correctness of the CSV file. If not, an error message is displayed.



Moreover the import is blocked when there are two duplicate rows in the csv file but with different OBS_VALUE.

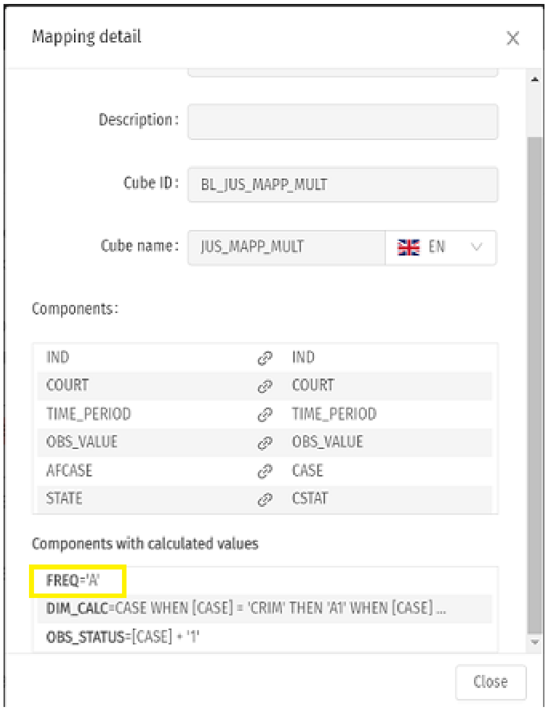


User can also download the report with the details of the error.
Finally, there are cases (e.g. reference to a code that does not exist in a codelist) in which, despite the presence of errors, the process still ends successfully, with a certain number of rows being discarded. The report details the discarded lines and, for each of them, the first error encountered. In this case, in addition to the report, it is also possible to download a csv file containing all the rows with errors that have been discarded.

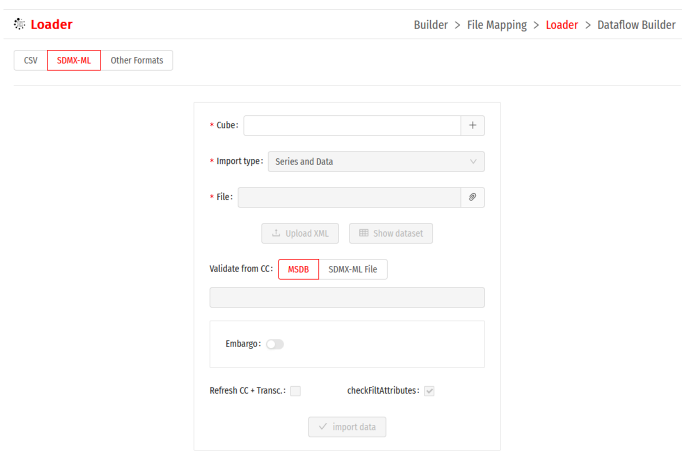


Even in the case of a successful import, the system displays a pop-up window with the details of the import and the possibility of downloading the report.

Note: if the selected mapping contains dimensions which were manually set as "fixed", such value will be editable during the Loading phase.

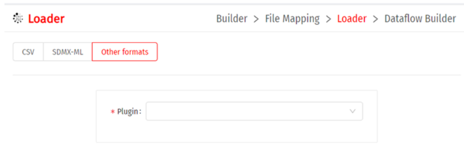


The user has the possibility to upload a file in "SDMX-ML" format. In this case, the following mask is available for data entry:

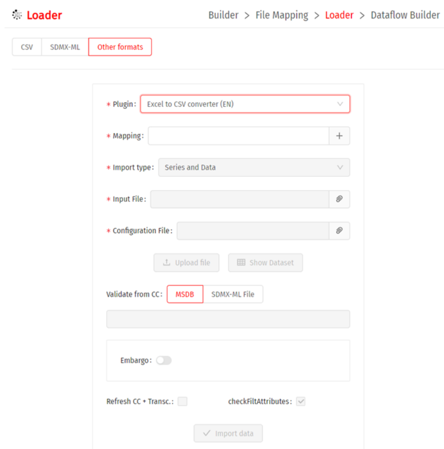


In this case the loading is done by choosing directly a Cube (skipping the mapping selection) and an SDMX file containing the data.
Pressing "+" in the list box "Cube" the System shows the Category Scheme tree for selecting the Cube, while pressing "+" in the list box "File" allows the user to search for the SDMX file in the filesystem.
The third mandatory field is the "Import Type" analogous when loading CSV files.
Once the mandatory fields are populated the "Load XML" button is activated; if loading is successful the "Import data" button is activated and the user can carry out the import.

The user has also the possibility to upload a file in "Other formats" format. This uses a plugin to convert different kinds of files into CSV.
In this case, the following mask is available for data entry:



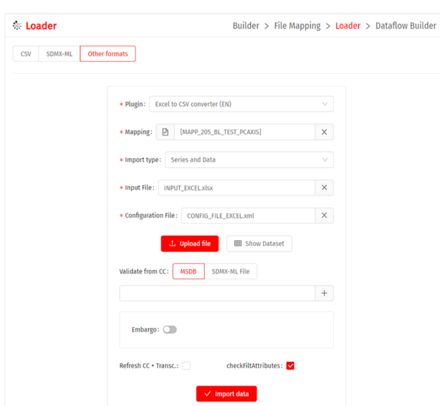
The user must choose the type of Plugin installed and configured for the Cube, then the rest of the component is rendered:



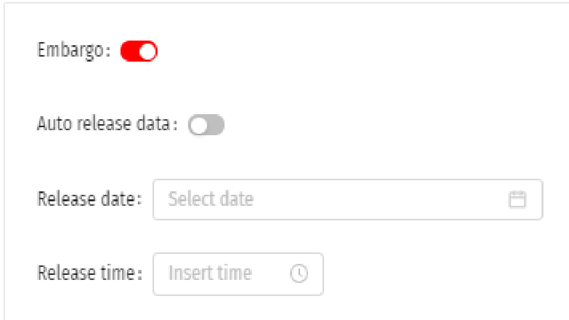
Loading takes place by choosing a Mapping. Pressing "+" in the mandatory list box Mapping the System shows the Category Scheme tree for selecting the Cube and the relative Mapping or Sequence Mapping.

The second mandatory field is the "Import type" similar to loading CSV files, while pressing "+" in the "Input file" list box the user can search for the type of file with the extension defined in the plugin configuration.

Once the mandatory fields have been filled, the "Upload file" button is activated and the user can upload the file. If the file conversion is successful, the "Show Dataset" button is activated and the loading can continue.



There are features that are common to all types of data imports.
The "Embargo" function is available to deny display of loaded data.

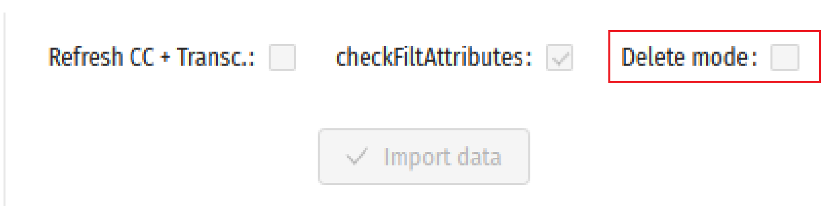


The embargo can be removed manually with the “Remove Embargo” button which can be found in the “Cubes List” section on the specific cube the user is working on. Otherwise it can be removed automatically by selecting the “Automatic data release” and a release date (functionality not active at the moment).
The “Regenerate CC+Transcoding” flag allows the loading of a data item recreating also the Transcoding and the Content Constraint if they are already present.

Once data has been loaded, the checkFiltAttributes function is used to verify consistency between the attributes stored in the Filt Table (attributes with attachement Level Group or Dimension Group) and the dimensions linked to the DSD used in the cube. If the check fails, the cube is rolled back to the situation when loading begun.

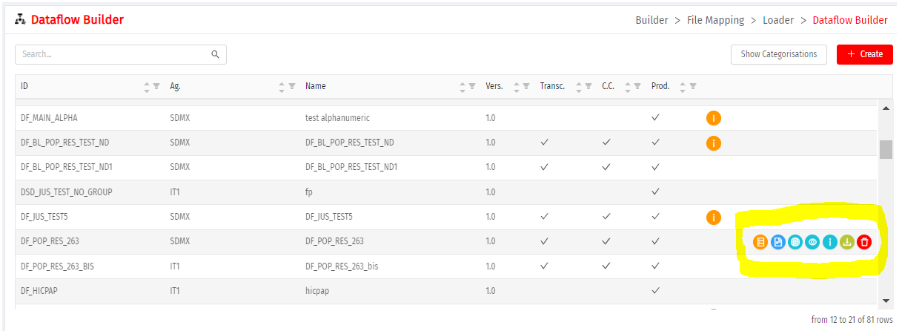
If the user decides to validate the data that he is loading with a content constraint, he can choose an existing one from the “Validate from CC” list or he has the option of selecting content constraints from SDMX-ML files. The latter content constraints will not be imported into the MSDB but will be used as a runtime object. Only one content constraint is selectable and, after the selection, the user continues with the import operation. If no content constraint is selected, validation will no longer be carried out, so the user will be able to start importing the file without the use of any validation.

The “Delete mode” option allows to delete cube data that matches that in the importing file.



8.1.4 Dataflow Builder

The Dataflow Builder window displays all the Dataflows in table format. Clicking the “Show Categories” button a pop-up window with a tree view of the Dataflows is opend.



For each Dataflow the following fields are present:

- *ID of the Dataflow*
- *Agency ID*
- *Name of the Dataflow*
- *Version of the Dataflow*
- *Transcoding check*
- *Content Constraint Check*
- *Check that indicates if the Dataflow is or is not in Production*

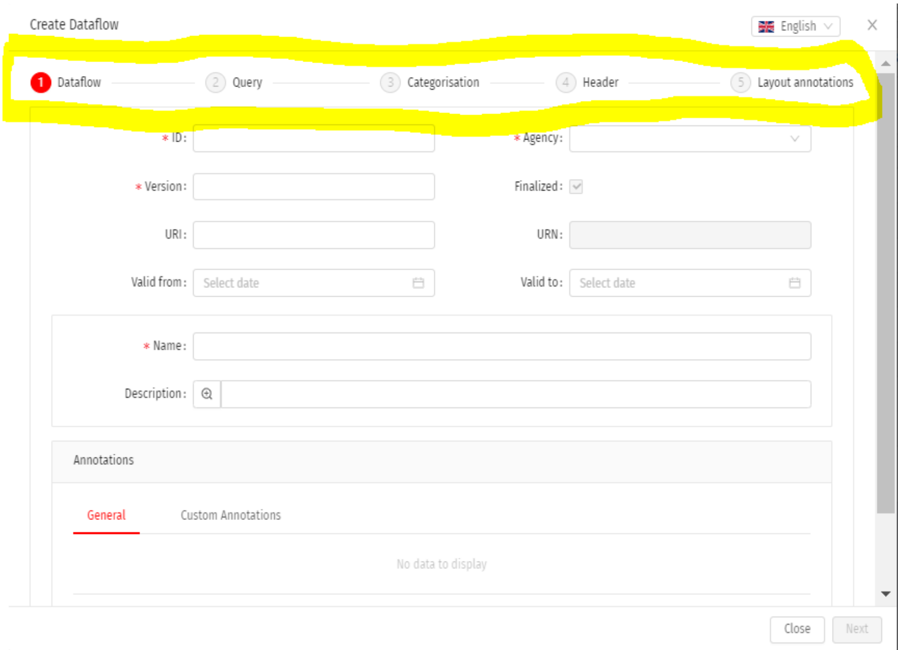
It is possible to:

- **Create a new Dataflow**
- **Perform operations (display, delete, etc.) on an existing Dataflow.**

for the operations on the Dataflow please refer to the section *Dataflow management*

Create a new Dataflow

The selection of the “New” button, in the upper right corner, opens a pop-up window for the setting of the necessary information for the creation of the Dataflow.
The system displays the steps to follow in the bar at the top: it is possible to move to the next step with the “Next” button or by selecting the step directly from the bar.
The user can go back to the previous steps to perform changes or to check what has been set.

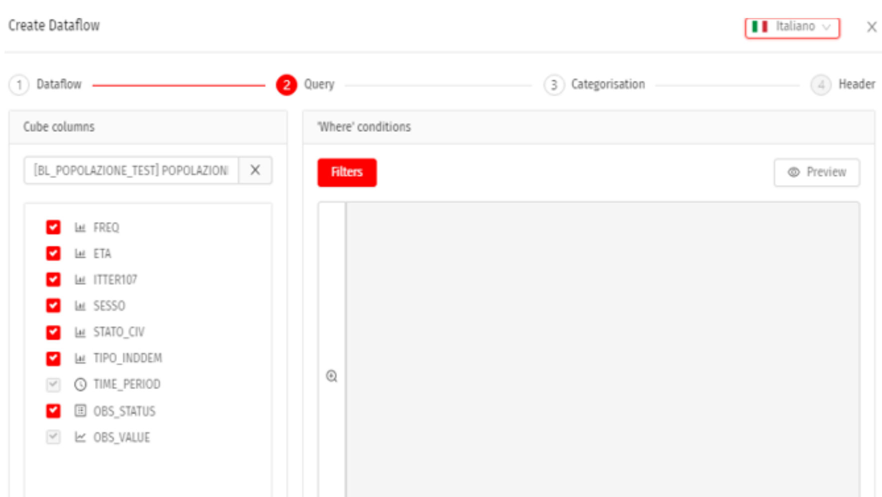


Dataflow
The first step opens the Dataflow's general data acquisition mask. As soon as the mandatory fields are filled in:

- *ID*
- *Version*
- *Agency*
- *Name*

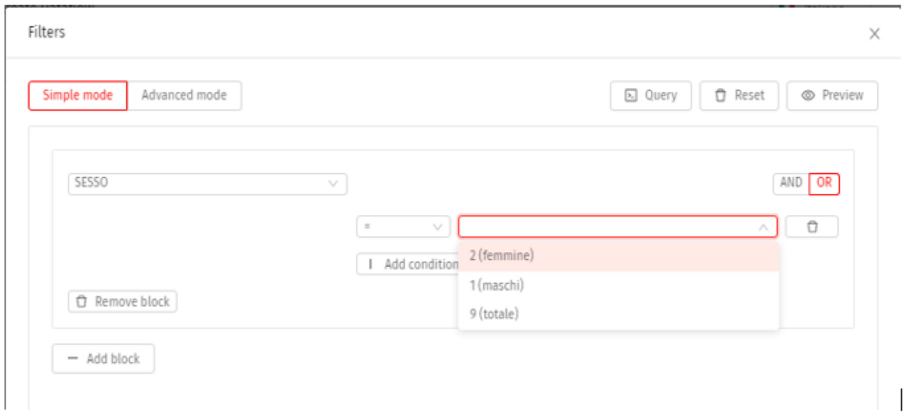
the "Next" button becomes active to proceed to the next step.

Query
In this mask the user chooses the Dataflow source cube from the tree of Category Schemes and composes the query that identifies the Dataflow.

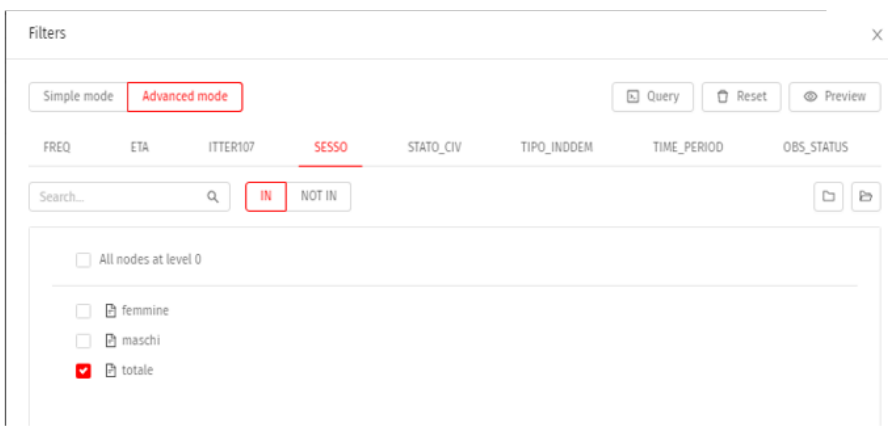


In the left window the user chooses the dimensions and attributes and with the "Filters" button composes the "where condition" that is the filter to be applied on the data.

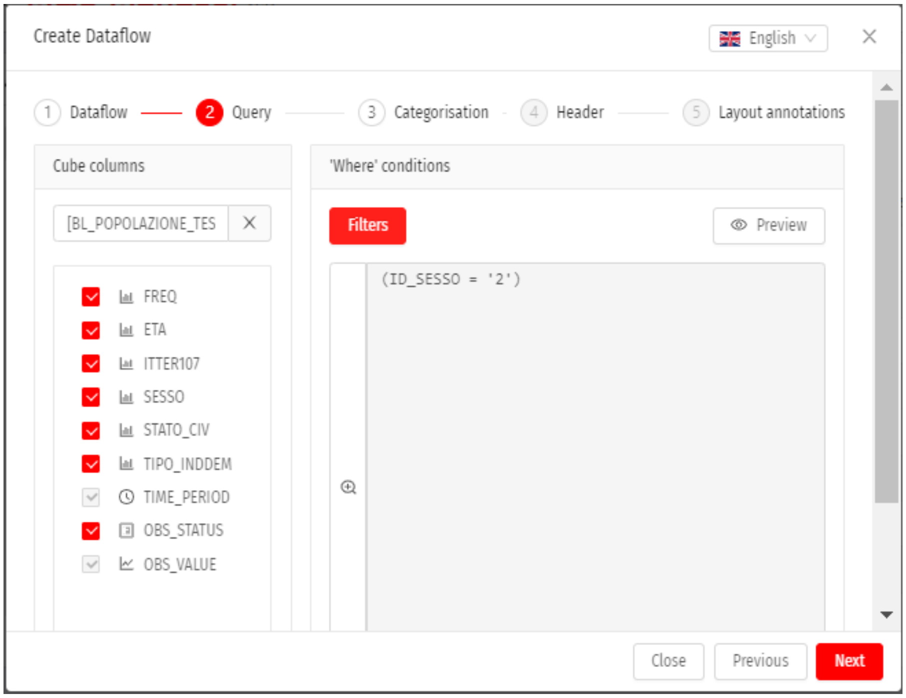
The filter can be created in simple mode ("Simple Mode" button) or in advanced mode ("Adanced Mode" button).
In simple mode the user can write the query in a guided way by choosing a dimension or attribute, operator and values and combining constructs with "and" or "or" operators.



In the Advanced mode the dimensions or attributes are presented in sequential tabs where the user selects the values for the creation of the Dataflow.
This way the System proposes for each element only the values actually present in the Cube, based on the choices made previously, so that the query does not return an empty set.

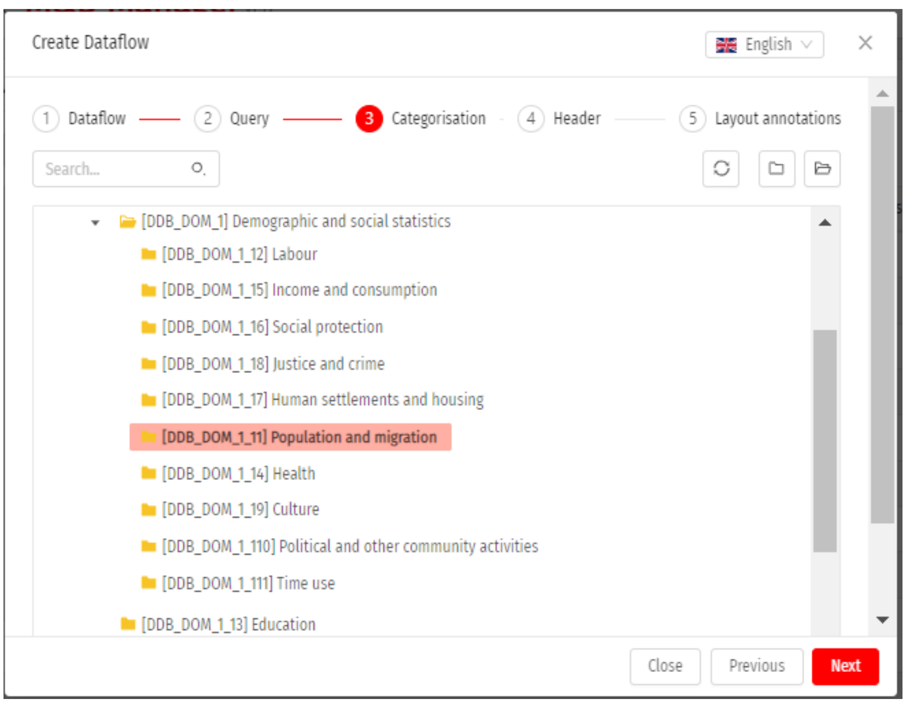


In both cases, the system displays the instruction corresponding to the query.



Categorisation

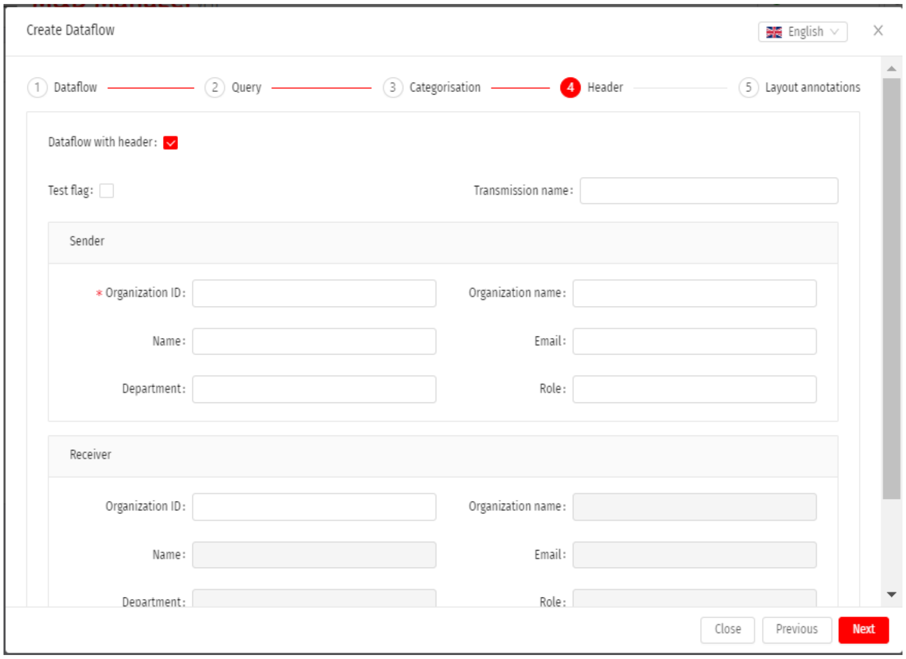
In this mask, the user associates the Dataflow to a category by selecting it from the Category Patterns tree.



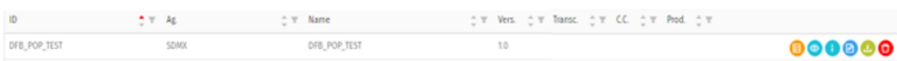
Header

In this section the user decides whether to insert a header to the Dataflow by acting on the “Dataflow with header” flag.

If the flag is selected the window shows the list boxes where to insert the information about “sender” and “receiver” prefixes.



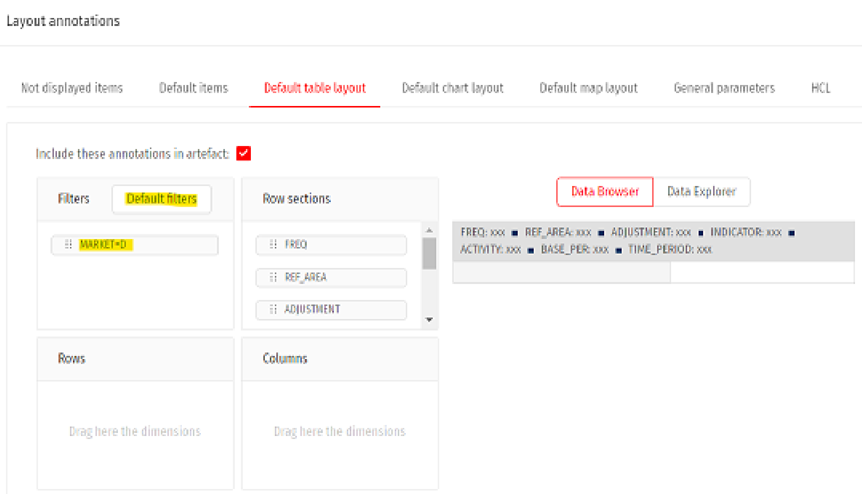
At the end of the 4 steps, pressing the "Save" button, the Dataflow is saved and inserted in the list.



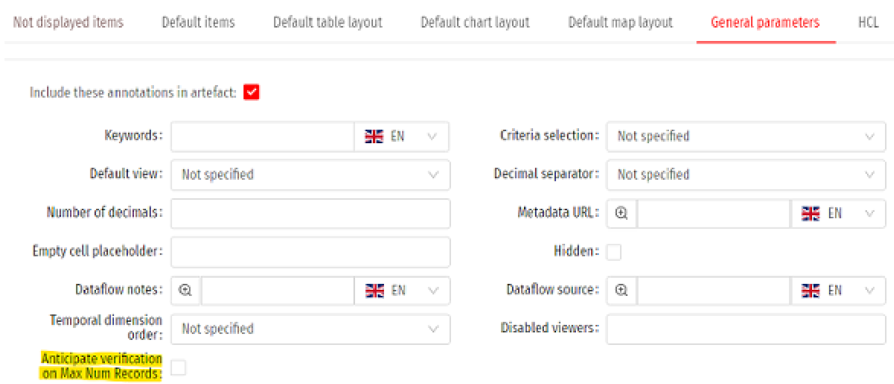
As soon as the Dataflow is created, the fields that identify it (ID, Agency, Name, Version) are valorized, while the fields: Transcoding, Content Constraint , Production are not valorized, for these it is necessary the publication.
Moreover, on the right side the following functionalities are available, as explained in the section *Dataflow management*:

- *Display Annotation*
- *Publication*
- *Preview on the Data Browser*
- *Preview*
- *Additional Information*
- *View/Edit*
- *Download*
- *Delete*

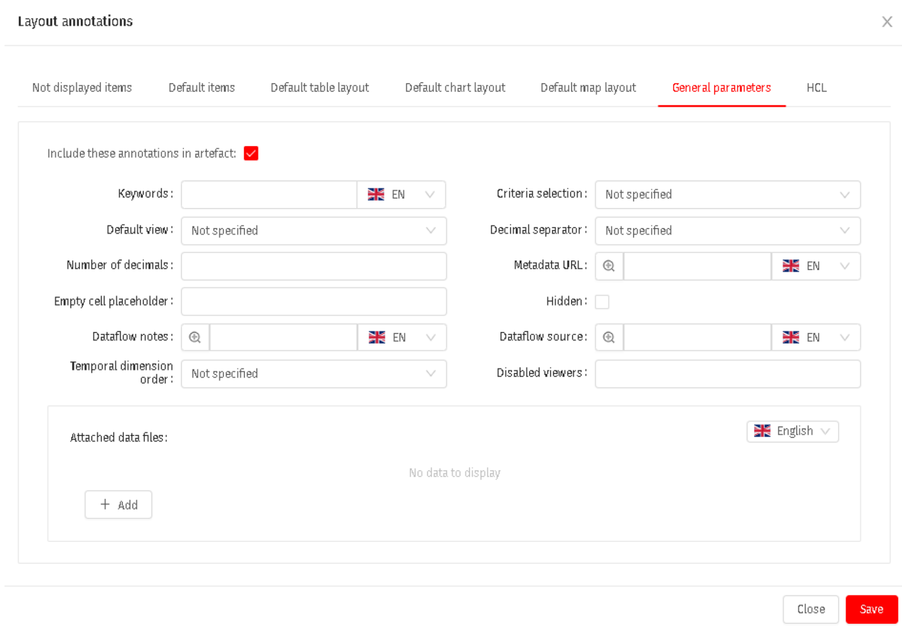
Layout Annotations
In this section the user decides whether to insert annotations at the level of dimensions to be hidden or to insert annotations that set the graphic structure of the table, chart or map.
In addition, other annotations can be set (such as decimal separator, value for empty table cells, etc.).
It is also possible to set specific items as default for dimensions specified in filters for all types of visualization (table, chart and map).



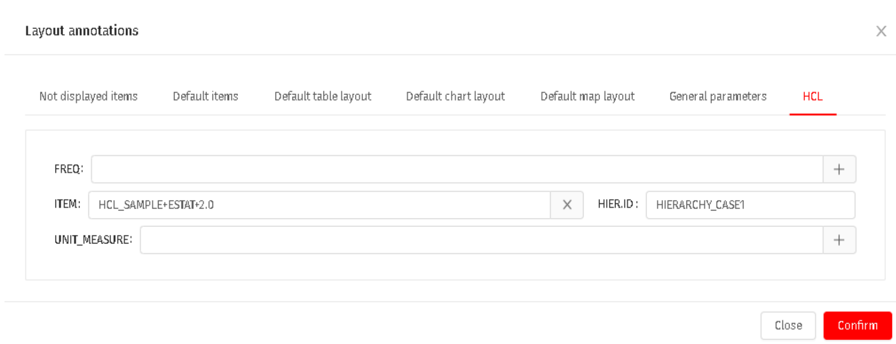
Another functionality has also been introduced to check in advance whether the threshold of the maximum number of records viewable in Data Browser has been exceeded by estimating the number of resulting observations. This annotation can be set in the layout annotation section, under the general parameters tab.



For all annotations' meaning please refer to the paragraph *Other node configurations*.



In the "HCL" tab, the user can enter the reference to the Hierarchical Codelists (hcl) already in the system for all desired dimensions and coded attributes.
In addition to the HCL codelist, the user must also specify one of the hierarchies in the codelist:



8.2 Dataflow Management

In this section we describe the functionalities available for Dataflow management:

- **View Annotation**
- **Publication**
- **Preview on the Data Browser**
- **Preview**
- **Additional Information**
- **View/Edit**
- **Download**
- **Delete**

some of these will be discussed in more detail in the following sections.

Display Annotation

The presence of an annotation is indicated by the symbol "i". The click allows the user to visualize the annotation details.

Publication

It allows the user to set the Dataflow in production, see section: *How to put a Dataflow into production*

Preview on the Data Browser

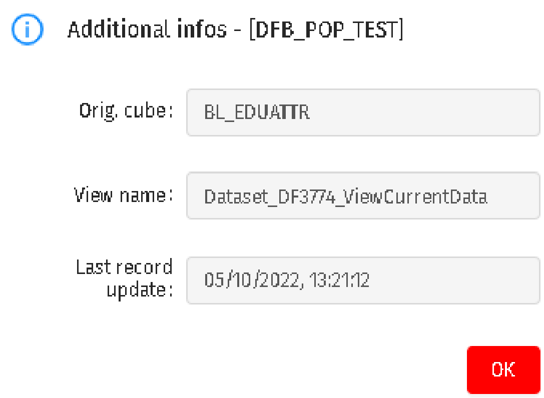
The icon is visible only if the Dataflow has been published and allows the visualization of the Dataflow with the Data Browser tool associated to the application, if present.

Preview

The information contained in the Dataflow is shown in table format with the possibility of sorting and filtering the columns or searching for items.

Additional information

The "Additional information" button allows the user to display information about the Cube and the corresponding view of the Dataflow.
This window also returns the information on the last update of the dataflow.



View/Edit

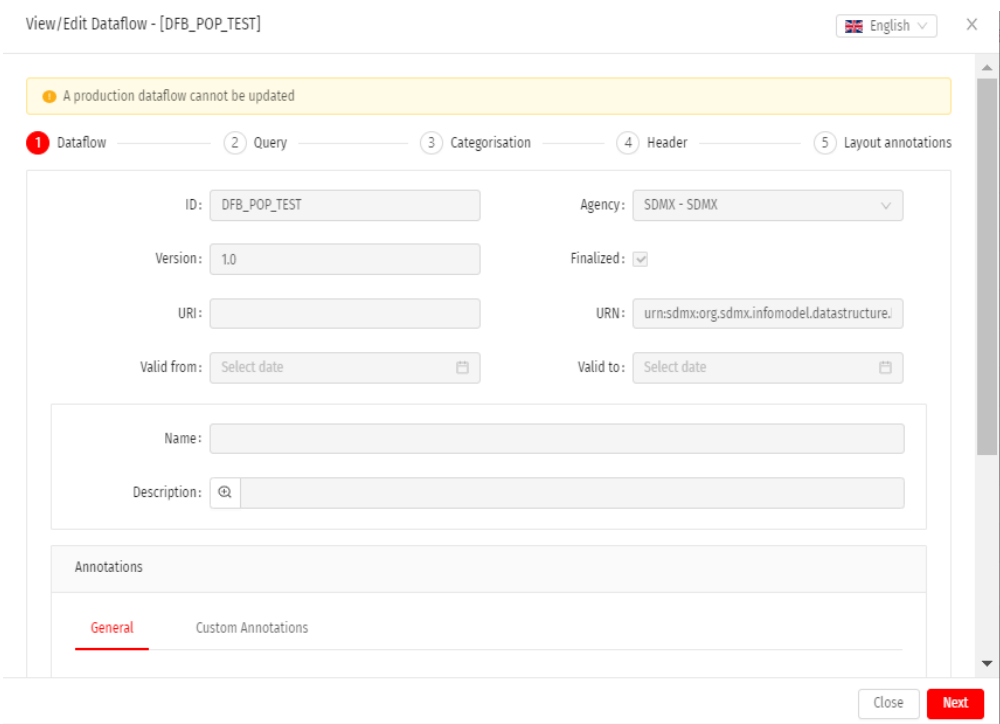
The "View/Edit" function allows the user to view or edit a Dataflow, for this last part see section: *How to update an existing Dataflow*

The system displays all the Dataflow elements:

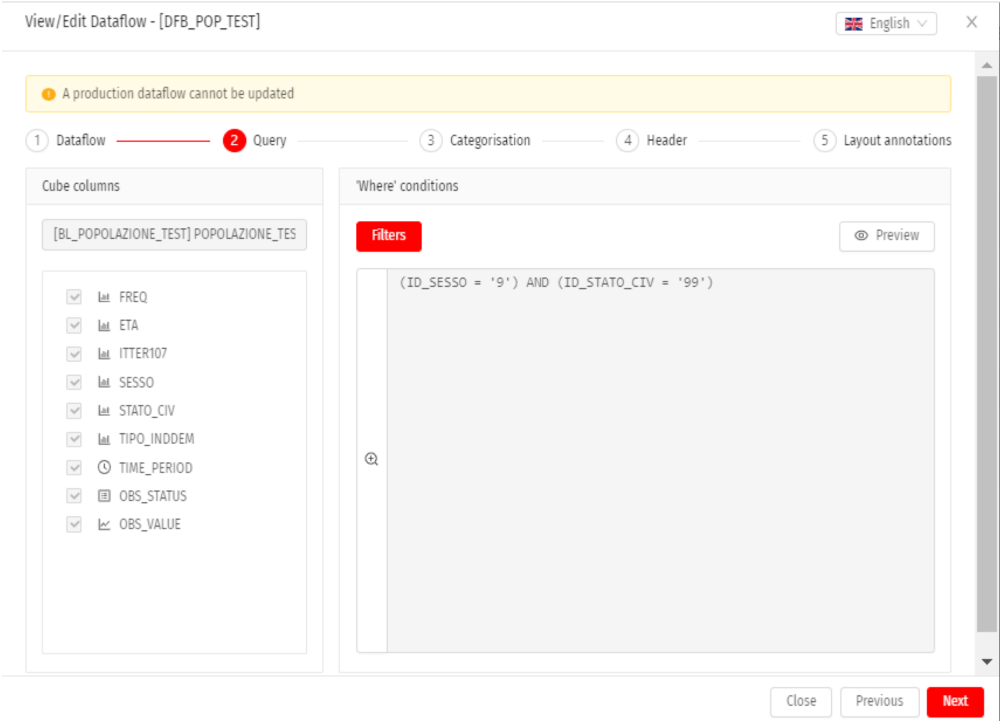
- *General Data*
- *Queries*
- *Categorisation*
- *Header*
- *Layout annotations*

in windows identical to those seen in the creation phase of the Dataflow.

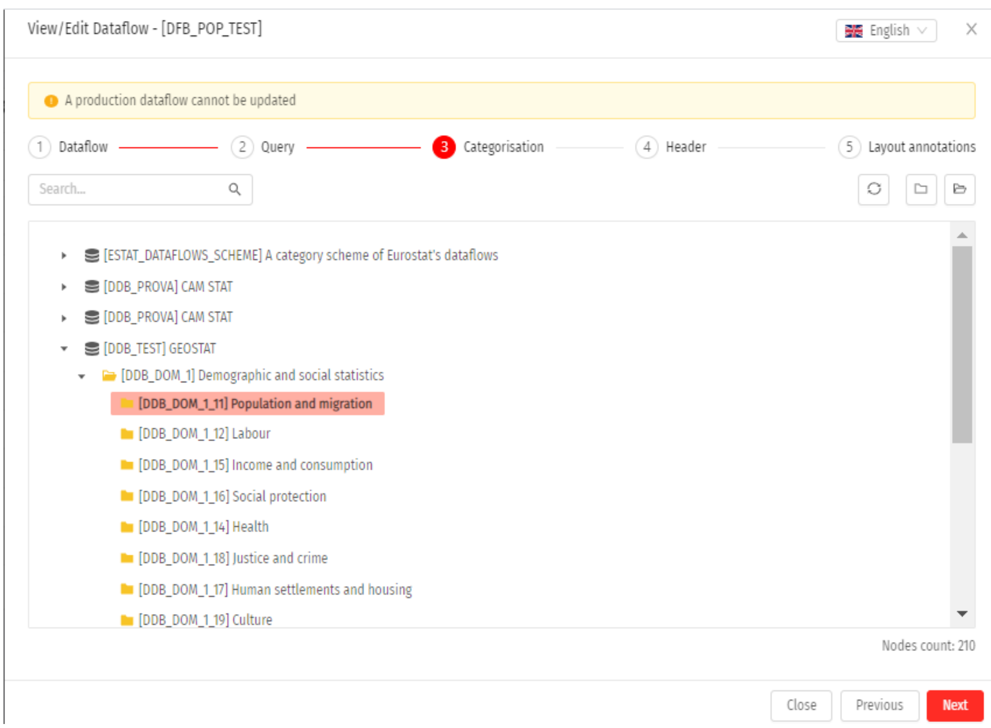
In the mask "General data" the general information of the Dataflow are shown.



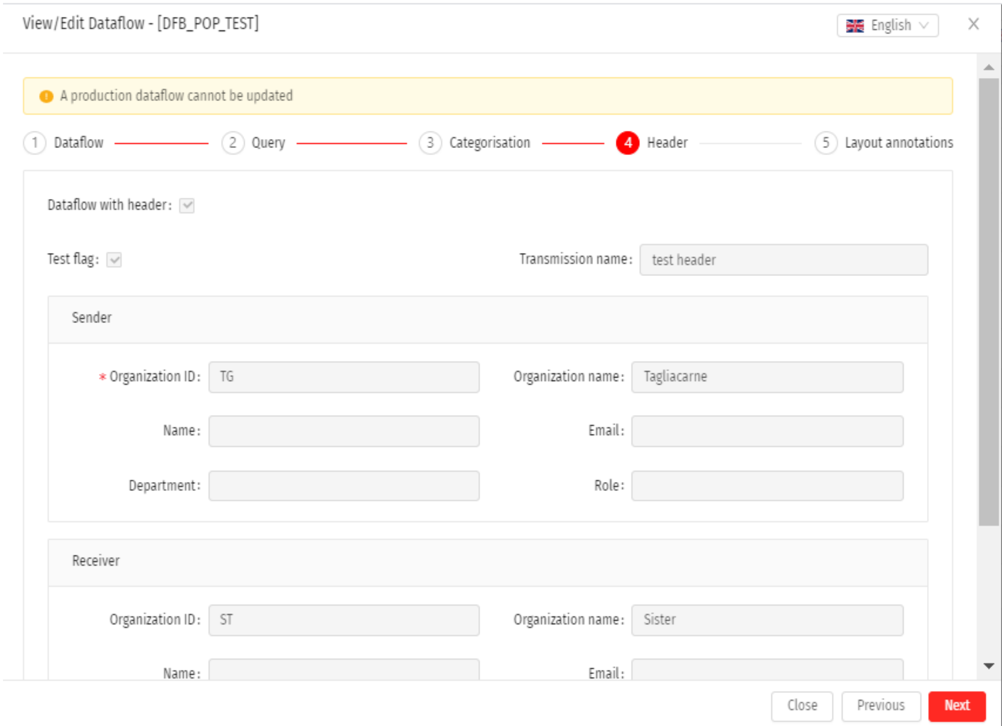
The "Query" mask displays the selected cube elements and the defined filter.



The "Categorisation" mask displays the category to which the Dataflow is associated.

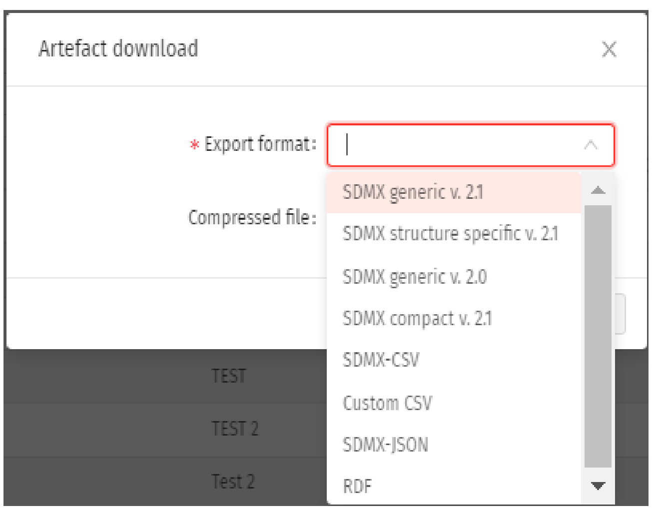


The "Header" mask displays information about the header if it has been selected.

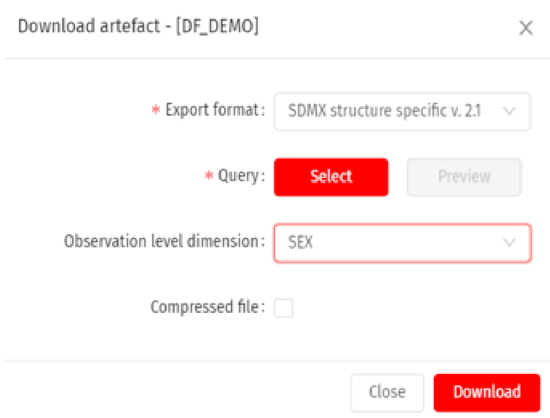


Download

The "Download" button allows the user to download the Dataflow in different formats. It is possible to download in compressed format.



"SDMX Structure Specific v 2.1" and "SDMX-JSON" are special data download formats in which the user can select the size to be used as the observation level



and get the downloaded data in the chosen form with the dimension at observation level:

```
<message:DataSet action="Information" ss:dataScope="DataStructure"
    xsi:type="ns1:DataSetType" ss:structureRef="TN1_DSD_EDUPUB_1_0">
    <Series FREQ="A" IND="NSTUD" CYCLE="2CSEC" CLASS="_Z" TIME_PERIOD
        ="1999">
        <Obs SEX="F" OBS_VALUE="464331" />
        <Obs SEX="M" OBS_VALUE="443917" />
        <Obs SEX="T" OBS_VALUE="908248" />
    </Series>
    <Series FREQ="A" IND="NSTUD" CYCLE="2CSEC" CLASS="_Z" TIME_PERIOD
        ="2000">
        <Obs SEX="F" OBS_VALUE="493783" />
        <Obs SEX="M" OBS_VALUE="469202" />
        <Obs SEX="T" OBS_VALUE="962985" />
    </Series>
```

Once the user has completed the mandatory fields the "Download" button is activated and he can download the dataflow.

Delete

The "Delete" button allows the user to delete a Dataflow. The System shows a pop-up mask where the user can confirm or not the operation.

8.3 How to update an existing Dataflow

From the list of created Dataflows select the Dataflow to be modified.
From the functions displayed in the icons on the right choose "View/Edit" which, in addition to displaying it, allows the user to edit the Dataflow.
The System shows all the elements of the Dataflow:

- *General Data*
- *Query*
- *Categorisation*
- *Header*
- *Layout annotations*

in windows identical to those seen in the creation phase of the Dataflow.

If the Dataflow is not in production and if the user has the necessary permissions he can use the windows to make changes to the Dataflow.

Some information, such as the Dataflow identifier, cannot be modified, a prohibition symbol appears on these and they cannot be edited.

The user can modify the URI, validity, name and description in the General Data mask.
In the mask "Query" it is possible to change the selection of dimensions and attributes and to change the set filters.
In the mask "Categorisation" the user can change the category to which the Dataflow is associated.
In the "Header" mask it is possible to modify the Dataflow header by clicking on the "Dataflow with header" flag.
The annotations previously set for the Dataflow can be modified in the "Layout annotations" mask.

For the details of the single operations see the *Dataflow builder*.

With the "Save" button all changes made to the Dataflow are saved.

8.4 How to add data to an existing Dataflow

The ability to add data to an existing Dataflow results in loading the data into the Cube from which the Dataflow is derived.

In the section: *How to load a new Dataflow* we have listed the actions necessary to create a Dataflow having only the DSD available:

- **Builder**: creation of a Cube associated with a DSD
- **File Mapping**: creation of a mapping between Cube and data
- **Loader**: data loading
- **Dataflow Builder**: Dataflow creation and publication

To add data to an existing Dataflow only the third step is needed: *Loader* which allows the loading of data from CSV or SDMX-ML files.
The loading of a new file in the Cube will affect all Dataflow derived from it.

In the case of published Dataflows in which "Transcoding" or "Content Constraint" are present, the update is not immediately visible as these functions lock the dimensions values to the codes present at the time of publication.
In order to acknowledge the presence of new values deriving from the data update, the "Regenerate CC+Transcoding" option must be selected, which allows the loading of data by recreating also the Transcoding and the Content Constraint.
For each operation's details see section: *Loader*.

8.5 How to create different Dataflows from the same Cube

The creation of different Dataflows from the same cube must be seen as the definition of different views from the same data.

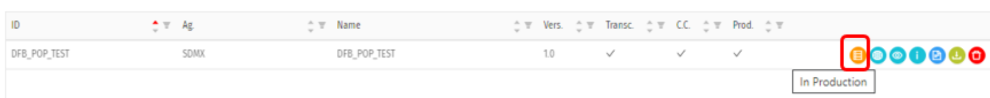
In the section: *How to load a new Dataflow* we have listed the actions necessary to create a Dataflow having only the DSD available:

- **Builder**: creation of a Cube associated with a DSD
- **File Mapping**: creation of a mapping between Cube and data
- **Loader**: data loading
- **Dataflow Builder**: Dataflow creation and publication

To create a new Dataflow from the same Cube only the last step is needed: *Dataflow Builder* which allows the the creation of a Dataflow from a Cube present in the tree of the Category Schemes.
For the details of each operation see the Dataflow construction in the section: *Dataflow Builder*.

8.6 How to put a Dataflow into production

Once created, the Dataflow must be published so that it can be available. This activity can be done in the *Dataflow Builder* section by selecting the Dataflow and clicking on the "Publish" icon.



The production set-up is done in 6 sequential steps:

- **Mapping Set**
- **Transcoding** (not mandatory)
- **Content Constraint** (not mandatory)
- **Production**
- **CC from Partial Codelists** (not mandatory)
- **Data Browser cache** (not mandatory)

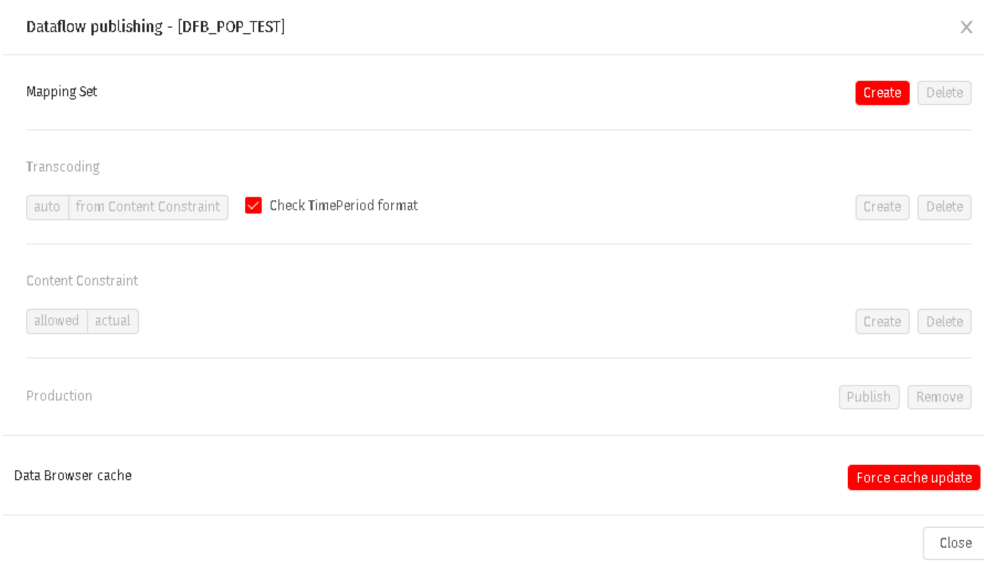
If the node settings include the configuration to connect to the Data Browser, the option to clear the dataflow cache will also be present in the publication window.

The completion of a step allows the passage to the next step.

The same mask allows the user to remove a Dataflow from production or to delete the settings set in the steps above with the "Delete" or "Remove" buttons.

Mapping set

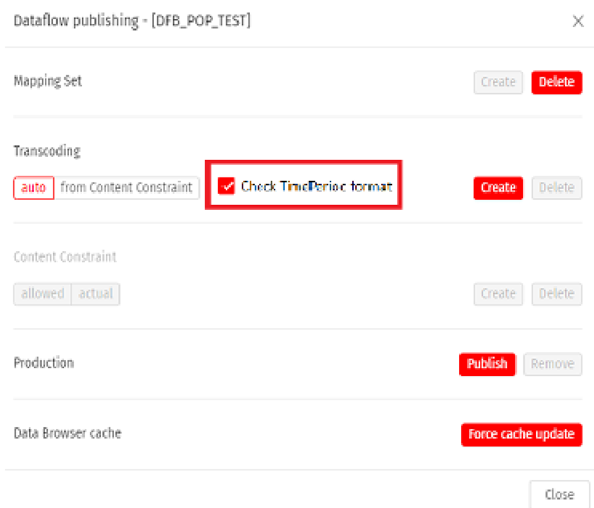
Pressing the "Create" button, the system opens a pop-up window where the user enters the "Default value" to be used for observations with a null value.



By clicking on "Ok" the system performs the mapping between the information present in the data and the elements of the DSD.
The mapping is done automatically based on the information provided in the section *File Mapping* during the construction of the Dataflow, in this step the conformity checks are carried out between the values contained in the dimensions and the corresponding Codelist items.

Transcoding

In this step the correspondence between the codes present in the data and the codes present in the Codelist is created.
For the temporal dimension, for which there is no Codelist, a format Transcoding is applied (e.g. year-month => yyyy-mm).
The Transcoding can be done automatically based on the information contained in the definition of the Dataflow or from an already existing Content Constraint linked to the Dataflow.



The user has the option of doing a consistency check (optional) between frequency and time period, so if the time period format does not conform to the frequency, a blocking message will be triggered.

Content Constraint

In this step the Content Constraint is created automatically according to the information contained in the Dataflow definition.

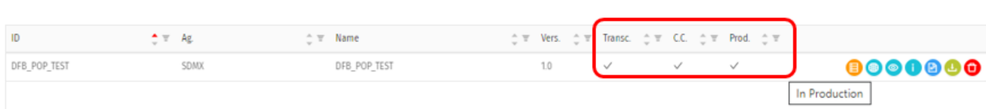
The user will have the possibility to select the content contstraint type that the system is going to create. The choice falls between:

- allowed (default)
- actual

Production

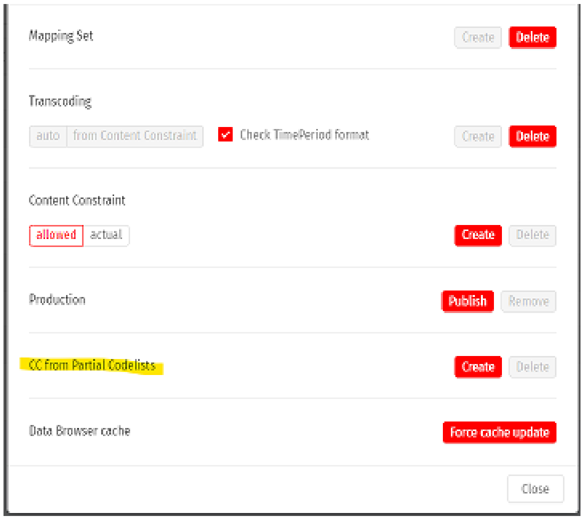
By clicking the "Publish" button the user puts the data into production; the button is active as soon as the first step "Mapping set" is completed, as publication is possible even without the "Mapping set". The button is active as soon as the first step "Mapping set" is completed, since the publication is possible even without the creation of Transcoding and Content Constraint.

To be in production, the presence of Transcoding and Content Constraint are indicated by the flags highlighted in the figure.

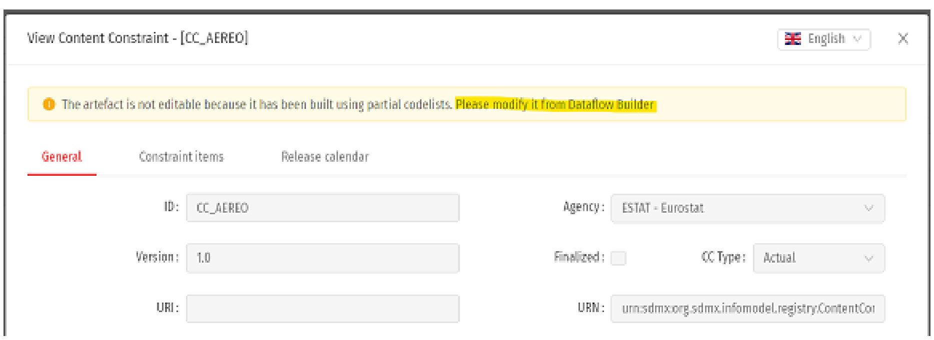


CC from Partial Codelists

This functionality allows the creation of content constraints (CC) from a dataflow created with the Dataflow Builder.



It allows the user to enter all general information about the dataflow as is currently done via Meta Manager (General tab) and he will be able to specify whether the CC should be actual or allowed.
The user will select for each dimension or coded attribute any codes to be included in the CC from the partial codelists of the dataflow itself.
The new CC will be present in the list of all CC in the Meta Manager section, but it will not be editable. Updates must be made from the Dataflow Builder functionality.



After creating a CC from Partial Codelist, the user will not be able to create CCs automatically (because the modes are alternate).

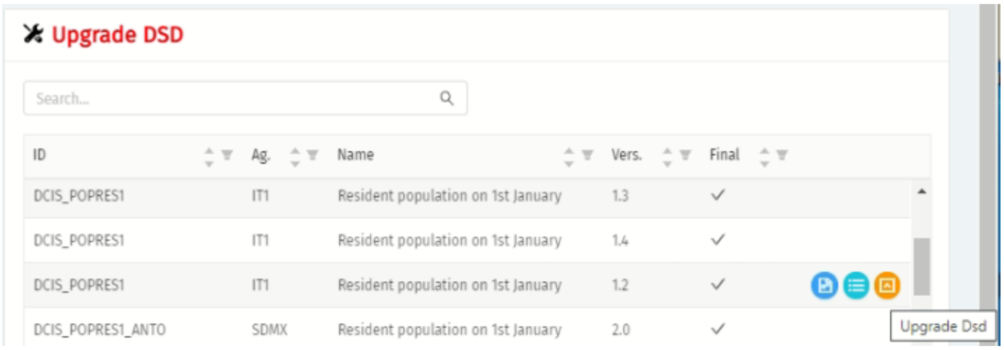
Data Browser cache

If the Data Browser section is properly set in configuration, through this button it is possible to invalidate the data cache of the Data Browser for the selected dataflow. In order to regenerate the cache it is sufficient to access the data in the Data Browser, for example by clicking on the button with the world icon (present only if the dataflow is in production) in the main screen of the Dataflow Builder. This operation is performed automatically in the following cases, which also produce the update of the LAST_UPDATE annotation of the related dataflow:

- Putting in/removing a dataflow from production
- Loading data into a cube (CSV or SDMXML format)
- Loading an attribute file
- Deleting a series
- Empting a cube

8.7 How to upgrade a DSD

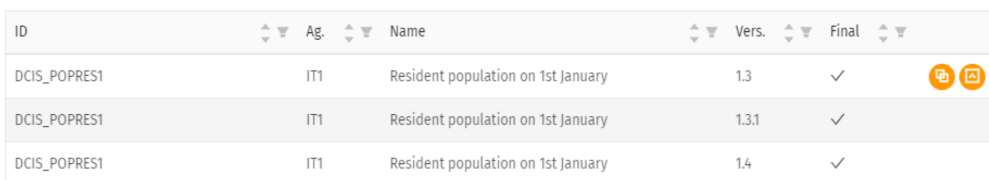
The "Upgrade DSD" function can be found in the menu on the left under "Data Manager" and allows the user to update the version of a DSD used in a Cube.
By selecting this item from the menu, it will show all the DSDs with at least one associated Cube.



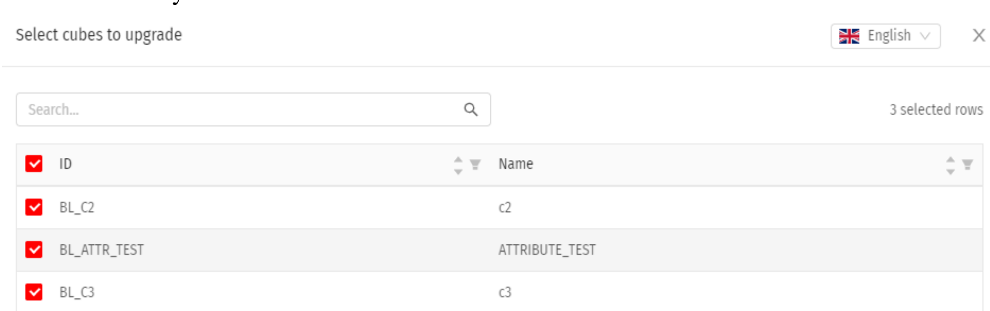
At line level on the right, icons are present for:

- *View the DSD*
- *View the associated Dataflows*
- *Upgrade the DSD*

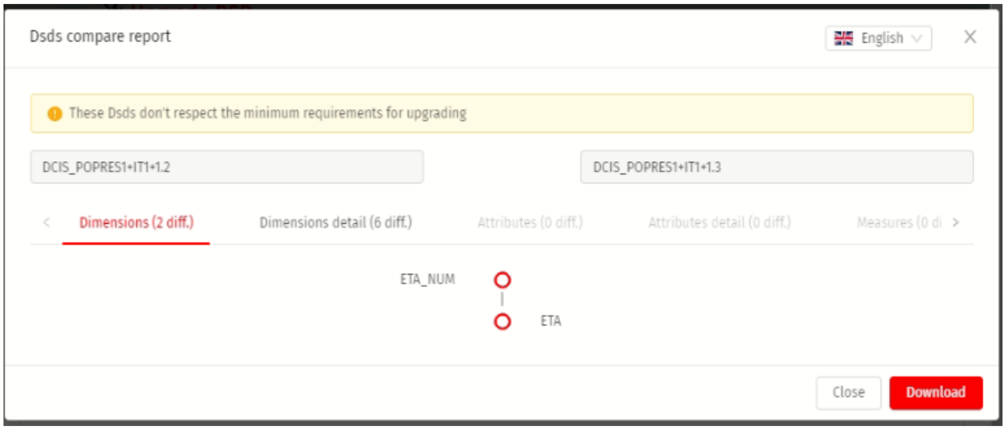
By clicking on the "Upgrade DSD" icon the System opens a window where the user can select the target DSD needed for the upgrade. On the line of the DSD the user wants to select, there are two buttons which allow comparison between this DSD and the orginal one, and the actual selection of the cubes the user is intended to upgrade.



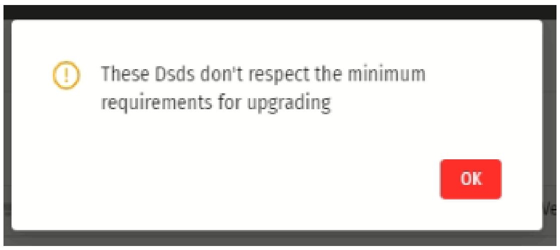
Once this last button is pressed, the cubes to be updated (among those that use this DSD) will be all selected by default.



If the user selects the comparison button, the system shows a report with the results of the comparison between the two DSDs. This comparison can be downloaded in a report with the "Download" button.



The "Update DSD" icon updates the DSD in all connected objects, e.g. if a Cube was created with the DSD: DCISPOP-IT1-1.2, after updating the DSD to version 1.3 the Cube will be associated with DSD: DCISPOP-IT1-1.3.
If the DSD chosen for the update does not meet the minimum requirements (e.g. there is no mandatory dimension or attribute) the update is blocked with an error message.



Using the Upgrade DSD functionality forces the re-creation of the dataflows for which the cube was upgraded due to the dsd upgrade.
Existing categorizations in the categorization schema themes are also recreated exactly as they were when the upgrade was initiated.

If categorization creation fails, the entire dataflow will still appear among those successfully upgraded but a message will be shown in the report indicating that categorization creation has failed.

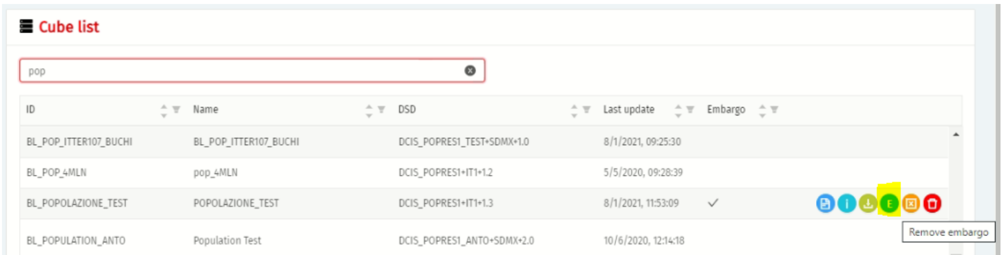
8.8 Other use cases

This section explains the other functions available in the menu under "Data Manager".

- **Cube Lists**
- **Manage Series**
- **Synchronise code lists**
- **Import attribute files**
- **Update Data Browser cache**
- **Update Databrowser DSD Cache**
- **DDB reset**
- **Remove temporary tables**

Cube lists

The "Cube List" menu item displays the list of cubes in the system.
The "Search" function can be activated from the mask to display one or more cubes that contain the entered text.

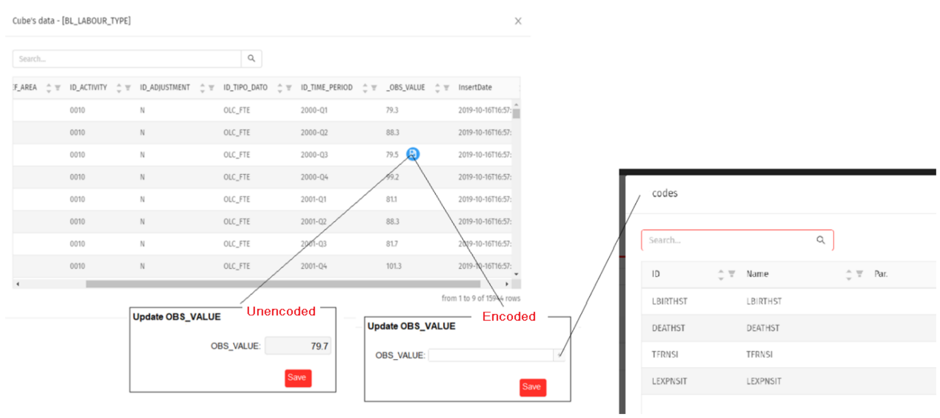


The list of Cubes has a tabular representation with the following columns:

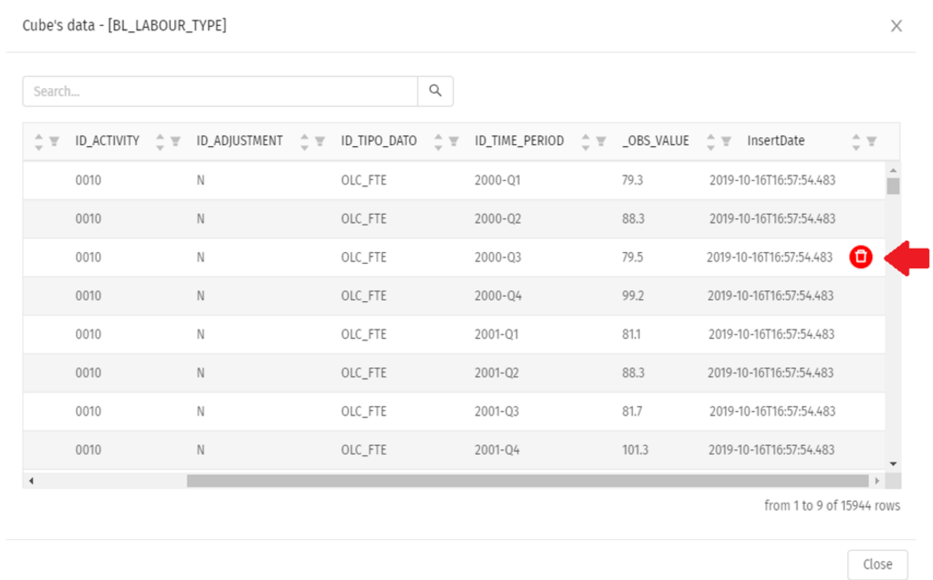
- *ID representing the identifier of the Cube.*
- *Cube name*
- *Name of the Reference DSD*
- *Last modification of the Cube*
- *Name of the Cube data view*
- *Flag embargo*

Functions provided for each item in the list are:

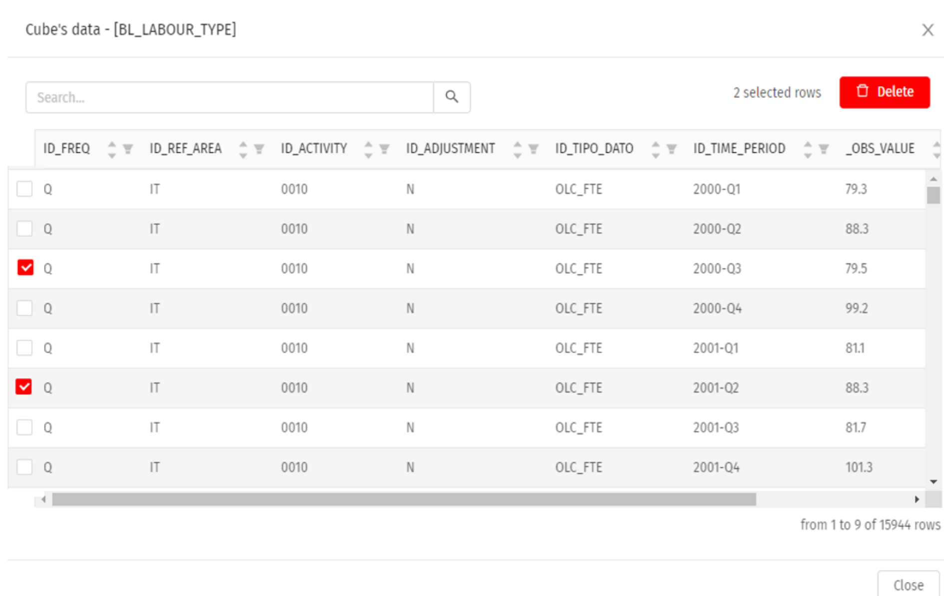
- *Preview/Edit Cube data:* allows the user to view data, to change the OBS_VALUE of a record (encoded or unencoded):



to delete a single record:



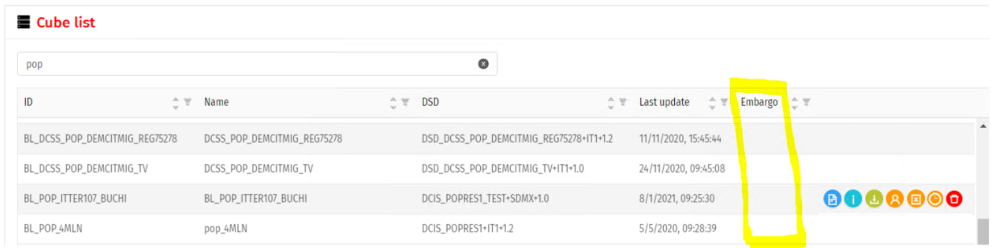
or to delete a list of records however punctually selected by the user:



- *Download:* allows the user to save the contents of the Cube to a CSV file, you can select only some dimensions or attributes and submit filters to select the data to export.
- *Remove Embargo:* allows the user to remove the embargo.
- *Manage owner:* allows the user to manage the rights on the Cube.
- *Empty Cube:* deletes the data contained in the Cube.
- *Set Lastdate:* allows the user to set the last updated date of the cube and all associated dataflows to today.
- *Delete:* deletes the Cube.

it is possible

For cubes that have embargoed data, the "Remove Embargo" function is available, which allows the manual release of the data that was loaded in the *Loader* section but not visible.
Once the embargo is removed, the data is visible and the flag and icon that indicated the embargo are no longer present.



A user who owns a Cube can assign the rights on the Cube to another user operating on the "Manage owner" function.
The System opens a mask allowing the user who owns the Cube to choose the user to whom he wishes to assign rights.

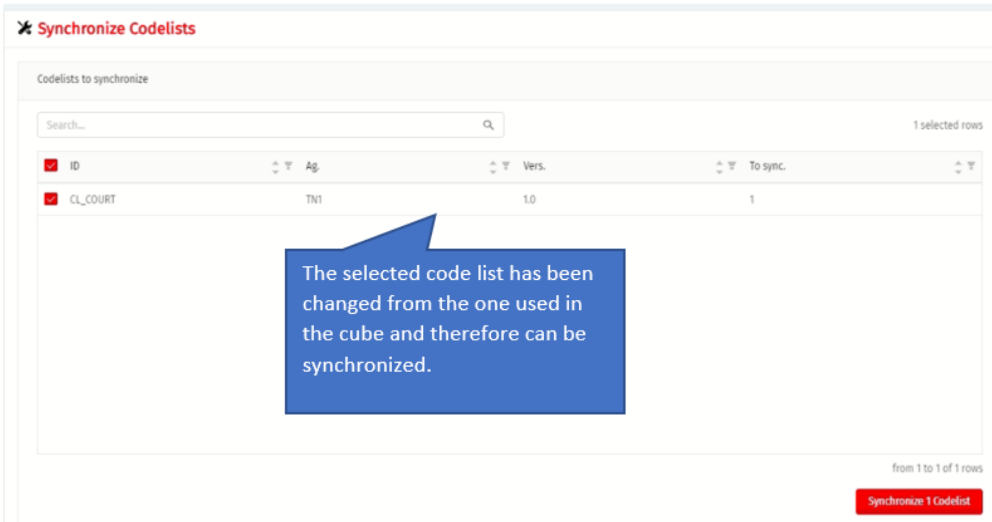
By clicking on the "Save" button, the selected user acquires rights on the Cube.

Manage Series

The menu item "Manage series" is used to edit several series at the same time. Once the cube has been selected from the category tree it is possible to see all the series (combinations of dimensions). It is possible to scroll through the complete list and make a choice of the individual series to be deleted. With both simple and advanced filters it is possible to identify exactly the series and eventually delete them.

Synchronise code lists.

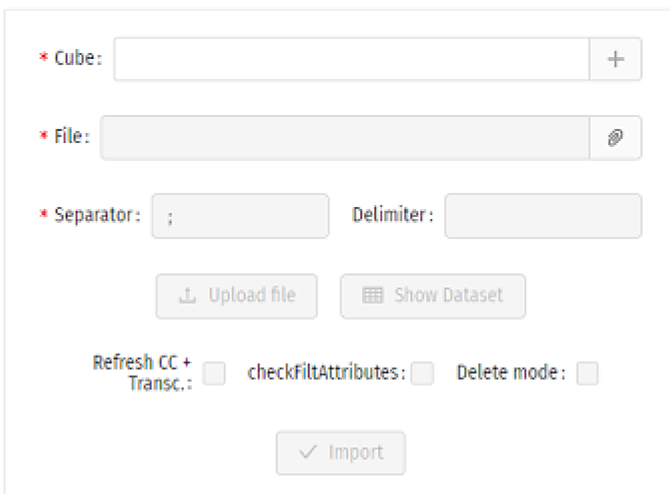
The menu item "Synchronise Code Lists" opens a window where the "Code Lists" that can be synchronised are displayed, e.g. Code Lists used in a Cube and that have been modified by adding an item.



With synchronisation, the structures created during the construction of the Cube are aligned with the changes in the code list, so that the loading of a new data element containing the new items can take place successfully.

Imports attribute files

The "Import Attribute File" menu item allows the user to import or overwrite the attributes of a cube using a list of rules defined in a csv file.



The user must define

- the cube into which import the attributes (mandatory)
- the file from which import the attributes (mandatory)
- the separator used within the file (mandatory)
- the delimiter used within the file (optional)

Once the file has been loaded (it will be possible to see a preview by clicking on the "Show dataset" button), the user will be able to decide whether to actually perform the import by specifying:

- to regenerate or not content constraint and transcoding for the dataflows derived from the cube on which the attributes are being loaded
- to check or not the correctness of the attributes defined on the File (i.e. those at Group and Dimension Group level). If the check fails, the attributes will not be loaded and the initial situation will be restored.

The file must have the following format:

- the header shall contain the identifiers of all the dimensions and attributes present in the cube
- for each line we specify for a given combination of dimension values (where we can leave empty the positions corresponding to dimensions for which the rule must be applied for all values) the value to be assigned to one and only one of the attributes present
- the rules are applied sequentially from the first to the last line so that the subsequent ones overwrite the previous ones. Rules for more specific cases than a more general case will therefore have to be contained in subsequent lines.

Here an example of a csv file:

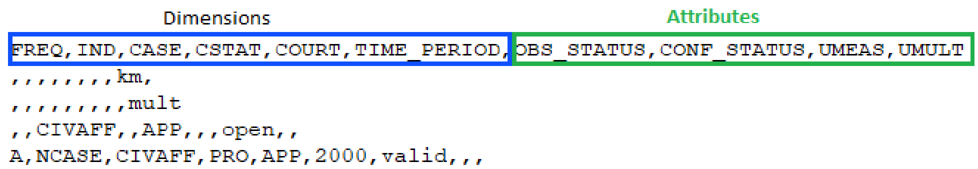
Dimensions Attributes

```
FREQ,IND,CASE,CSTAT,COURT,TIME_PERIOD,OBS_STATUS,CONF_STATUS,UMEAS,UMULT
,,,,,,,,km,
,,,,,,,,,mult
,,CIVAFF,,APP,,,open,,
A,NCASE,CIVAFF,PRO,APP,2000,valid,,,
```

- Row 1: header with dimensions and attributes
- Row 2: the UMEAS attribute has the value "km" for any combination of dimensions (attribute with attachment level Dataset)
- Row 3: the attribute UMULT has the value "mult" for any combination of dimensions (attribute with attachment level Dataset)
- Row 4: the CONF_STATUS attribute has the value "open" for all data points where the CASE dimension has value IVAFF and the COURT dimension has value APP (attribute with attachment level Dimension Group)
- Row 5: the attribute OBS_STATUS has the value "valid" for the given values of each dimension (attribute with attachment level Observation).

Obviously it would have been possible to indicate different values of the last two attributes for different combinations of dimension values.

This functionality also allows deletion of conditional attributes already inserted in the cube. This can be done by checking the "DeleteMode" checkbox in the mask, and by uploading a CSV file, as previuosly explained for the import, containing NULL values for the attributes the user wants to delete. The operation works only on conditional attributes of course, because mandatory attributes can never be set to NULL by definition. The following example shows how the CSV file looks like if the user selects deletion mode:




```
FREQ,IND,CASE,CSTAT,COURT,TIME_PERIOD,OBS_STATUS,CONF_STATUS,UMEAS,UMULT
,,,,,,,,km,
,,,,,,,,,mult
,,CIVAFF,,APP,,,open,,
A,NCASE,CIVAFF,PRO,APP,2000,valid,,,
```

- Row 1: header with dimensions and attributes
- Row 2: the UMEAS attribute is set to NULL for each combination of dimensions (attribute with attachment level Dataset)
- Row 3: the attribute UMULT is set to NULL for each combination of dimensions (attribute with attachment level Dataset)
- Row 4: the CONF_STATUS attribute is set to NULL for each datapoint in which the dimension CASE case has value CIVAFF and the dimension COURT has value APP (attribute with attachment level Dimension Group)
- Row 5: the attribute OBS_STATUS is set to NULL for values indicated in each dimension (attribute with attachment level Observation)

Update Data Browser cache

Using the "Update Data Browser cache" menu item it is possible to invalidate the catalogue cache of a Data Browser node appropriately defined in the configuration section of an MDM node. This functionality is performed automatically in the following cases:

- Putting in/removing a dataflow from production
- Creating a new dataflow from Data Manager
- Deleting a dataflow from the Data Manager
- DDB reset

Update Data Browser DSD Cache

Using the "Update Data Browser DSD Cache" menu item it is possible to invalidate cache regarding DSD of a dataflow with the conceptscheme and codelists referenced therein linked to data included in a Data Browser node appropriately defined in the configuration section of an MDM node. The functionality of clearing a DSD cache will only occur at the request of the user and never automatically.

DDB reset

The menu item "DDB reset" resets all the operations done in the "Data Manager".
A pop-up window is shown to confirm the removal.

Remove temporary tables

The "Remove temporary tables" menu item removes the temporary tables created by the system during operations on Cubes or Dataflows (e.g. support tables for loading).
A pop-up window will be displayed for confirmation by the user.

CHAPTER

NINE

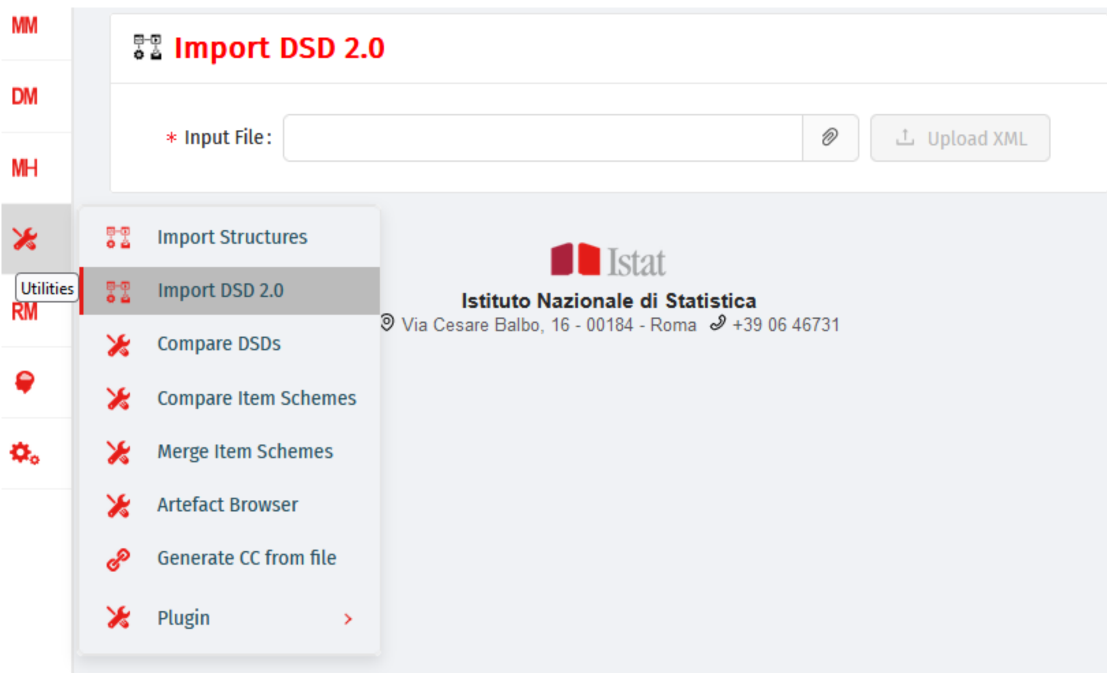
UTILITIES

9.1 Import structures

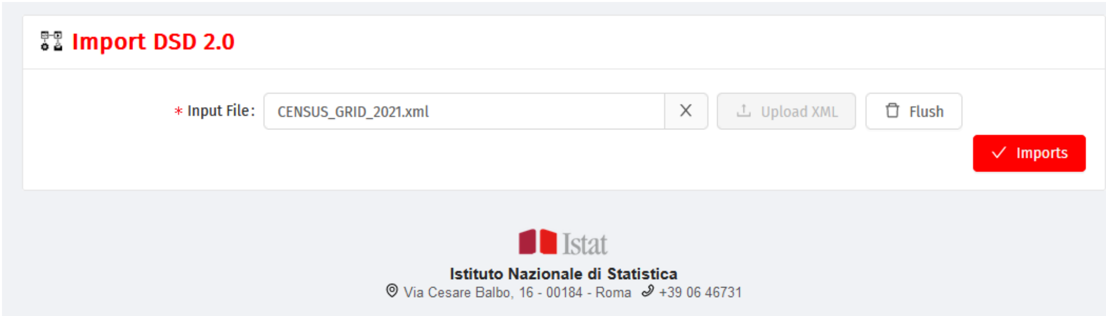
The **Import Structures** function is present in the left-hand side menu starting from the "Tools" item and also in the list of artefacts. For the functionalities see the paragraph: *Import an Artefact* described in the Meta Manager.

9.2 Import DSD 2.0

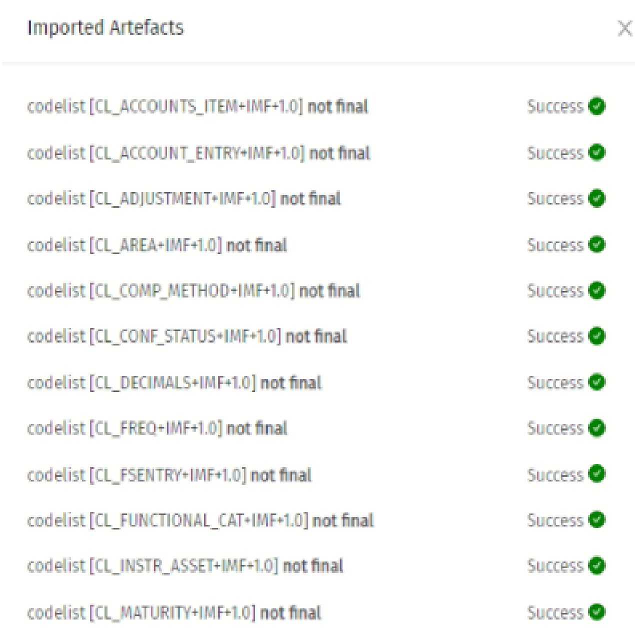
The **Import DSD 2.0** function is present in the left-hand side menu starting from the "Tools" item.



This function allows the user to import structures of SDMX 2.0 type into the system. The user selects a file with data to load from *filesystem* .
During the upload function, activated by the selection of the **Upload XML** button, the System checks the correct format of the file and the consistency of the information.
In case of anomaly a blocking message is displayed to the user that will not allow the file to be uploaded.



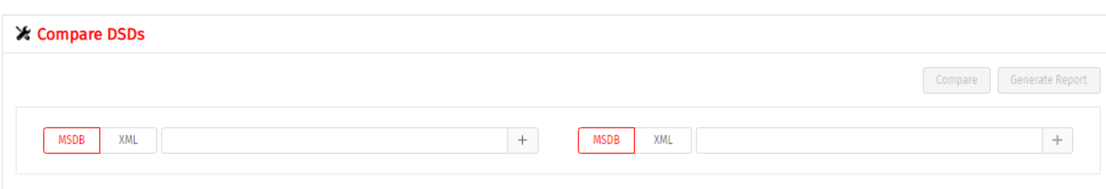
By pressing the import button, import operation is done and the list of imported artefacts is shown.



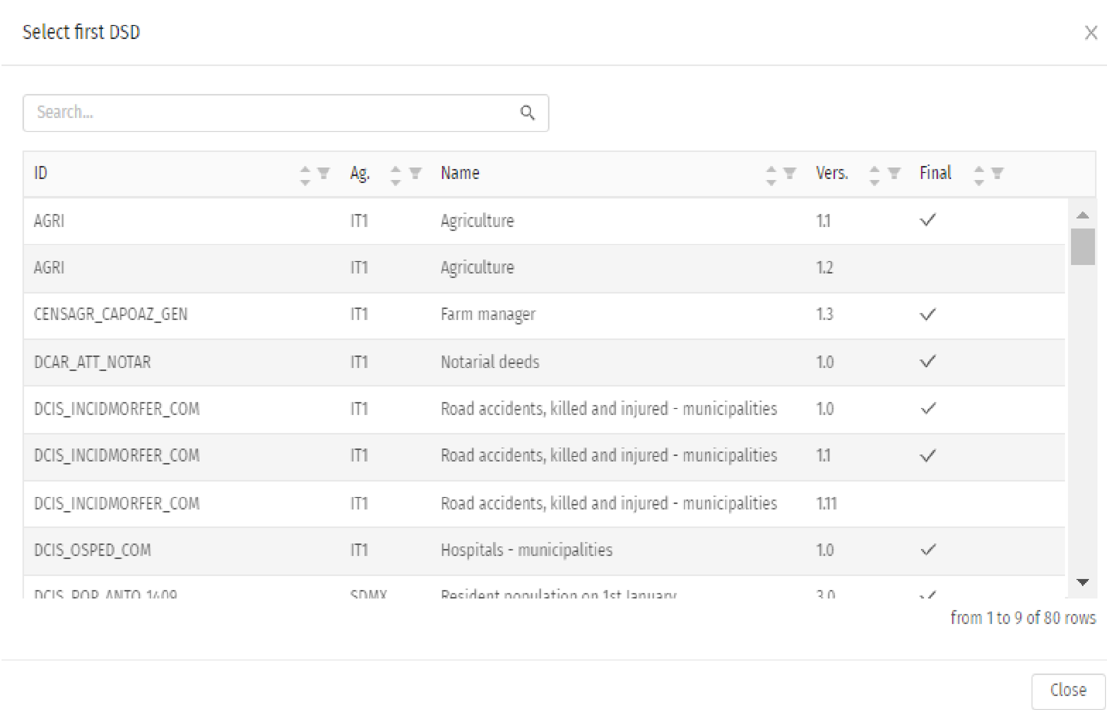
The artefacts already present in the system will not be imported.

9.3 Compare DSDs

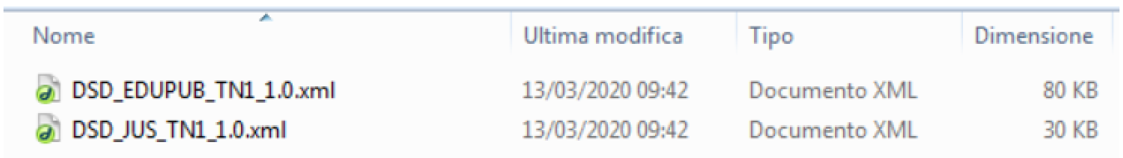
The **"Compare DSD "** function can be found in the left sidebar menu starting from the "Tools" item.
The function allows the user to compare two DSDs highlighting the differences between them.



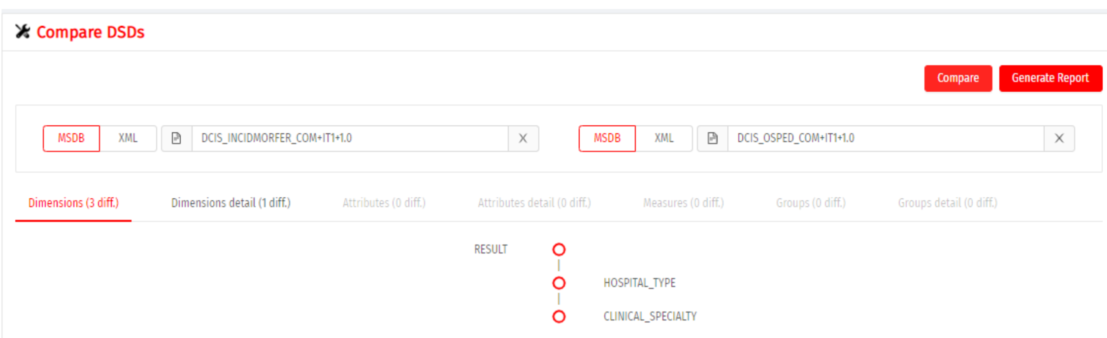
The user can choose, for both DSDs, whether to take the DSD from the Structure DataBase Metadata or to load it from an external file. Obviously, when comparing two DSDs, it is also possible to choose one from the MSDB and the other from an external file to see if there are differences.
If loading is made from the MSDB then the user will be able to choose the DSD from a list of those available in the System:



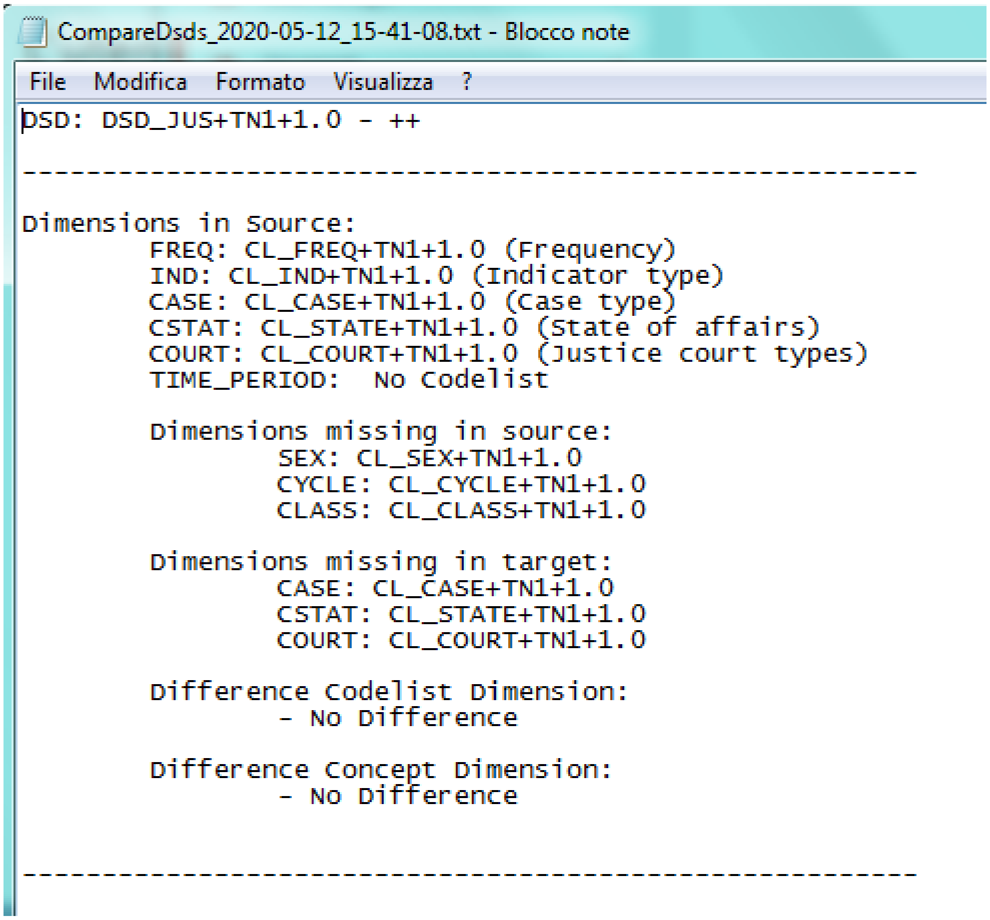
while choosing the XML option, the user selects the file from *filesystem*:



Once the choice of the two DSDs has been made, the System activates the **Compare** and **Generate Report** buttons.
Then the user, by pressing **Compare**, can perform the comparison and obtain the result with the differences on the screen:



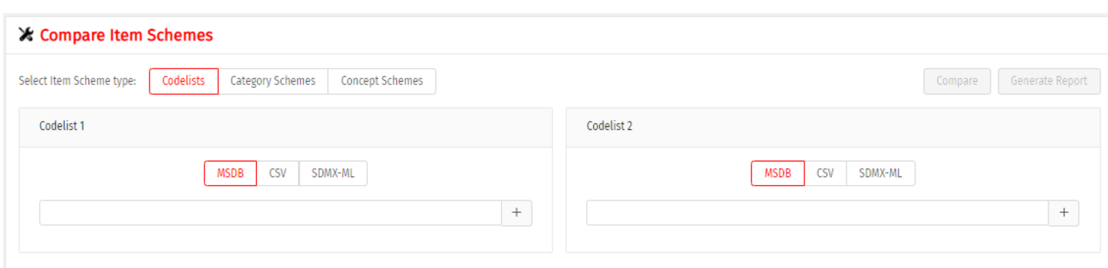
and, by pressing **Generate Report**, he obtains a comparison report in txt format.




```
DSD: DSD_JUS+TN1+1.0 - ++

--------------------------------------------------------

Dimensions in Source:
        FREQ: CL_FREQ+TN1+1.0 (Frequency)
        IND: CL_IND+TN1+1.0 (Indicator type)
        CASE: CL_CASE+TN1+1.0 (Case type)
        CSTAT: CL_STATE+TN1+1.0 (State of affairs)
        COURT: CL_COURT+TN1+1.0 (Justice court types)
        TIME_PERIOD:  No Codelist

        Dimensions missing in source:
                SEX: CL_SEX+TN1+1.0
                CYCLE: CL_CYCLE+TN1+1.0
                CLASS: CL_CLASS+TN1+1.0

        Dimensions missing in target:
                CASE: CL_CASE+TN1+1.0
                CSTAT: CL_STATE+TN1+1.0
                COURT: CL_COURT+TN1+1.0

        Difference Codelist Dimension:
                - No Difference

        Difference Concept Dimension:
                - No Difference


--------------------------------------------------------
```

9.4 Compare Item Schemes

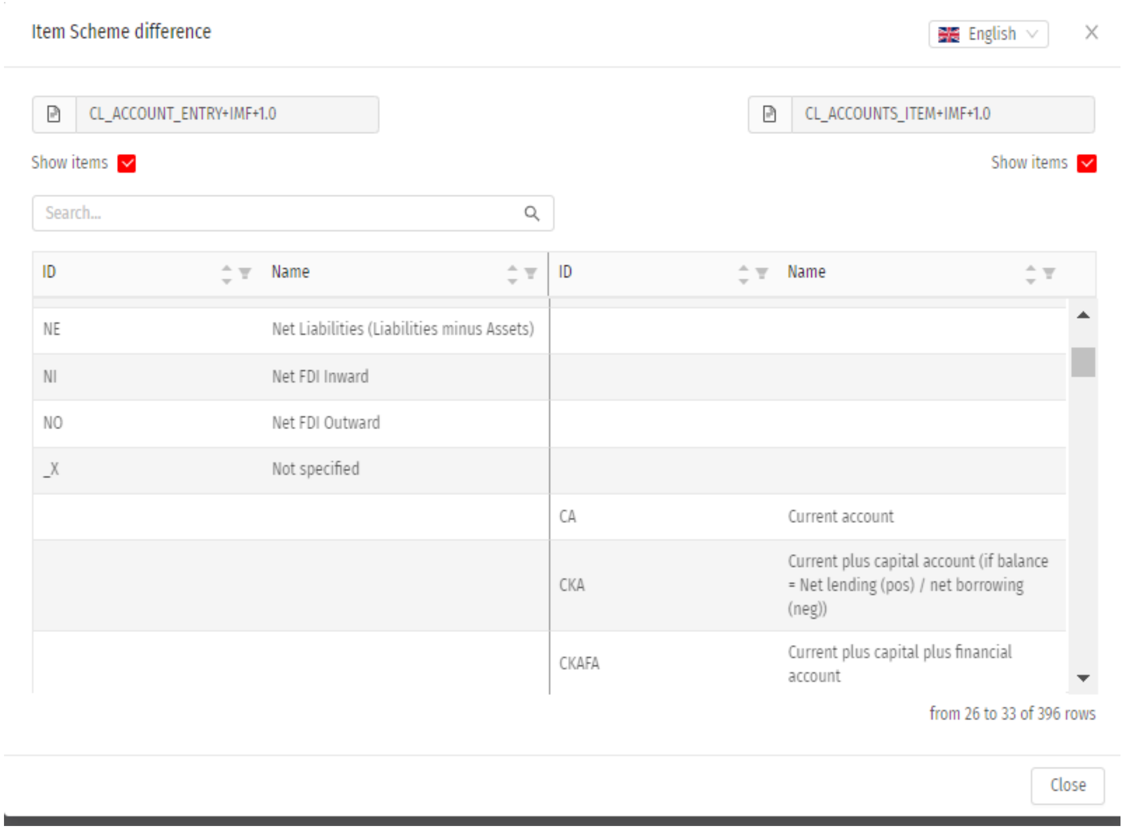
The **Compare Item Scheme** function can be found in the left-hand side menu starting with the "Tools" item.
The function allows a two-by-two comparisons for: **Code Lists**, **Category Schemes** and **Concept Schemes**.



Item schemes can be loaded indifferently from:

- MSDB
- CSV
- SDMX-ML

Once the choice of the two Item Schemes (e.g. two Code Lists) has been made, the System activates the **Compare** and **Generate Report** buttons.
The user, by pressing **Compare**, can perform the comparison obtaining the result with the differences on the screen:

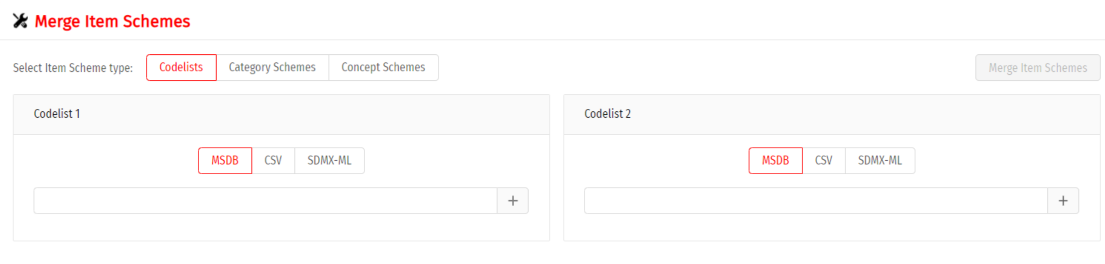


and, by pressing **Generate Report**, he obtains a comparison report in txt format.

```
Artefacts: CL_ACCOUNT_ENTRY+IMF+1.0 - CL_ACCOUNTS_ITEM+IMF+1.0
        Missing item in Source:
                [CA]:  (Current account)
                [CKA]:  (Current plus capital account (if balance = Net lending (pos) / net borrowing (neg)))
                [CKAFA]:  (Current plus capital plus financial account)
                [D1]:  (Primary income, Compensation of employees)
                [D11]:  (Primary income, Wages and salaries excluding employers social contributions)
                [D12]:  (Primary income, Employers social contributions)
                [D121]:  (Primary income, Employers actual social contributions)
                [D1211]:  (Primary income, Employers actual pension contributions)
                [D1212]:  (Primary income, Employers actual non-pension contributions)
                [D122]:  (Primary income, Employers imputed social contributions)
                [D1221]:  (Primary income, Employers imputed pension contributions)
                [D1222]:  (Primary income, Employers imputed non-pension contributions)
                [D2]:  (Primary income, Taxes on production and imports)
```

9.5 Merge Item Schemes

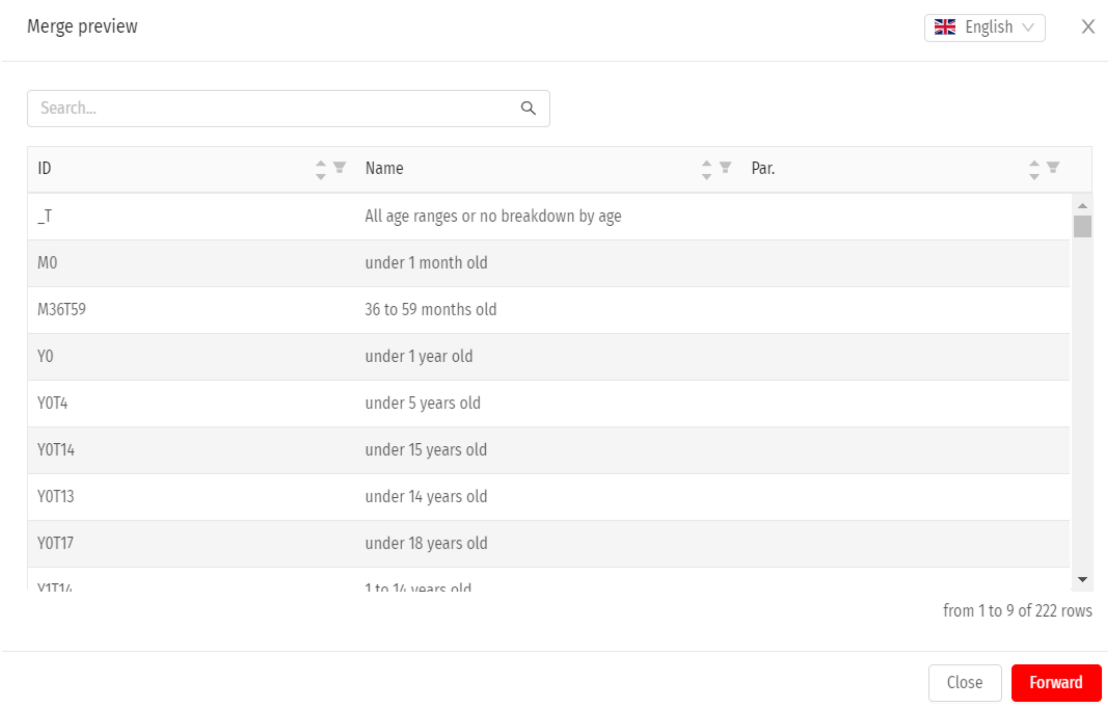
The **Merge Item Schemes** function can be found in the left-hand side menu starting from the "Tools" item. The function allows a two-by-two merge for: **Code Lists**, **Category Schemes** and **Concept Schemes**.



Item schemes can be loaded indifferently from:

- MSDB
- CSV
- SDMX-ML

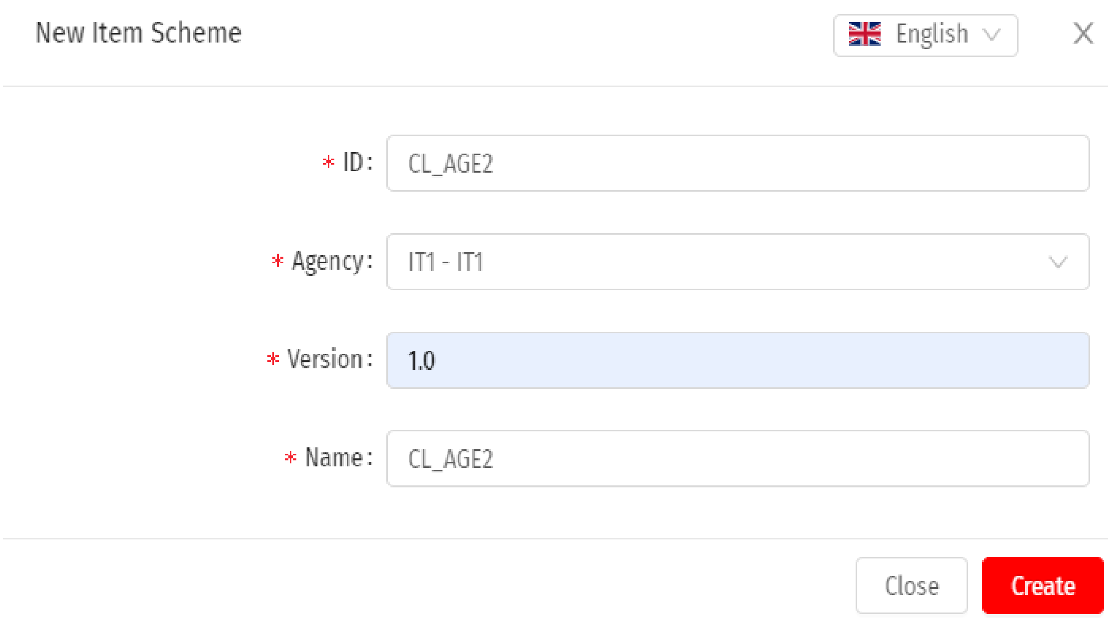
Once the choice of the two Item Schemes (e.g. two Code Lists) has been made, the System activates the **Merge Item Schemes** button.
By pressing **Merge Item Schemes** the Merge preview will be shown:



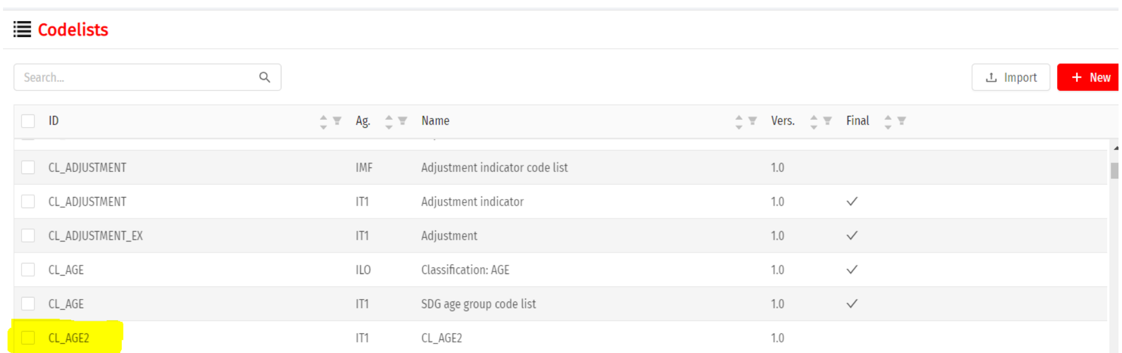
.

By Pressing **Next** the user can create a new Item Scheme filling the mandatory fields:

- ID
- Agency
- Version
- Name



Finally by pressing **Create** the new Item Scheme will be created and the user will be able to locate it in the Meta Manager.

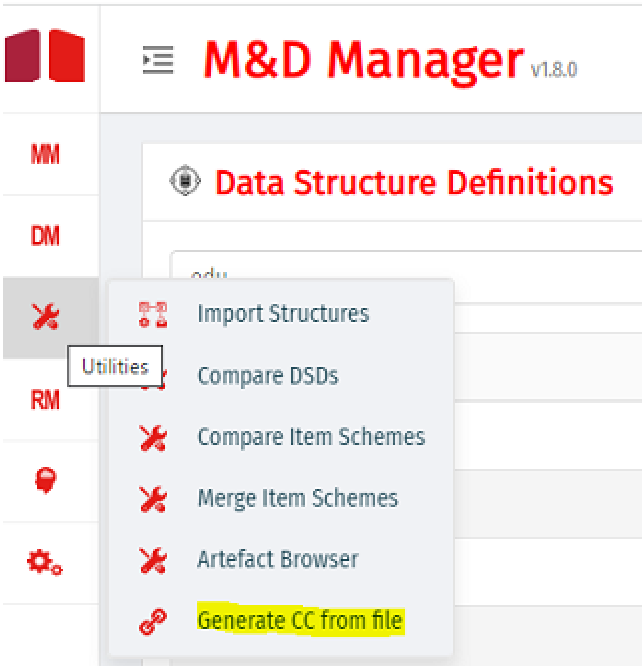


9.6 Browser artefacts

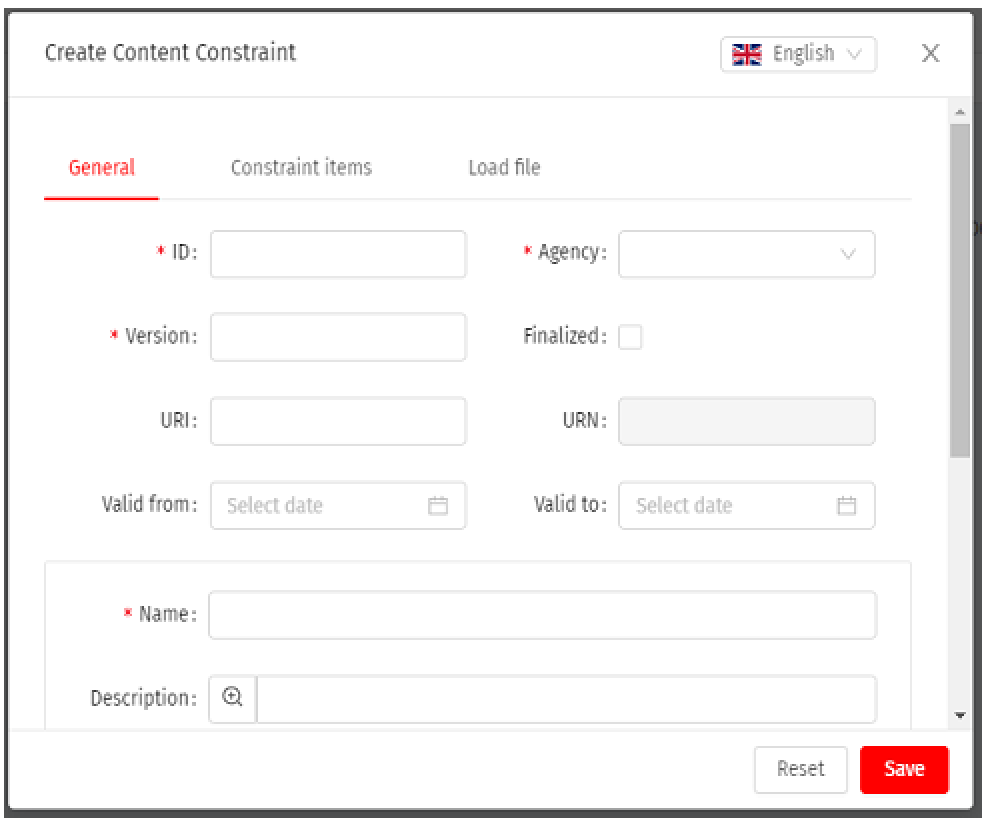
….. coming soon….

9.7 Generate Content Constraint from file

This functionality allows the user to create a content constraint just by uploading a CSV file. The "Generate CC from file" option is selectable from the utility menu



Once the user clicks on the menu, this window appears:

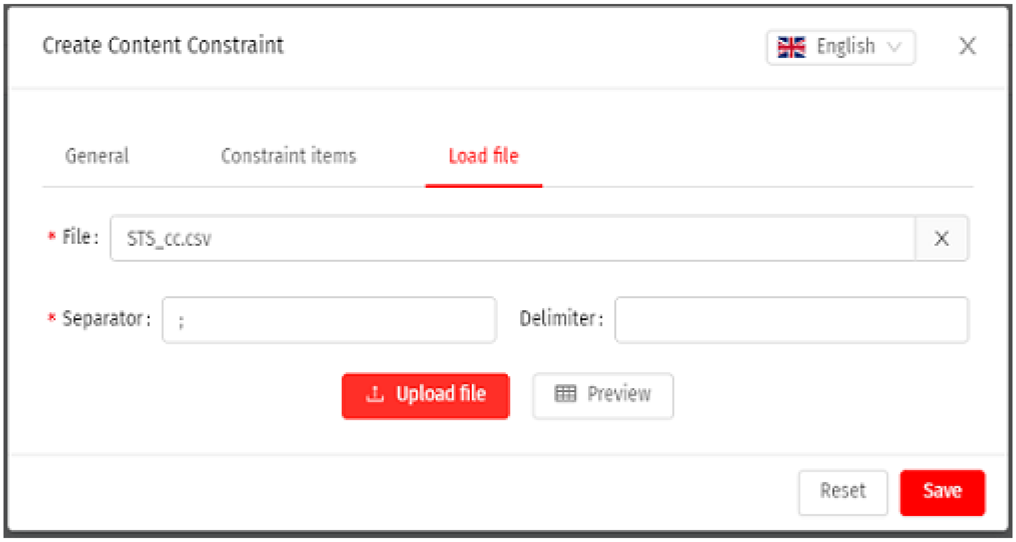


While the first two tabs are the same as the ones seen in the ***Content Constraint* ** paragraph, in the **Load File** tab the system displays the mask for acquiring data from CSV files. The required fields for loading from CSV file are:

- *File* (mandatory)

- *Separator* (mandatory, it is pre-set with ";")
- *Delimiter* (optional)

The user can load the file by clicking on the "Upload file" button and after loading the CSV it is possible to view the data ("Preview" button). Only after completion of the minimum and mandatory information, the System activates the save function using the **Save** button. The new content constraint will then be accessible in the "Content Constraint" section in the Meta Manager menu.



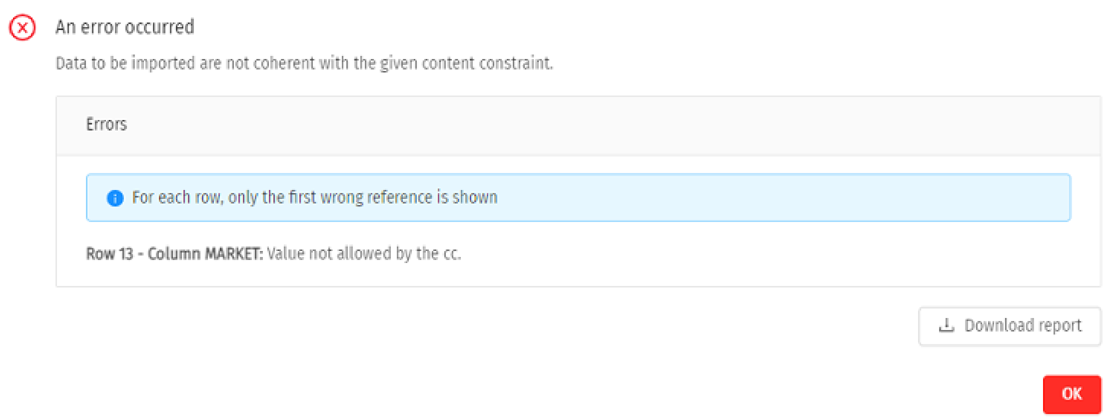
Hereby is a possible example of a CSV file used for CC creation. Considering that neither time_period or non-coded dimensions/attributes are considered while reading a file for CC creation, the user can create a simple csv file containing these columns:

```
FREQ;MARKET;REF_AREA;ADJUSTMENT;INDICATOR;ACTIVITY;BASE_PER;UNIT_MEASURE;SOURCE
A;T;IT;N;TURN;0040;2015;IX;ISTAT
M;D;;;;0050;;;
;E;;;;0080;;;
```

the header must have the same names used as ids in the DSD declaration. Each column indicates the dimension/attribute with the items the user wants to allow in his content constraint. The example specifies that possible items for:

- FREQ are "A" and "M";
- MARKET are "T","D" and "E";
- REF_AREA is just "IT";
- ADJUSTMENT is "N";
- INDICATOR is "TURN";
- ACTIVITY are "0040", "0050" and "0080";
- BASE_PER is "2015";
- UNIT_MEASURE is "IX";
- SOURCE is "ISTAT".

If this content constraint will then be used to validate the upload data operation (during the Loader step), an error will be shown if the uploaded file contains items that differ from the once allowed.
For example, given the rules described earlier, a row containing an item different from T,D or E for the MARKET dimension, will cause an error message like this:

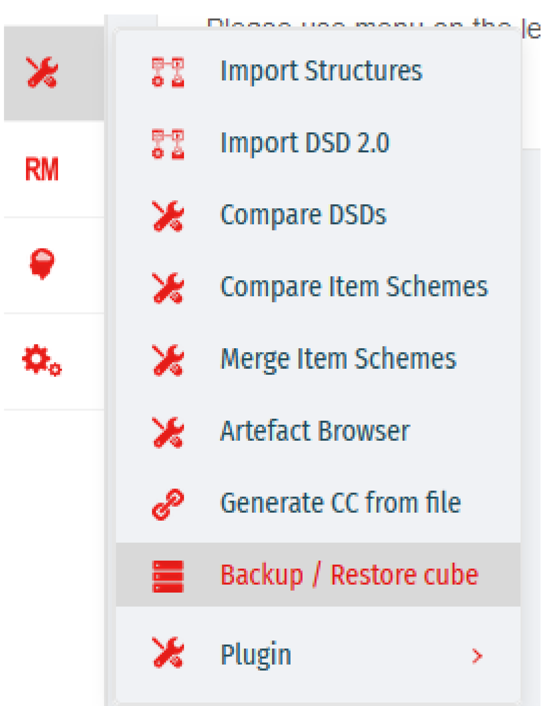


It is also possible to download data from a dataflow and use it as input to generate the content constraint keeping only the information the user wants to use in order to create a CC.

```
MARKET;NOTE;REF_AREA;ADJUSTMENT;INDICATOR;ACTIVITY;BASE_PER;UNIT_MEASURE;SOURCE;TIME_PERIOD
T;;IT;N;TURN;0040;2015;IX;ISTAT;2018-03
D;;IT;N;TURN;0040;2015;;ISTAT;2018-03
E;;IT;N;TURN;;2015;IX;ISTAT;2018-03
T;;;N;TURN;0050;2015;IX;ISTAT;2018-03
T;;IT;N;TURN;0080;2015;IX;ISTAT;2018-03
T;;;N;TURN;0090;2015;;ISTAT;2018-03
```

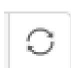
9.8 Backup / Restore cube

The **“Backup / Restore cube”** function can be found in the “Utilities” menu.



This functionality allows the user to backup and restore cube data. Once the user clicks on menu command, this window appears:

The three buttons on the upper side of the window allow the user to

-  backup a cube
-  restore data from backup
-  delete a backup

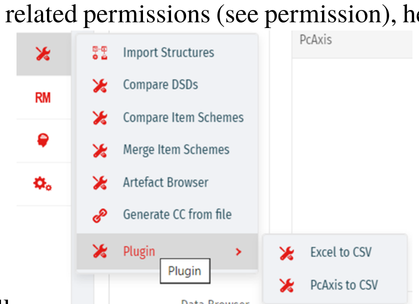
The selection of a cube changes their state:

- if the selected cube does not have a backup, only the "Backup" button is active;
- if it alredy has a backup, the other two buttons will be active, "Restore" and "Delete backup" button.

At the end of the backup procedure, the cube will be marked with a different icon  .

9.9 Plugin

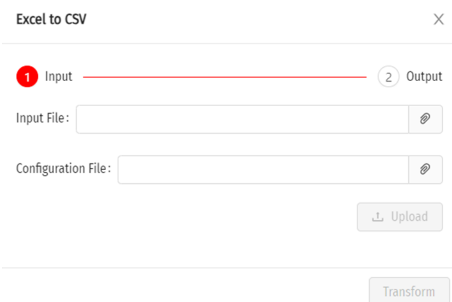
If the online user has the related permissions (see permission), he can view and use the standalone plugins.



plugin_standalone_menu

This functionality allows the user to convert a file from one format to another by choosing the related plugin. In the example, there are 2 installed plugins for the selected node.

Once the user clicks on a plugin, this window appears:

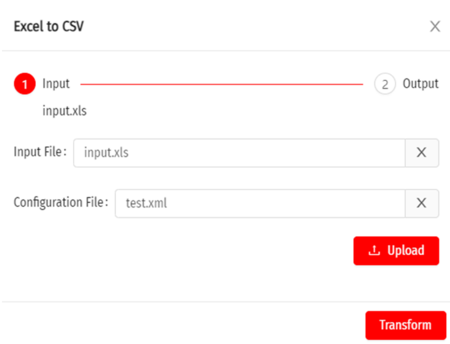


Note : The configuration field will appear if the plugin has the "config file" option checked in his configuration settings.

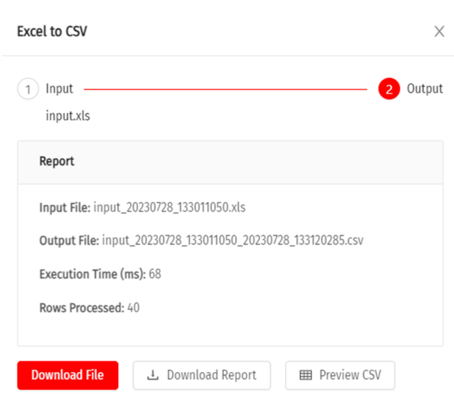
The user has to select one input file to transform, according to file type saved in plugin settings. Otherwise, a warning message will be displayed:

In the same way, the user has to select a configuration file and a process file, if expected by the plugin's settings.

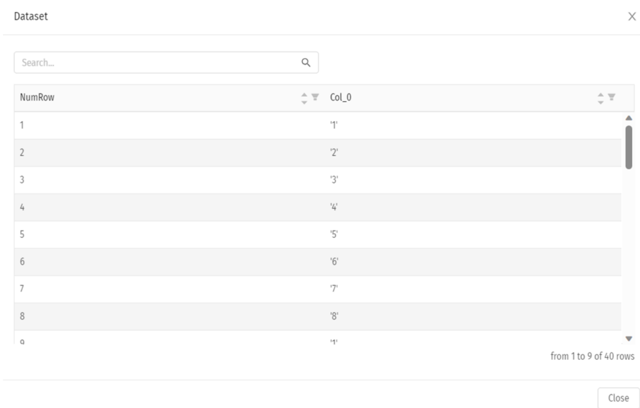
In the form is correctly filled, the Upload button will be enabled. By clicking on Upload, the file(s) will be uploaded and the button Transform will be enabled.



By clicking Transform, the input will be transformed in the expected type and this window appears:



The Download File button allows to download the transformed file.
The Download Report button allows to download the transform report (Input File,Output File,Execution Time (ms),Rows Processed,Series Processed,Error Message) in a CSV file.
The Preview CSV button shows a new window with of the preview of csv data converted (see below):



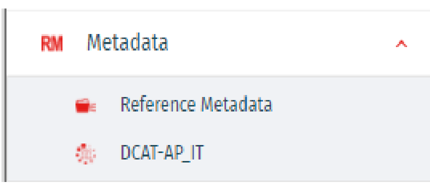
CHAPTER

TEN

METADATA MANAGEMENT

The Metadata item in the application menu has two sections:

- **Referential Metadata**
  Dedicated to the management of Referential Metadata, that is metadata that can be associated to any artefact in a generic way.
- **DCAT-AP_IT**
  Dedicated to the management of Metadata in compliance with the DCAT-AP_IT standard (defined by Agit, the agency for digital Italy), which defines a set of descriptive metadata (mandatory, recommended or optional) used to document both data and catalogues dedicated to public administrations.

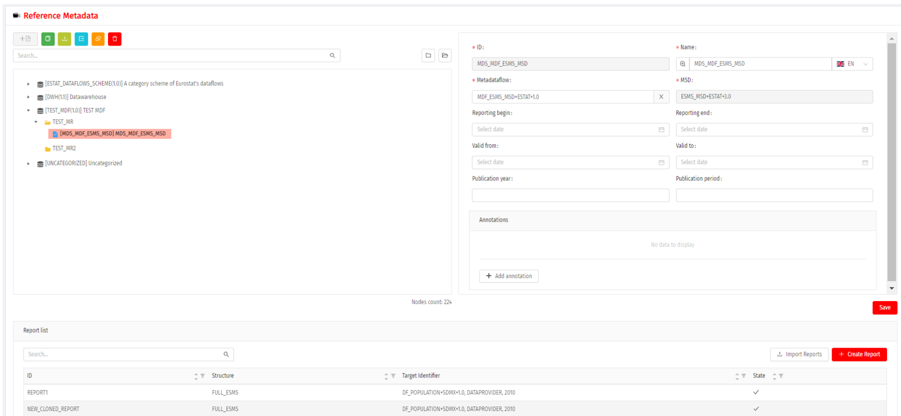


10.1 Referential Metadata

Referential Metadata is linked to Structural Metadata (managed in the MetaManager) and the link is the *Metadata Structure Definition* (MSD) which is a model that specifies the types of constructs to which metadata can be related and the structure of this metadata at the time of dissemination.
In particular from an MSD it is possible to create *MetadataFlows* which represent a metadata flow with the characteristics defined by the MSD and are the starting point in the management of Referential Metadata.

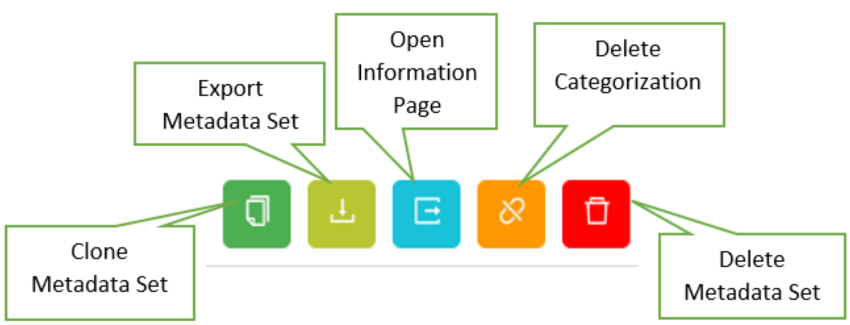
10.1.1 MetadataSet Management

The Referential Metadata module allows the management of Metadata Sets, that is a set of Metadata organised in Category Schemes; each Metadata Set is created from a Metadataflow, with which it shares the categorisation.
The Metadata Sets present will be displayed by browsing the category tree.
Clicking on a single Metadata Set the application will display its details, including the list of associated reports:

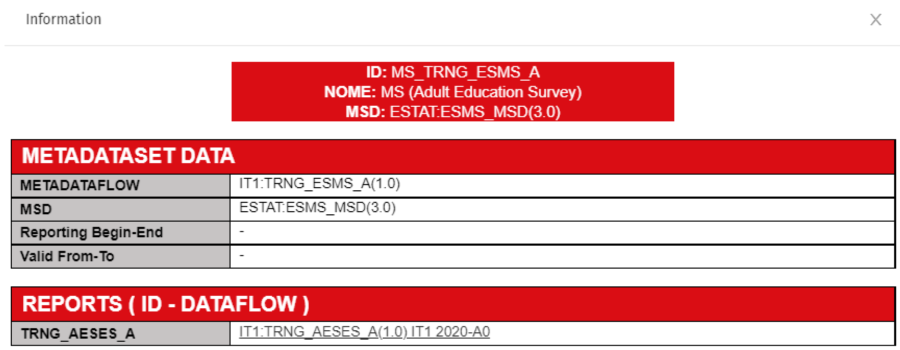


A user has read/write permissions on a Metadata Set and related reports if and only if the user owns the associated Metadataflow.
If a user does not have permissions, he will still be able to see the details of a Metadata Set, but he will not be able to modify them; as regards the list of reports associated with it, the user will only see those exposed via API, again without the possibility of modification.
Therefore, if a user has permissions on the Metadataflow associated with the Metadata Set, he will be able to edit both the Metadata Set's details and create/edit/delete/expose the reports associated with it.

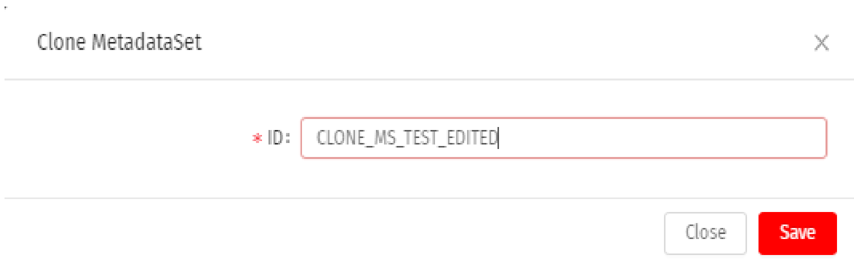
By clicking on a single Metadata Set the following buttons will also be activated by the System:



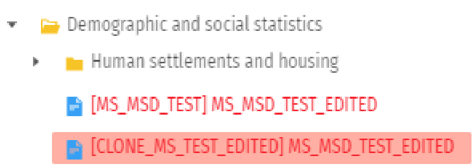
Pressing **Open Information Page** a popup window like the following will be opened:



Pressing **Clone Metadata Set** the user will have to enter a new id to identify the clone:



and the system will create an identical copy of the original Metadata Set with the new identifier:



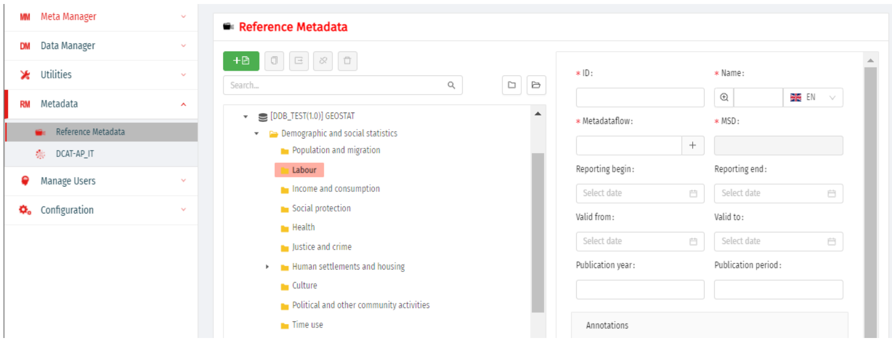
Pressing **Export Metadata Set** the system allows the user to obtain a file in SDMX-JSON format containing the Metadata Set with all its reports.

Pressing **Delete categorization** the system will remove the Metadata Set from a category while the recategorization of a Metadata Set can be done simply by drag&drop operations.

Pressing **Delete Metadata Set** the chosen Metadata Set will be deleted but only if there are no reports linked.

10.1.2 Creating a new MetadataSet

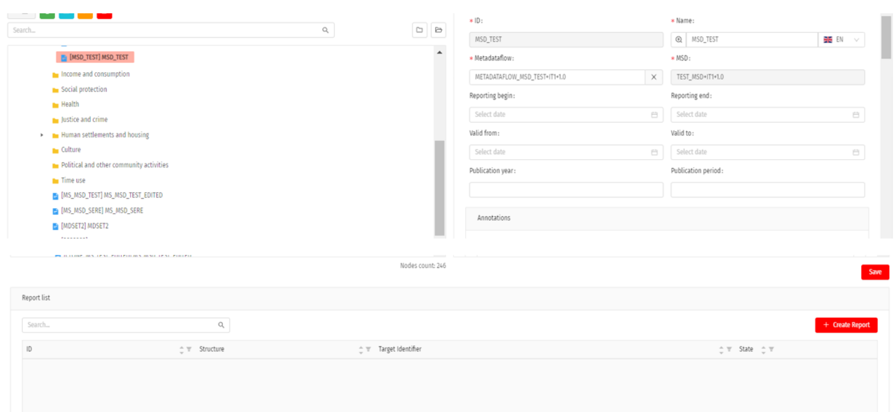
Selecting the category with the mouse, the “Create Metadata Set” button will be enabled, and the user can create a new Metadata Set (after filling in the required fields).



One of the mandatory fields for the creation of a Metadata Set is the linked Metadataflow; the Metadataflow will be selectable from a specific component, in which will be present only the Metadataflow owned by the user.
Other fields are not mandatory such as the validity period of the report and the Annotations that can be inserted with additional structured information even if this element is not an artefact.
When the user has filled in the mandatory fields then the System will enable the “Save” button and the Metadata Set can be saved.

10.1.3 Compilation of a report

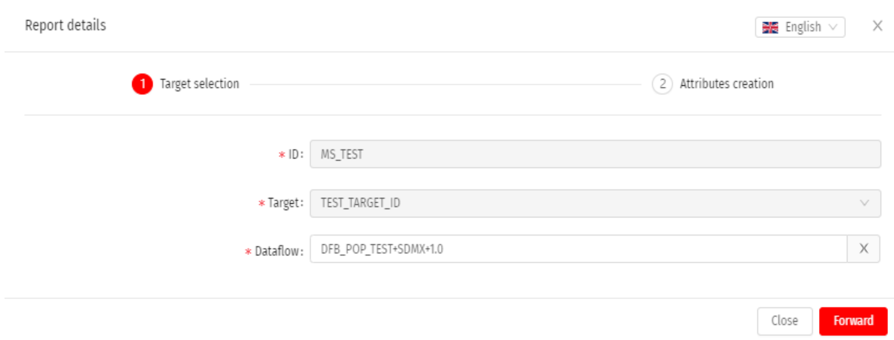
Once the Metadata Set has been created, it can be viewed and one or more reports can be added using the “Create Report” button that the System will have enabled.



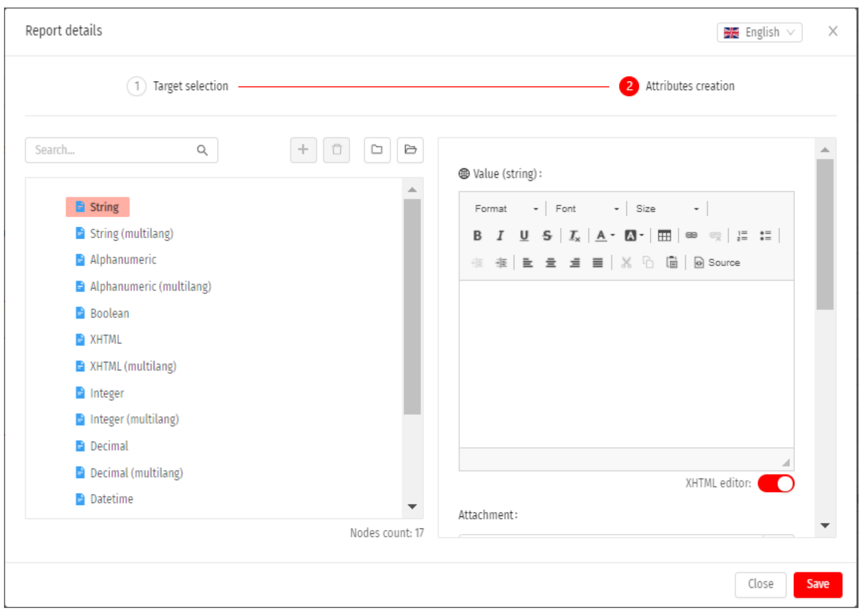
The compilation of the report is done in two steps:

- **Target selection**:
  In the compilation of the report the user must fill in the following fields:
  ID: (mandatory, alphanumeric) freely typed.
  Target: (mandatory, alphanumeric) chosen from a drop-down menu from the list provided by the MetadataFlow associated to the MSD.
  There can be more than one target and according to the choice made, i.e. the target object type, the System will dynamically compose the following requests.
  In the following example the type of target object is the *Dataflow* so the System

asks the user to identify which Dataflow to consider among those present in the node:
DataFlow: (mandatory, alphanumeric) chosen from the drop-down menu.



- **Attribute creation**:
  The 2nd step is the compilation of the attributes coming from the MSD:

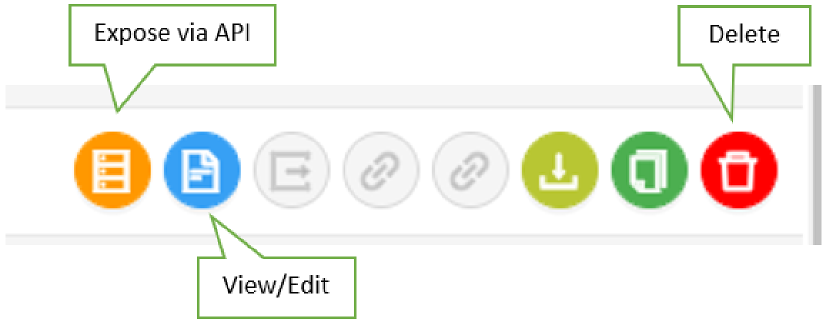


It is possible to save a draft of the report at any time (the system warns the user about the fields that still need to be completed) and the complete report can be saved, using the "Save" button, only after all the mandatory fields have been completed.

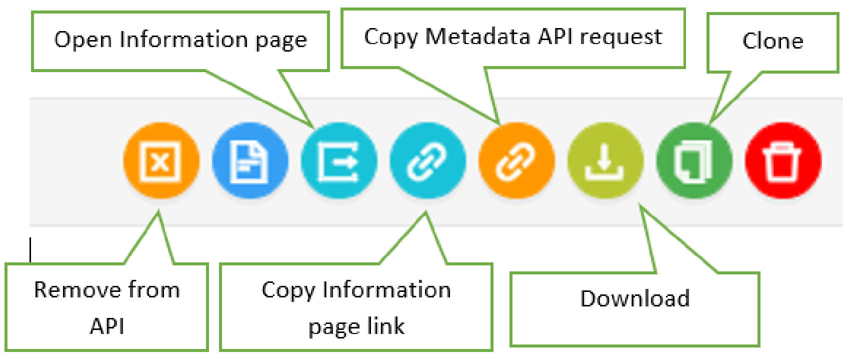
Once the report is saved, the user can use a series of tools on the line corresponding to the report:

– **Publish Report via API**
  To make report data accessible via API, after checking the necessary permissions,the user can use the "Expose via API" button, this way he can access the report data in SDMX-JSON format also through a call to the appropriate REST

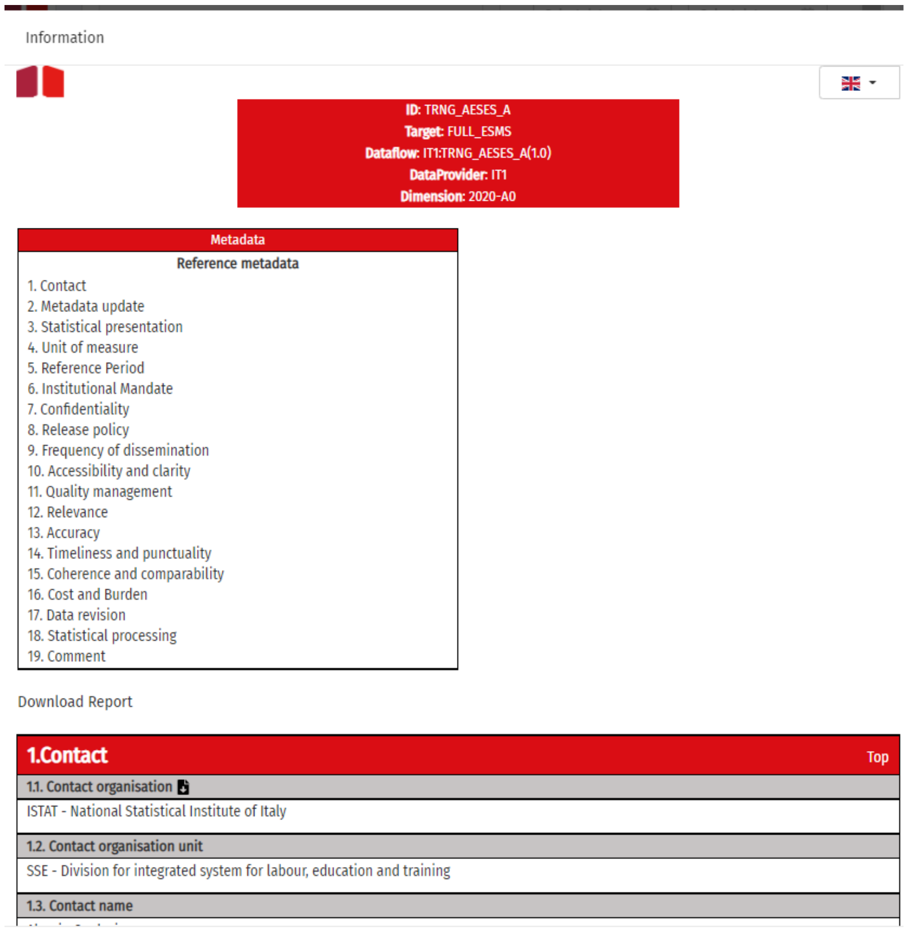
web service.



- **Remove from API - Delete**
  It is possible to cancel the publication of a report using the "Remove from API" button and delete it with the "Delete" button.
- **Open information page - Copy information page link**
  For each report exposed via API, an HTML view is available.
  In order to access directly from the interface to this visualisation, it will be sufficient to select the button "Open information page" on the row corresponding to the report:

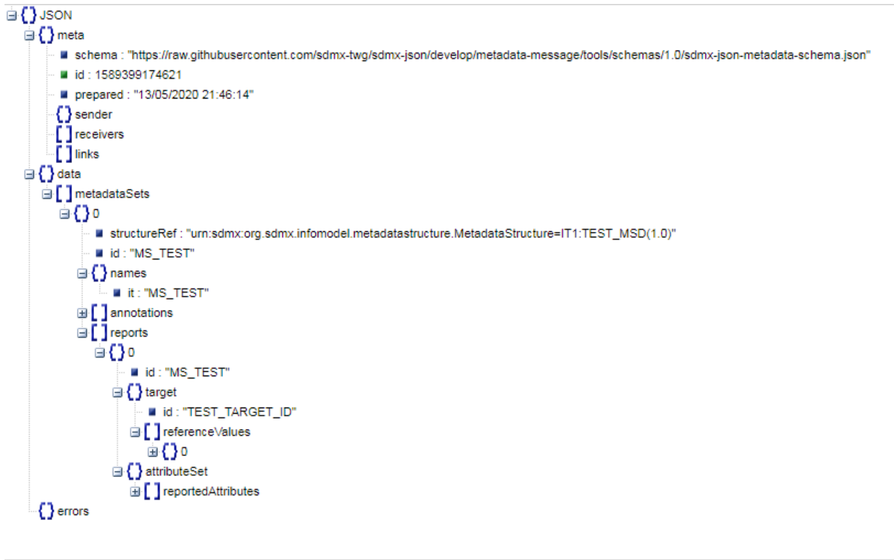


And a pop-up window like the following will open, which can also be downloaded in HTML:



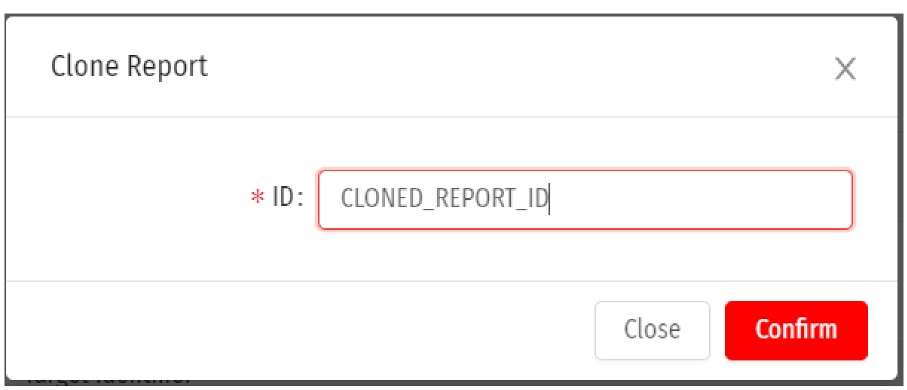
If the user wants to embed this visualization in other contexts, he can obtain the link using the "Copy information page link" button.

- **Copy Metadata API request link**
  If the user wants to access to the report data in SDMX-JSON format also through the call to the proper REST web service, he will be able to get the link through the button "Copy Metadata API request link".



- **Download**
  The button will open a window from which you can choose the export format: SDMX-ML 2.0 or SDMX-JSON.

- **Clone**
  Among the functions offered for the management of a Metadata Set there is also the duplication of a report.
  Clicking on the relative button opens a window in which you are asked to specify the id of the new report, as follows:

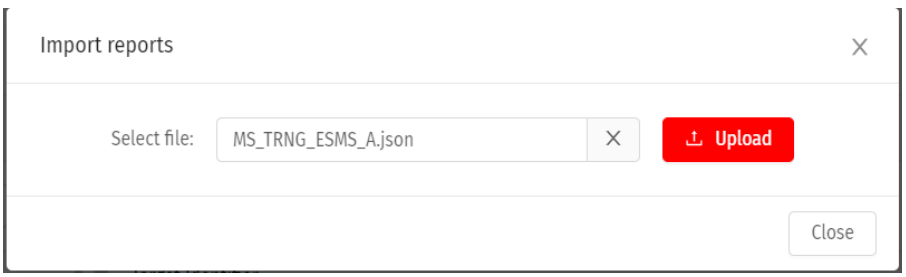


By clicking on the confirm button a new report will be created with all the data of the selected one, but with the ID previously inserted by the user and, if the operation is successful, it will be immediately present in the list of the reports of the starting Metadata Set.

Import Report
It is possible, by means of the import function, to add reports to an existing Metadata Set starting from a file in SDMX-JSON format.

In the part dedicated to the Report List of the selected Metadata Set, if the user has the right to create reports, there is an Import Report button.

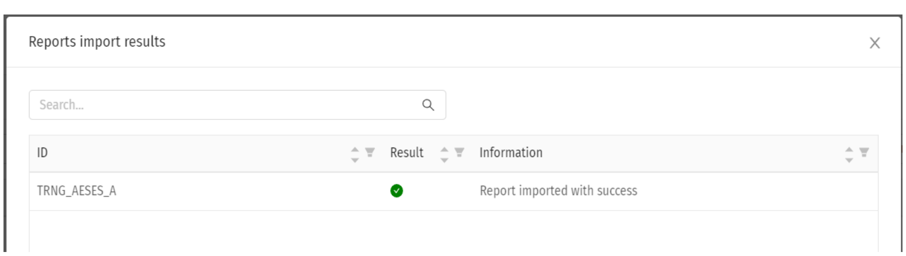
Click on Import Report to open a window as in the following figure where you can indicate the file in SDMX-JSON format containing the Metadata Set with the reports you want to import into the selected Metadata Set.
The report import is possible only if the MSD of the Metadata Set contained in the file is the same as that of the selected Metadata Set.



By clicking on the Upload button, checks will be made on the MSD, the data will be acquired and a window will be shown with information on the reports contained in the file:

From here it will be possible to choose which reports to import (among the importable ones) and confirm by clicking on the Import Report button.

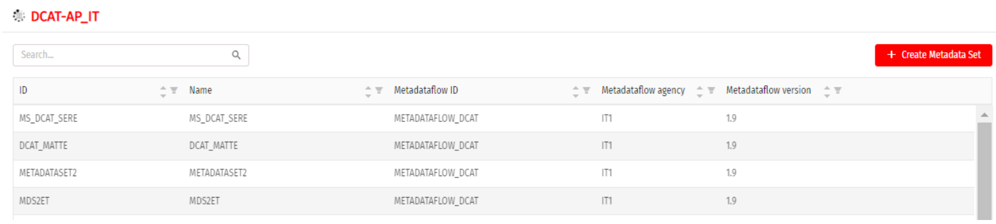
At the end of the process you will see the result of the import of each selected report:



At the end of the operation it will be possible to see the reports imported correctly (as in the example above) in the list of reports of the Metadata Set selected at the beginning.

10.2 The DCAT-AP_IT Standard

Selecting the DCAT-AP_IT item from the Menu leads to the section dedicated to the compilation/publication of Metadata according to the DCAT-AP_IT standard.

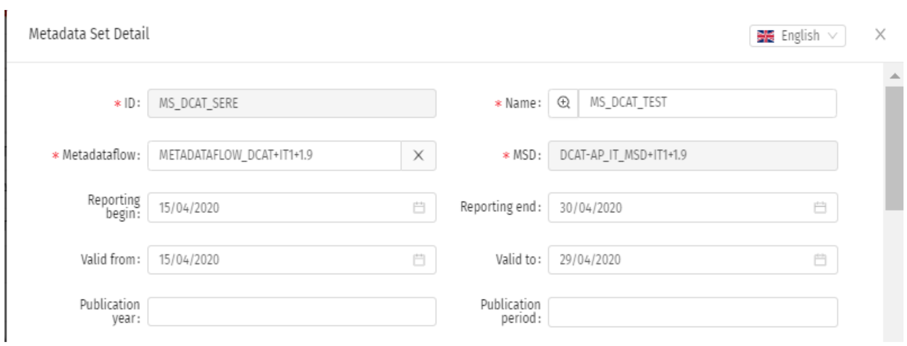


The standard requires the compilation of Metadata inherent to the Catalogue and Metadata inherent to the single published Dataflow.

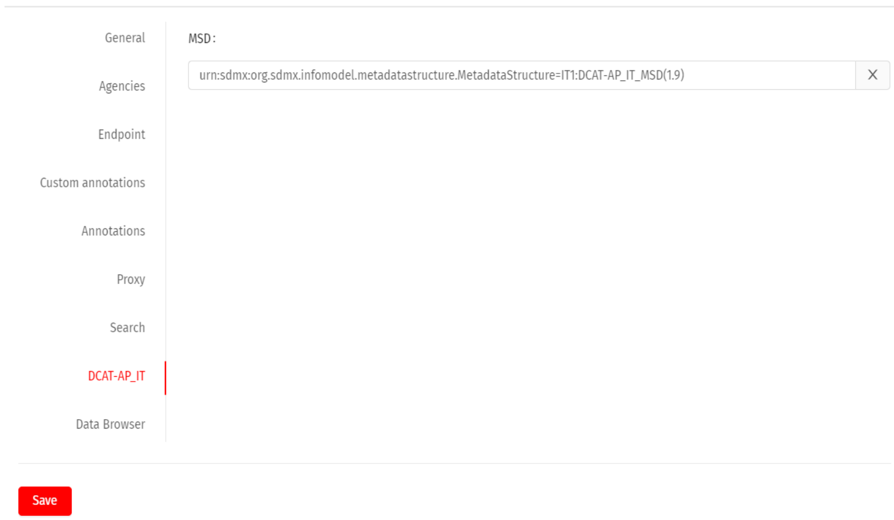
- *New MetadataSet DCAT-AP_IT*
- *Compilation of a DCAT-AP_IT report*

10.2.1 New MetadataSet DCAT-AP_IT

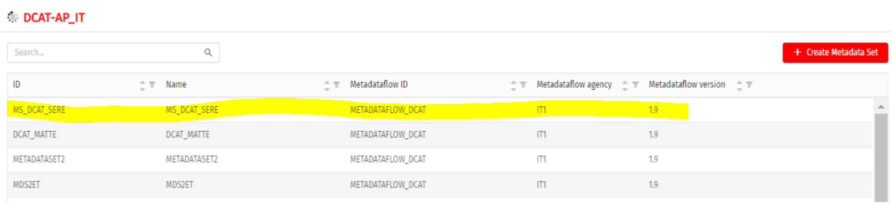
By pressing the "Create Metadata Set" button the user can create a new Metadata Set (after filling in the required fields):



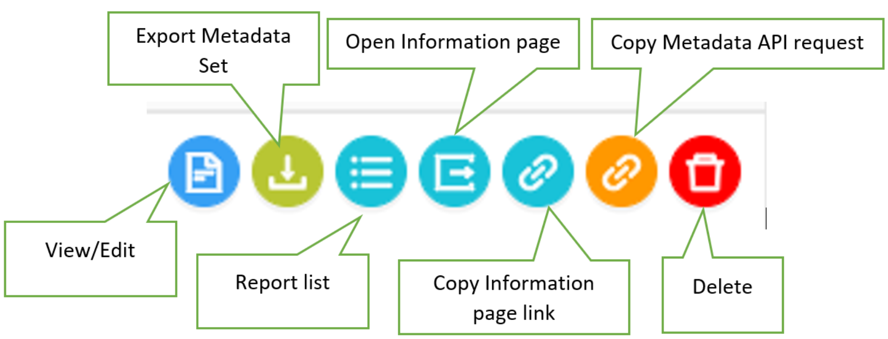
By filling in the mandatory fields the user will have to choose a Metadataflow among those available through a drop-down menu.
The Metadataflow available will only be those dedicated to the MSD of the DCAT-AP_IT (whose configuration, in the next image, is made by the Superuser at Node level) owned by the user.



Once the mandatory fields have been filled in, the user can press "Save" and the new DCAP-AP_IT Metadata Set will be inserted into the list:

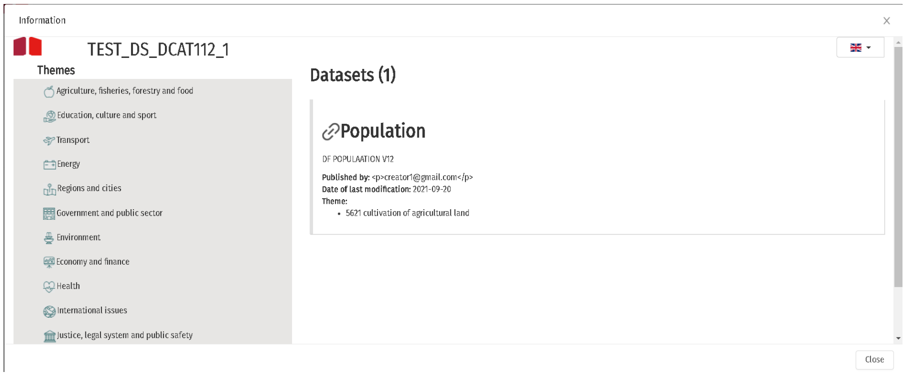


Clicking on a single Metadata Set activates the following system buttons on the chosen row:



Pressing **Export Metadata Set** the system allows the user to obtain a file in SDMX-JSON format containing the Metadata Set with all its reports.

Pressing **Open Information Page** a popup window like the following will be opened:



Pressing **Copy info page link** then system allows the user to obtain the link to embed this visualization in different contexts.

Pressing **Copy Metadata API Request Link** the system allows the user to obtain the link to access the metadata set through the call to the appropriate REST web service.

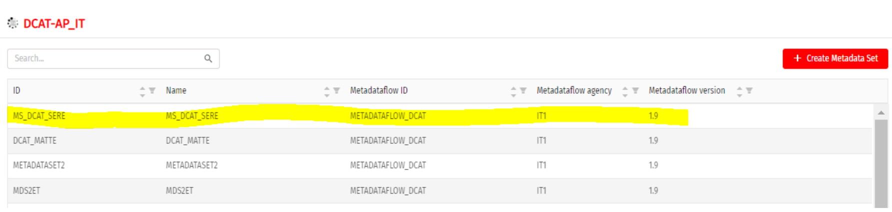
Pressing **Delete** the chosen Metadata Set will be deleted but only if there are no reports linked.

10.2.2 Compilation of a DCAT-AP_IT report

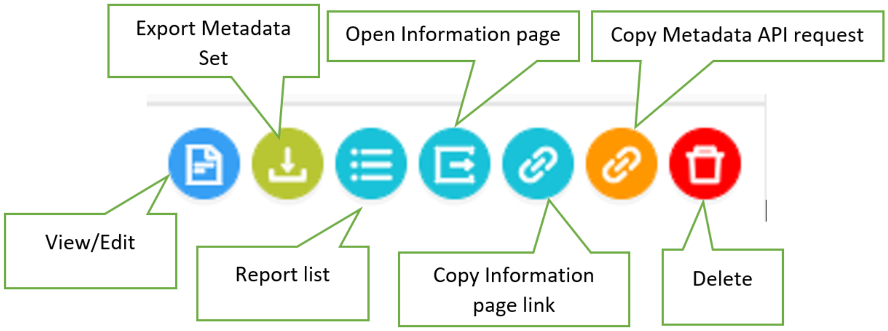
DCAT-AP_IT is a standard for meta information of catalogues, which allows the user to create metadata for the catalogue and for the dataflows categorised in it, so the dedicated DCAT-AP_IT MSD has been modelled by introducing two targets:

- The CATALOGUE target
- The DATAFLOW target

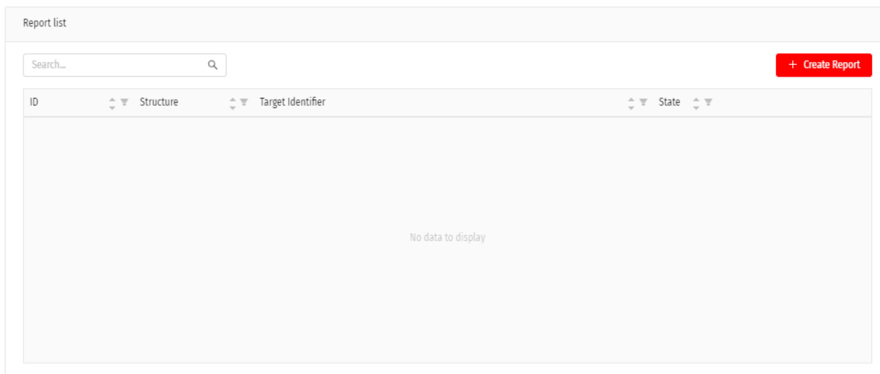
so the user can create a report for each of the two targets.
All users can create/edit/expose/delete a DCAT-AP_IT report if and only if they own the Dataflow for which they want to create the report, regardless of the permissions on the Metadataflow.
The role of ownership of a Dataflow can be transferred to other users directly from the Dataflow display interface. On the line relative to the Metadata Set DCAT-AP_IT chosen:



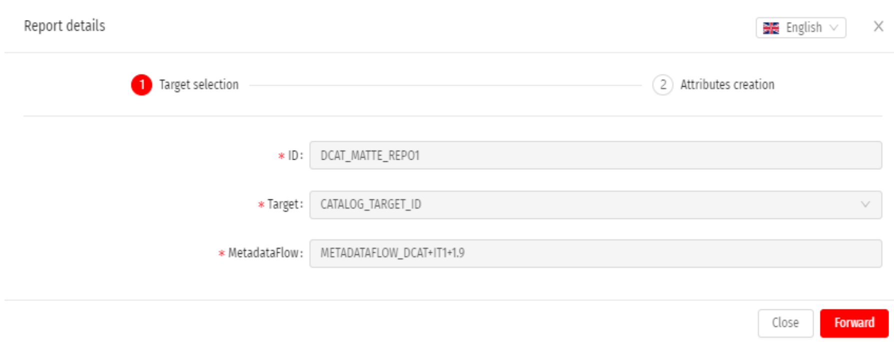
the user can insert new reports by clicking on the button "Report List"



which will open the list of reports (if any):

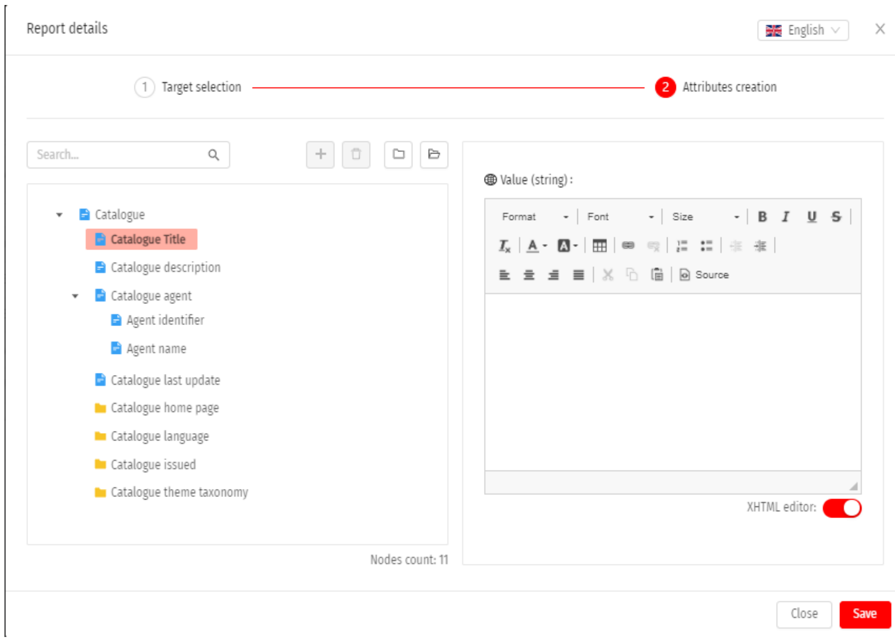


from which you can create a report for the catalogue with "Create Report":

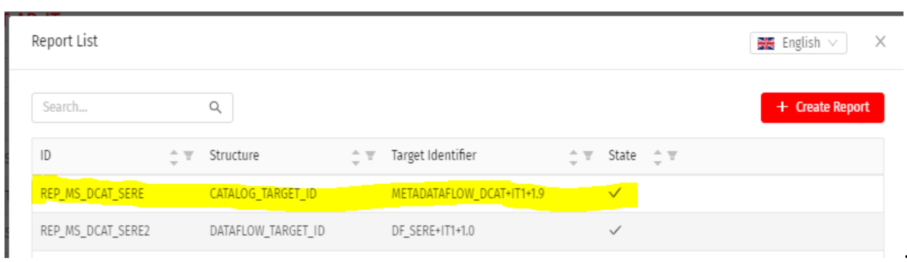


which obviously refers to the metadataflow relative to the initial DCAT-AP_IT and, by pressing "Next", requires the user to fill in a series of attributes that will be associated with the catalogue:

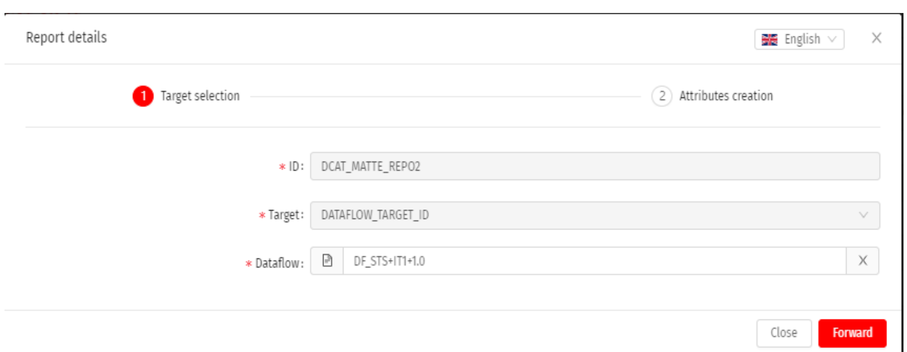
- Catalogue title
- Description
- Identifier
- Etc. . .



The report will be actually saved and will become publishable only after the user has filled in all mandatory fields.
The intermediate saves, during which the user will have on screen the report of the missing fields, will be only working drafts.
Only one report of this type is possible per catalog:



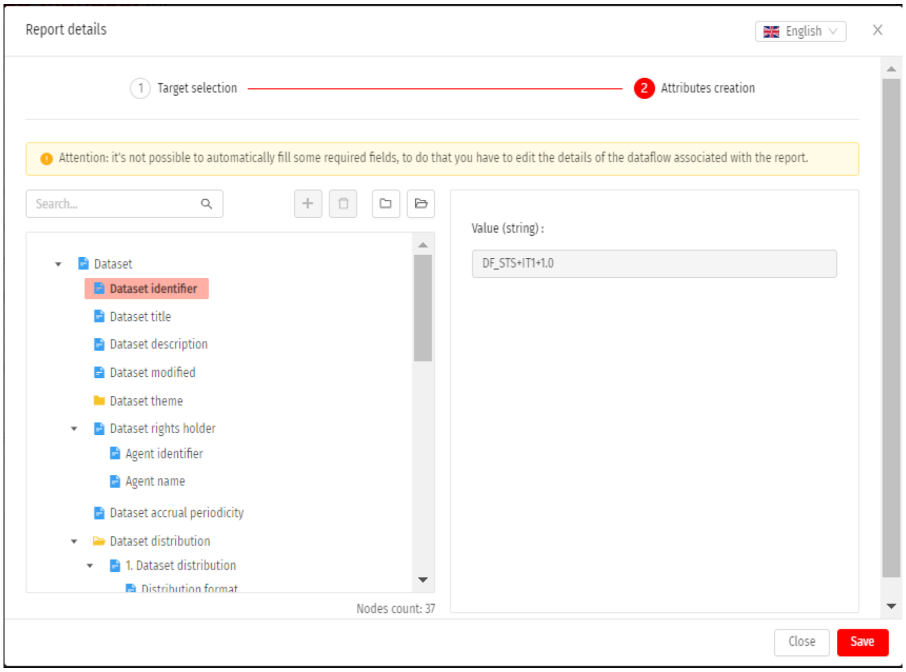
Then by pressing again "Create Report" it is possible to create a report for the dataflow:



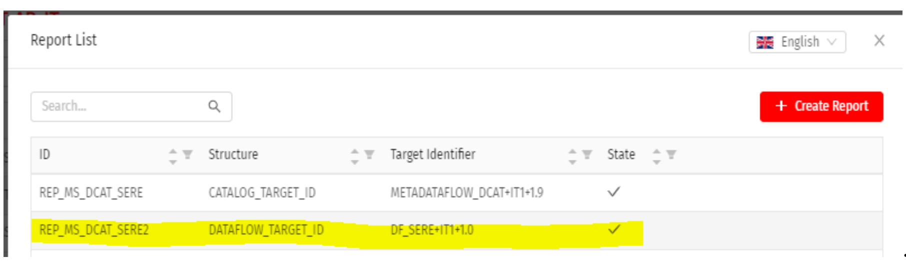
and also in this case by pressing "Next" the system requires the user to fill in a series of attributes that will be associated with the dataflow:

- Description of the dataset
- Date of last modification
- Theme of the dataset

- Etc.

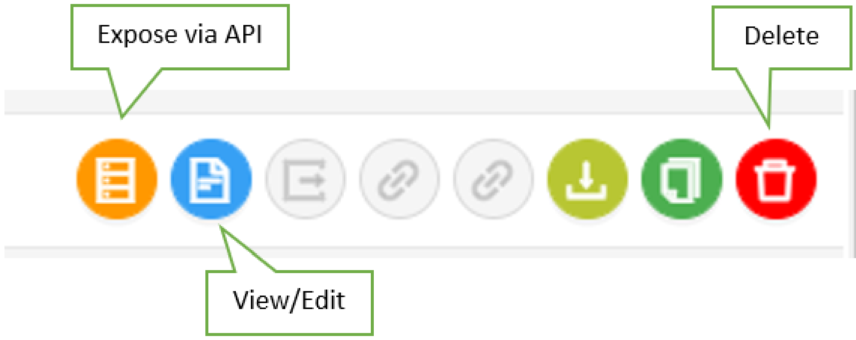


similarly, the report will actually be saved and will become publishable only after the user has filled in all mandatory fields.
The user can create more than one type of report.



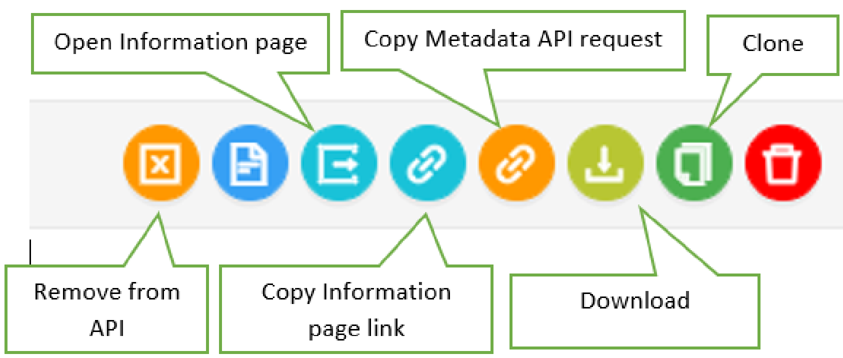
Once the report is saved, the user can use a series of tools on the line corresponding to the report of interest:

- **Publish Report via API**
  The information of the published report can also be consulted via REST API (CKAN compliant); in order to publish the report of a Dataflow via API it will be sufficient that a user with sufficient rights exposes it via the Expose via API button.

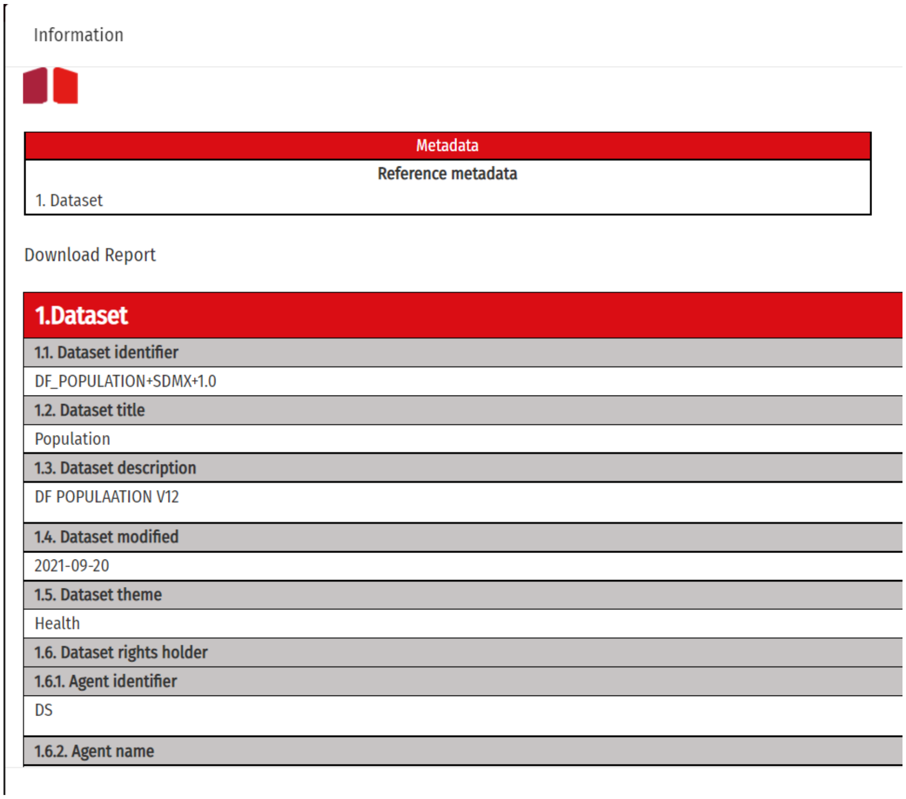


The exposure of metadata will respect the standard CKAN format (version 3), limited to guarantee the federation of the entire Catalog with the National Metadata Catalog (Dati.gov.it).

- **Remove from API - Delete**
  It is possible to cancel the publication of a report using the "Remove from API" button and to delete it with the "Delete" button, taking care to delete the report dataflows first, otherwise the report in the catalogue cannot be deleted.
- **Open information page - Copy info page link**
  For each report exposed via API an HTML view is available.
  In order to access directly from the interface to this visualisation, it will be sufficient to select with the mouse the button "Open information page" on the row corresponding to the report:

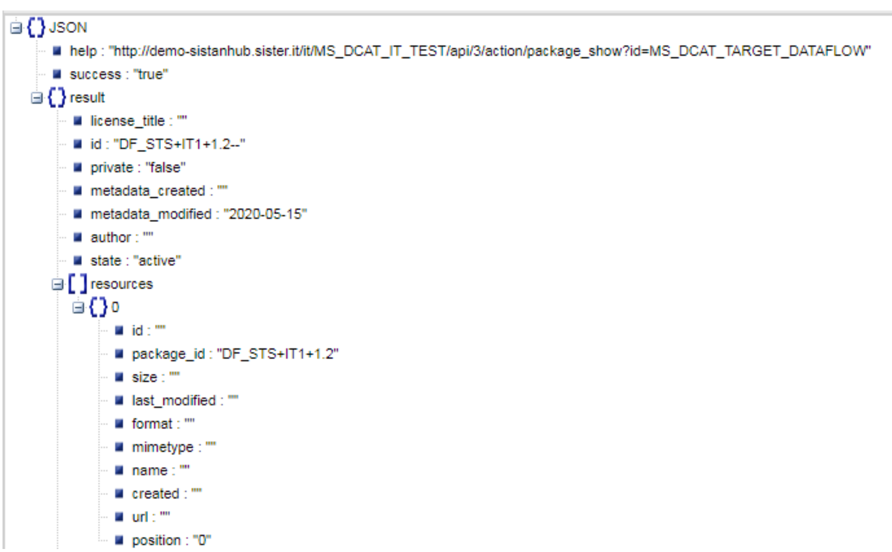


And a pop-up window like the following will open, which can also be downloaded in HTML:

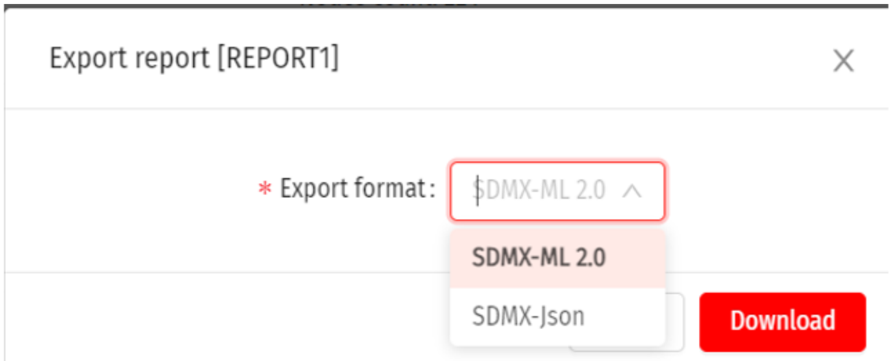


If the user wants to embed this visualization in different contexts, he can obtain the link by clicking on the "Copy information page link" button.

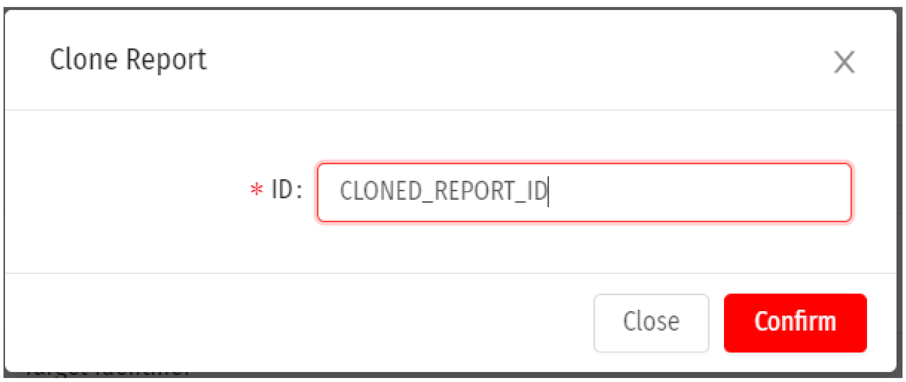
- **Copy Metadata API Request Link**
  If the user wants to access the report data in SDMX-JSON format also through the call to the proper REST web service, he will be able to get the link through the button "Copy Metadata API Request Link".



- **Download**
  The button will open a window from which you can choose the export format: SDMX-ML 2.0 or SDMX-JSON.

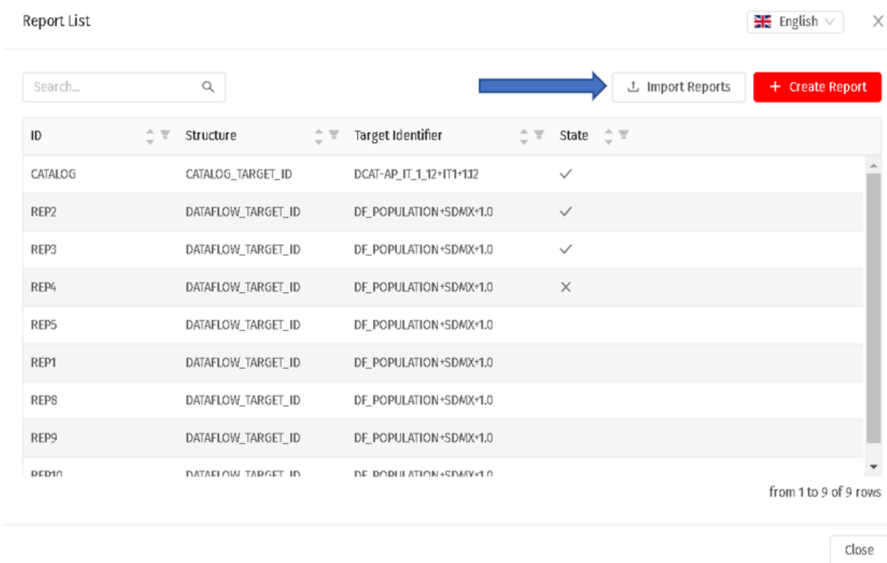


- **Clone**
  Among the functions offered for the management of a Metadata Set there is also the duplication of a report.
  Clicking on the relative button opens a window in which you are asked to specify the id of the new report, as follows:

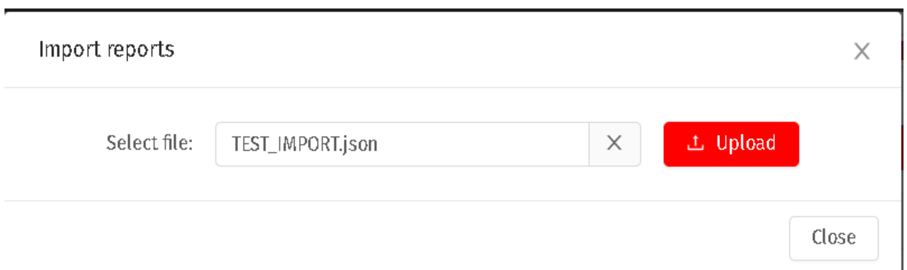


By clicking on the confirm button a new report will be created with all the data of the selected one, but with the ID previously inserted by the user and, if the operation is successful, it will be immediately present in the list of the reports of the starting Metadata Set.

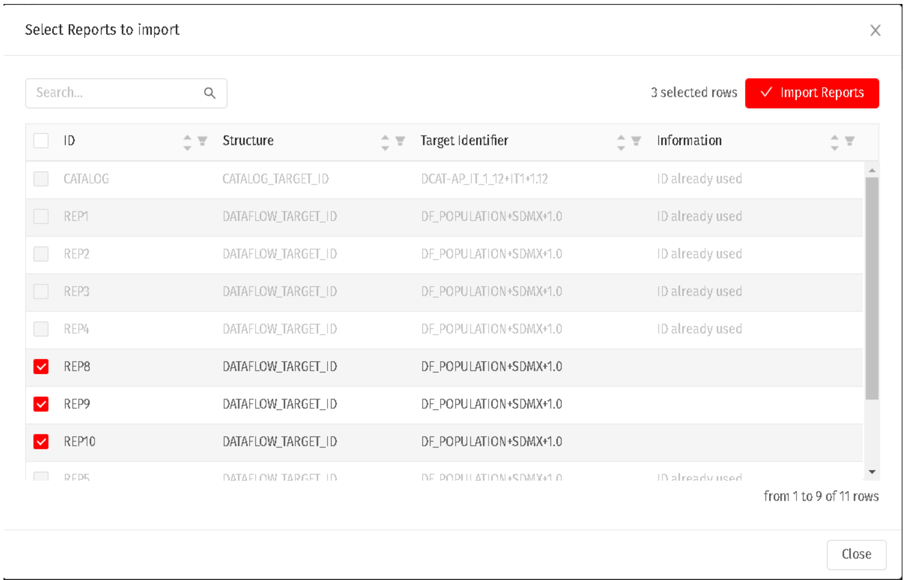
Import Report
It is possible, by means of the import function, to add reports to an existing Metadata Set starting from a file in SDMX-JSON format.
In the part dedicated to the Report List of the selected Metadata Set, if the user has the right to create reports, there is an Import Report button.



Click on Import Report to open a window as in the following figure where you can indicate the file in SDMX-JSON format containing the Metadata Set with the reports you want to import into the selected Metadata Set.
The report import is possible only if the MSD of the Metadata Set contained in the file is the same as that of the selected Metadata Set.

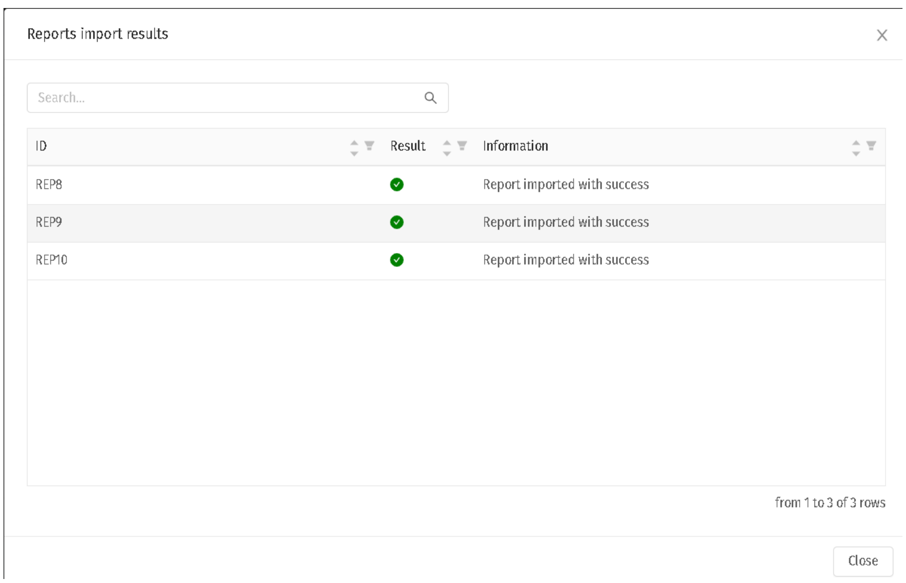


By clicking on the Upload button, checks will be made on the MSD, the data will be acquired and a window will be shown with information on the reports contained in the file:



From here it will be possible to choose which reports to import (among the importable ones) and confirm by clicking on the Import Report button.

At the end of the process you will see the result of the import of each selected report:



At the end of the operation it will be possible to see the reports imported correctly (as in the example above) in the list of reports of the Metadata Set selected at the beginning.

10.3 Metadata API

Below is a description of the exposed APIs:

- **PING**
  Checks whether the module endpoint is running by returning a boolean value (true/false).
  [METADATA_API_URL]/Ping
- **Package_search**
  Exposes the list of dataflows "metadata" and made available via API.
  [METADATA_API_URL]/[LANGUAGE]/[DCAT_METADATASET_ID]/api/3/action
  /package_search?sort=id+asc&start=**[STARTING_FROM]**
  &rows=**[NUM_RISULT]**


```
   success: "true",
 - result: {
       count: "2",
       sort: "id asc",
     - results: [
         + {…},
         + {…}
       ]
   }
}
```


  filtered by theme:
  [METADATA_API_URL]/[LANGUAGE]/[DCAT_METADATASET_ID]/api/3/action
  /package_search?q=groups:**["[THEME]"]**
- **Package_show**
  Exposes the detail of the individual metainformation in the dataflow.
  [METADATA_API_URL]/[LANGUAGE]/[DCAT_METADATASET_ID]/[LANGUAGE]
  /api/3/action/package_show?id=**[IDENTIFIER]**

```
result: {
    license_title: "PUBL",
    id: "SDG_4_RECORDS-IT1-1.0",
    private: "false",
    metadata_created: "",
    metadata_modified: "2019-05-21T13:20:51",
    author: "IT1",
    state: "active",
  - resources: [
      - {
            id: "Distribuzione dataflow SDG_4_RECORDS",
            package_id: "SDG_4_RECORDS",
            size: "",
            last_modified: "2019-05-21T13:20:51",
            format: "SDMX",
            mimetype: "SDMX",
            name: "Distribuzione dataflow SDG_4_RECORDS",
            created: "2019-05-21T13:20:51",
            url: "https://demo-sistanhub.sister.it/MA_WS/sdmx/rest/data/IT1,SDG_4_RECORDS,1.0/all",
            position: "0"
        }
    ],
  - tags: [
      - {
            display_name: "",
            name: "",
            state: "active",
            id: ""
        }
    ],
  - groups: [
      - {
            id: "Economia e finanze",
            package_count: "0",
            isOrganization: "false"
        }
    ],
  - organization: {
        is_organization: "true",
        id: "IT1",
        name: "IT1"
    },
    name: "",
    isopen: "true",
```


- **Group_list**
  Displays the list of topics for which there is at least one associated metadata report.
  [METADATA_API_URL]/[LANGUAGE]/[DCAT_METADATASET_ID]
  /[LANGUAGE]/api/3/action/group_list
- **GetMetadata**
  Displays metadata in sdmx-json format, searching for a specific Metadata Set or Report and filtering by published data (default choice) or all.
  [METADATA_API_URL]/api/getMetadata?metadataSetId=
  [METADATASET_ID]&reportId=**[REPORT_ID]**

```json
{
  "meta": {
    "schema": "https://raw.githubusercontent.com/sdmx-twg/sdmx-json/develop/metadata-message/tools/schemas/1.0/sdmx-json-metadata-schema.json",
    "id": 1576770598557,
    "prepared": "19/12/2019 16:49:58",
    "sender": {

    },
    "receivers": [

    ],
    "links": [

    ]
  },
  "data": {
    "metadataSets": [
      {
        "id": "DCAT_AP_IT_METADATASET",
        "names": {
          "it": "DCAT-AP_IT MetadataSet"
        },
        "annotations": [
          {
            "id": "MetadataSetId",
            "texts": {
              "en": "10"
            },
            "text": "10"
          },
          {
            "id": "MetadataflowId",
            "texts": {
              "en": "DCAT_AP_IT_METADATAFLOW"
            },
            "text": "DCAT_AP_IT_METADATAFLOW"
          },
```


10.4 Customizations

10.4.1 Catalog widget customizations

Title modification

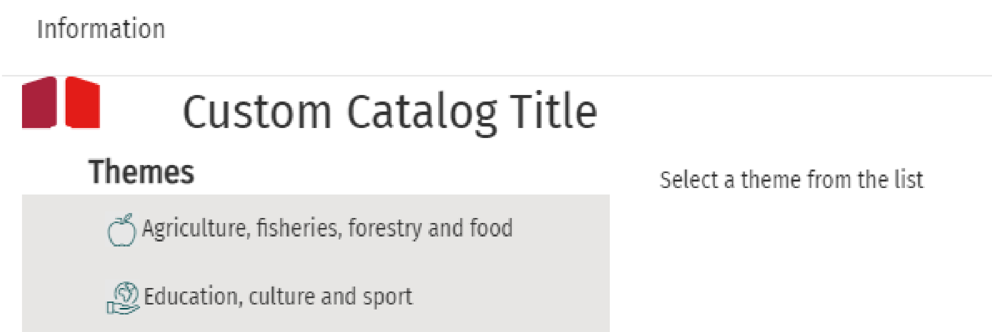
It is possible to edit the page's title only by accessing the CategoryTemplate.html file stored in the **[HOME_DEPLOY_CLIENT_MDM]\static\referenceMetadata\template** folder and by searching for the div element identified by the **id=header-title-catalog**:

```html
<body onload="loadLanguageMenu()">
   <div class="container-fluid">
       <div class="row">
           <div class="col-1">
               <div id="header-img-catalog">
                   <img src="../images/istat_logo_mini.png" class="Cl-Header-Img-Catalog">
               </div>
           </div>
           <div class="col-10">
               <div id="header-title-catalog" class="Cl-Header-Title-Catalog"></div>
           </div>
           <div class="col-1">
               <div class="btn-group Cl-flag-btn-group">
                   <button type="button" class="btn btn-default Cl-dropdown-menu Cl-flag-btn
                       <div id="spa-language-it"><i class="Cl-FlagIT Cl-Flag"></i><i class="
                       <div id="spa-language-en"><i class="Cl-FlagEN Cl-Flag"></i><i class="
                   </button>
```

Once the reference line has been identified, the user can add the text he wants to display as title. For example, by setting "Custom Catalog Title" as shown below:

```
<body onload="loadLanguageMenu()">
<div class="container-fluid">
  <div class="row">
    <div class="col-1">
      <div id="header-img-catalog">
        <img src="../images/istat_logo_mini.png" class="Cl-Header-Img-Catalog">
      </div>
    </div>
    <div class="col-10">
      <div id="header-title-catalog" class="Cl-Header-Title-Catalog">Custom Catalog Title</div>
    </div>
    <div class="col-1">
      <div class="btn-group Cl-flag-btn-group">
        <button type="button" class="btn btn-default Cl-dropdown-menu Cl-flag-btn" data-toggle="dropdown">
          <div id="spa-language-it" style="display: none;"><i class="Cl-FlagIT Cl-Flag"></i><i class="caret"></i></div>
          <div id="spa-language-en" style="display: block;"><i class="Cl-FlagEN Cl-Flag"></i><i class="caret"></i></div>
        </button>
```

The DCAT-AP_IT catalog information page will show the new inserted title:



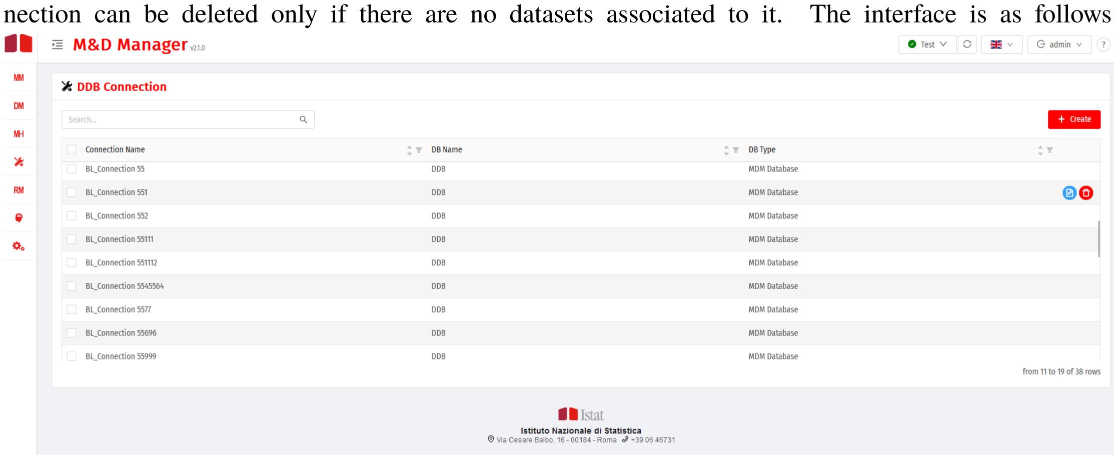
CHAPTER

ELEVEN

MAPPING HANDLER

11.1 DDB connection

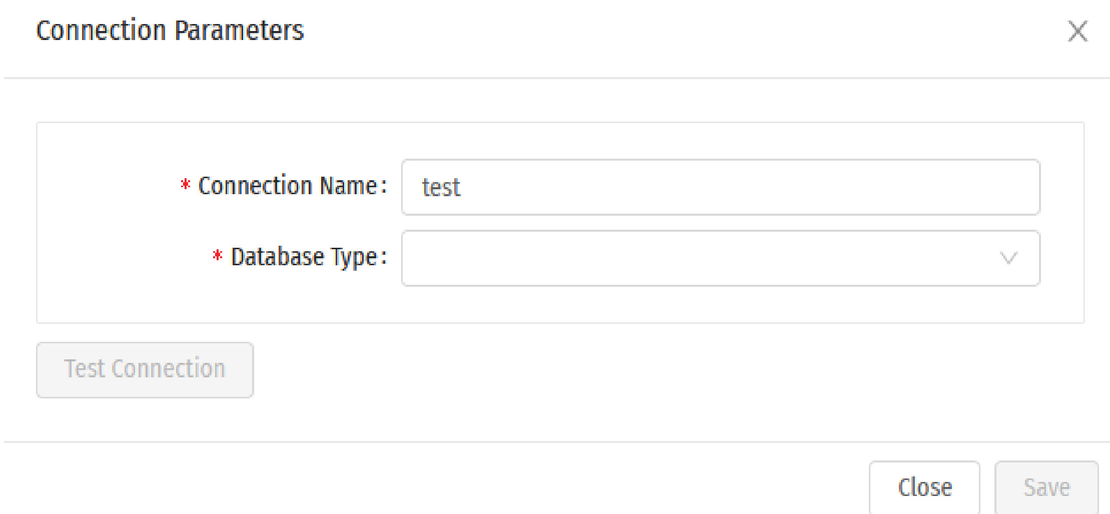
The **DDB connection** feature is located in the left-hand side menu inside the "Mapping Handler" item (MH). This feature enables the user to see the list of connections that he has access to and lets him create new connections, edit existing ones or delete ones that are not needed anymore. A connection can be deleted only if there are no datasets associated to it. The interface is as follows:



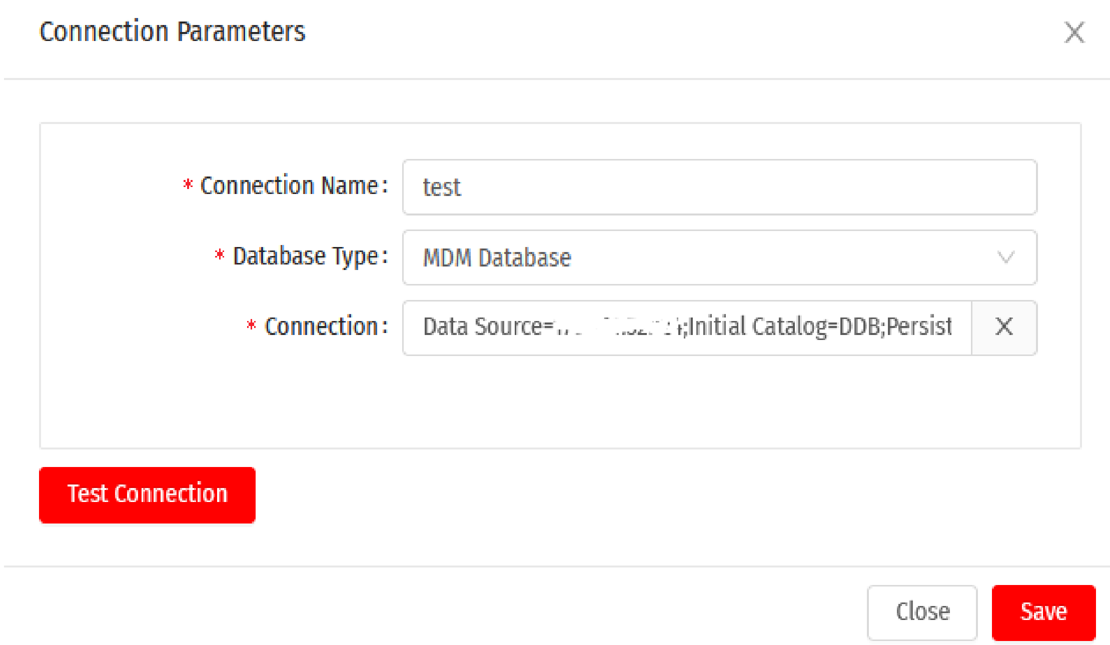
The "Create" button on the top right lets the user create a new connection, while on each row on the right the user can click on the blue icon to view/edit the connection or on the red icon to delete it. It is also possible to batch delete the connections by selecting the ones that need to be deleted on the left: when that is done a "Delete" button will appear on the left of the "Create" button.

When a user clicks on the "Create" button, a modal shows up prompting the user to enter the name of the connection and the database type, which can be chosen amongst:

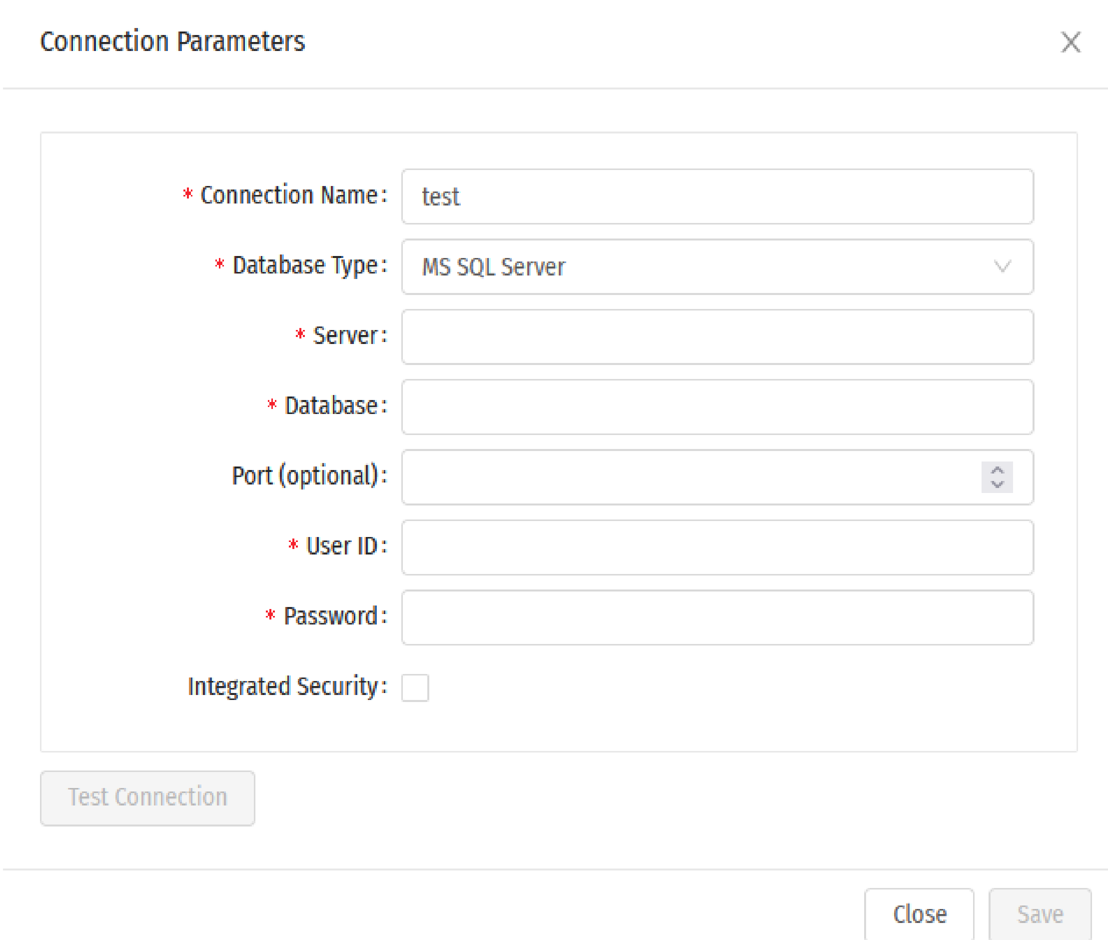
- MDM Database
- MS SQL Server
- Oracle
- MySQL/MariaDB



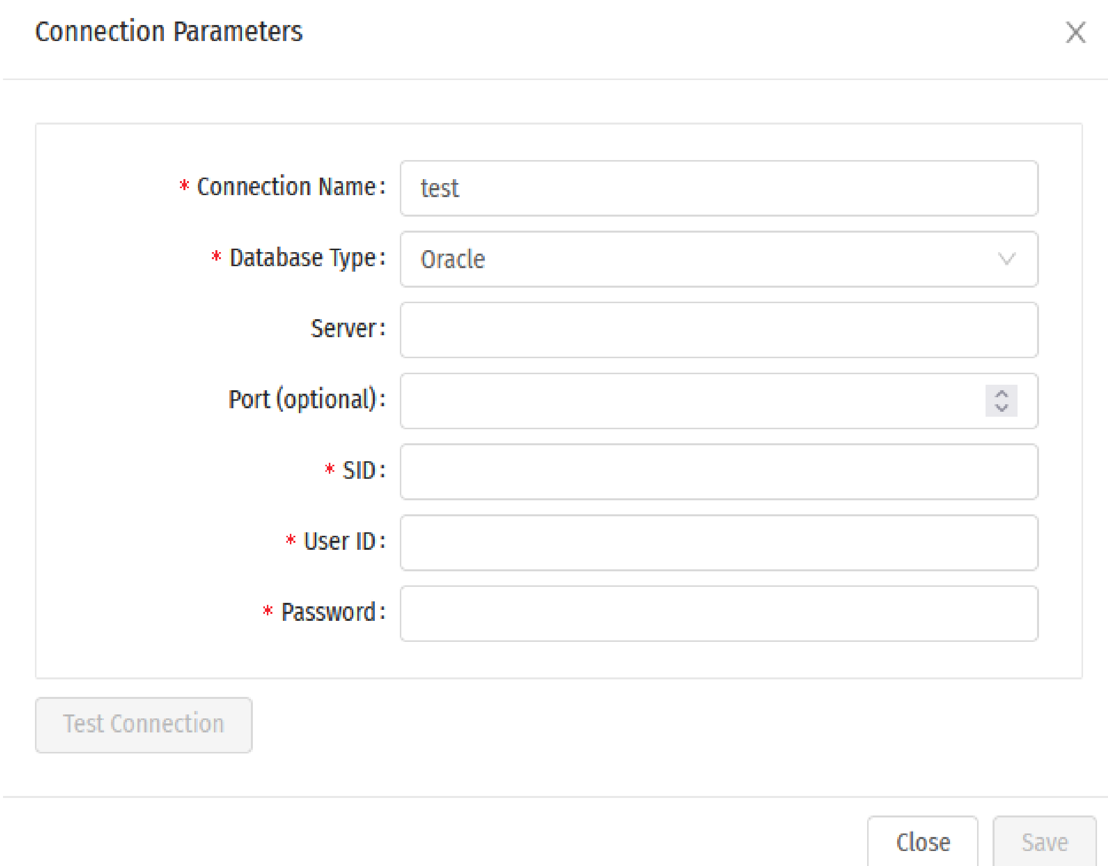
Each connection type has its own parameters to be filled. The MDM Database is the simplest one, since it uses MDM to choose amongst already-configured connections.



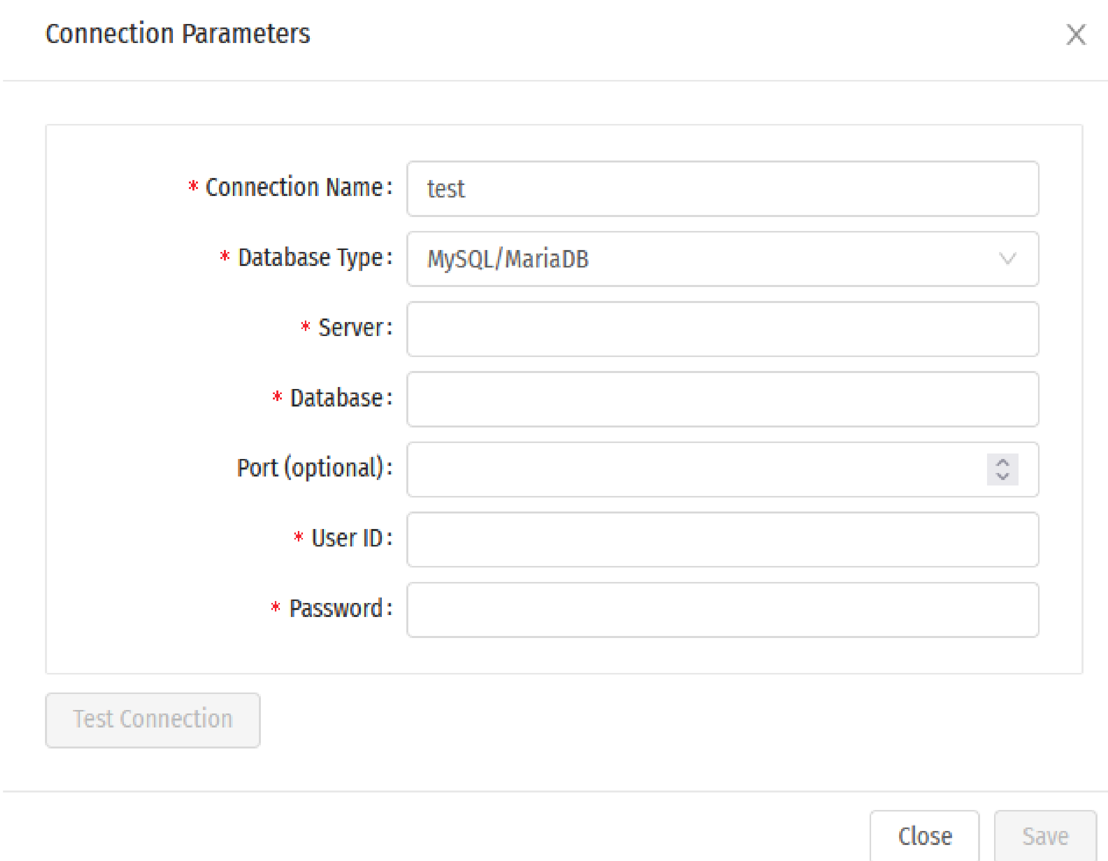
If a user selects MS SQL Server, the interface is as follows:



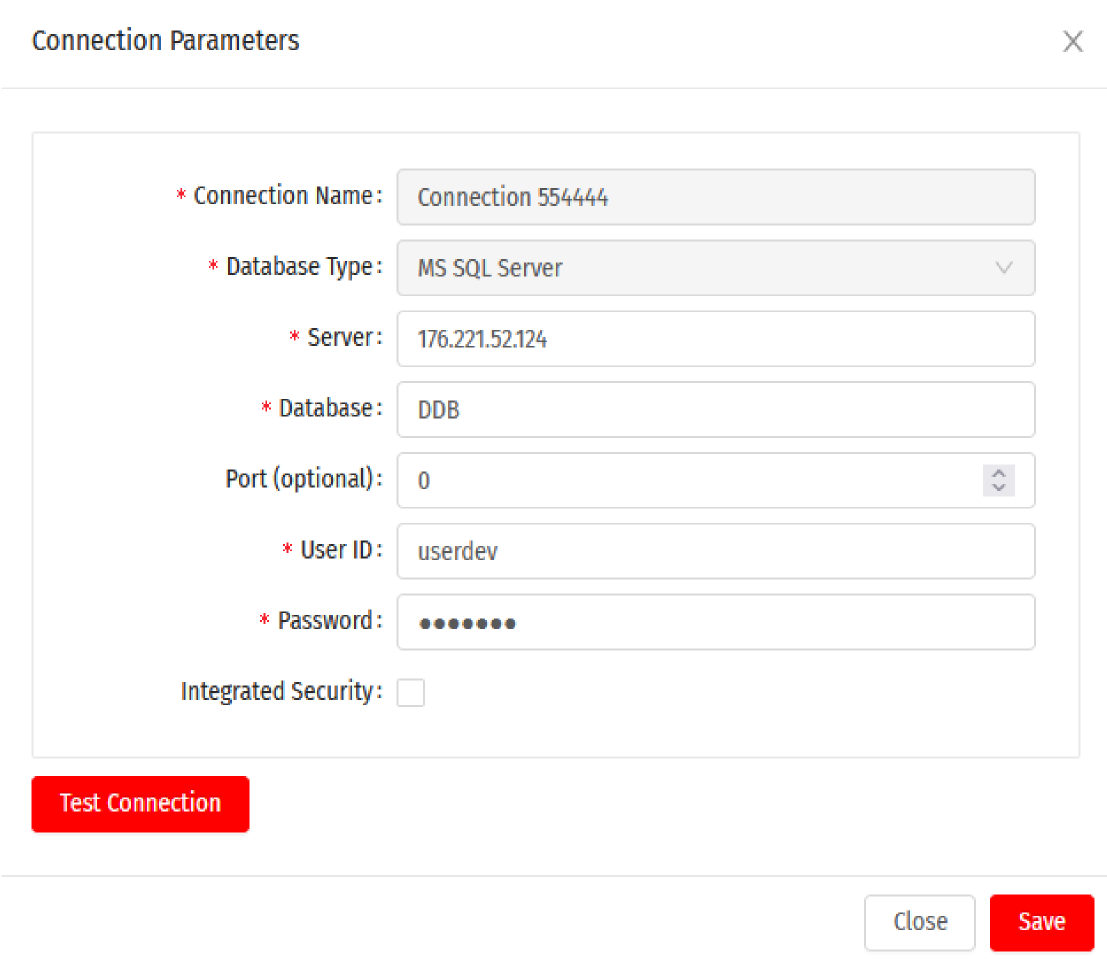
If a user selects Oracle Server, the interface is as follows:



If a user selects MySQL/MariaDB Server, the interface is as follows:

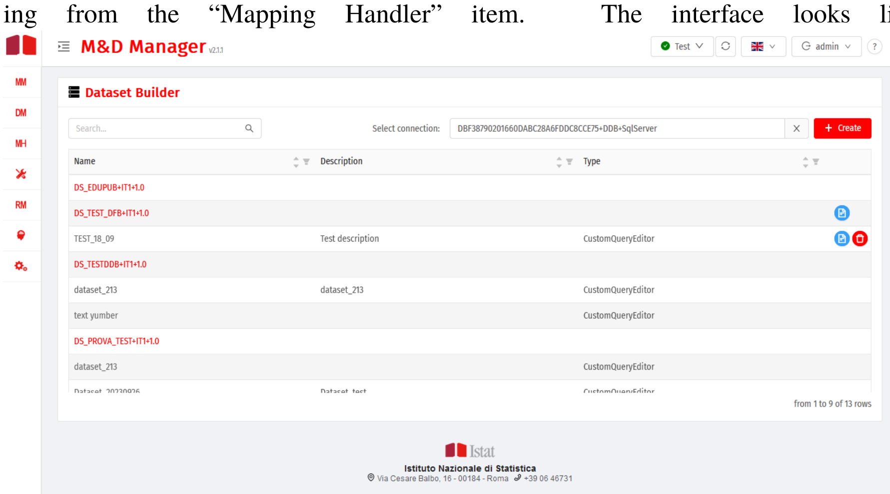


Once the connection has been chosen the user can click on the "Test Connection Button" to make sure that the connection is working. After that the connection can be saved with the "Save" button.
In order to edit an existing connection, the user can click on the blue view/edit icon present on each row of the connection list and a popup similar to the creation one will appear. In the edit mode, the user will not be allowed to change the connection name or the database type.



11.2 Dataset builder

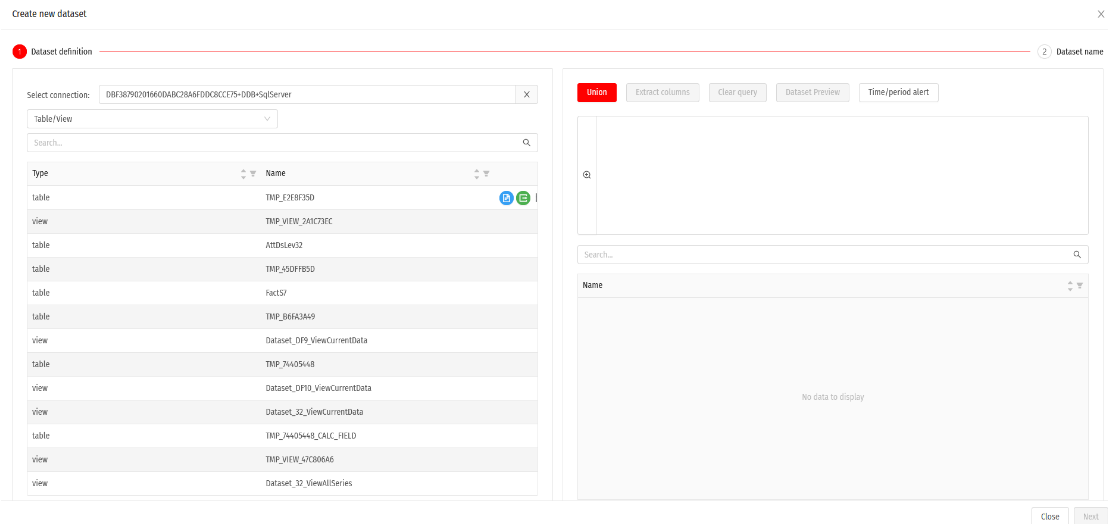
The **Dataset Builder** function is present in the left-hand side menu starting from the "Mapping Handler" item. The interface looks like the following:



In the top right corner, a connection can be chosen among the ones that the user has access to. On its right, the button to create a new dataset is present. Below that, the list of datasets can be seen. The list will be visible **only after** a connection has been chosen.

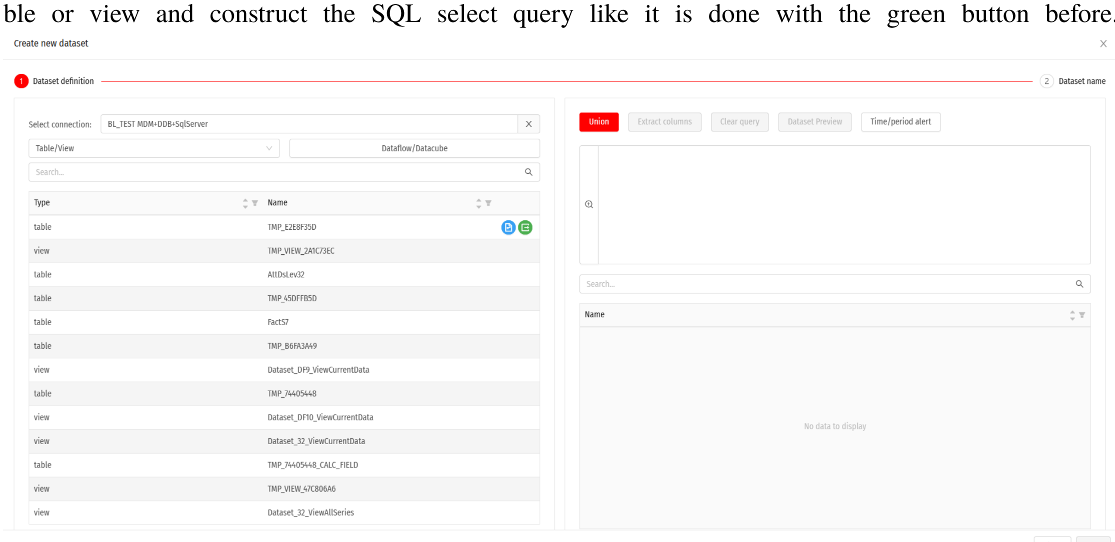
The datasets in red are the ones linked to a dataflow built with the *Dataflow builder* and cannot be edited or deleted, only viewed in read-only mode.
The red button on each row can be used to delete a dataset, however, in case the dataset is already used in one or more mapping sets, an alert will popup telling the user that the removal of the dataset will not be possible.

When the user selects the "Create" button, a new modal pops up:



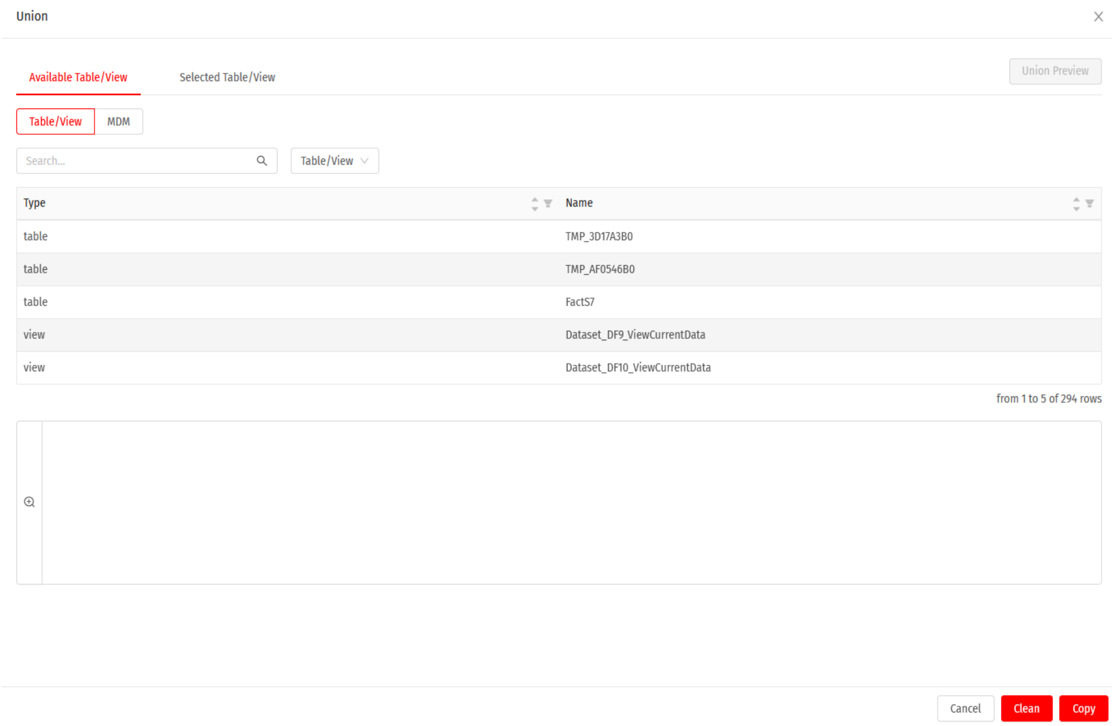
On the left of the modal, the list of tables and views of the connection is present. This can be filtered with the selector on top: it can show both tables or views, or only one of the two types. The user can also click on the blue button on each row to preview its data. By clicking the green button, a SQL select query is built using the columns of the table or view in question and is written on the right, where the user can modify it as needed.

If the connection is of type MDM, an additional button pops up besides the Table/View selector and lets the user select a datacube or dataflow. Then the app will get the relative table or view and construct the SQL select query like it is done with the green button before.

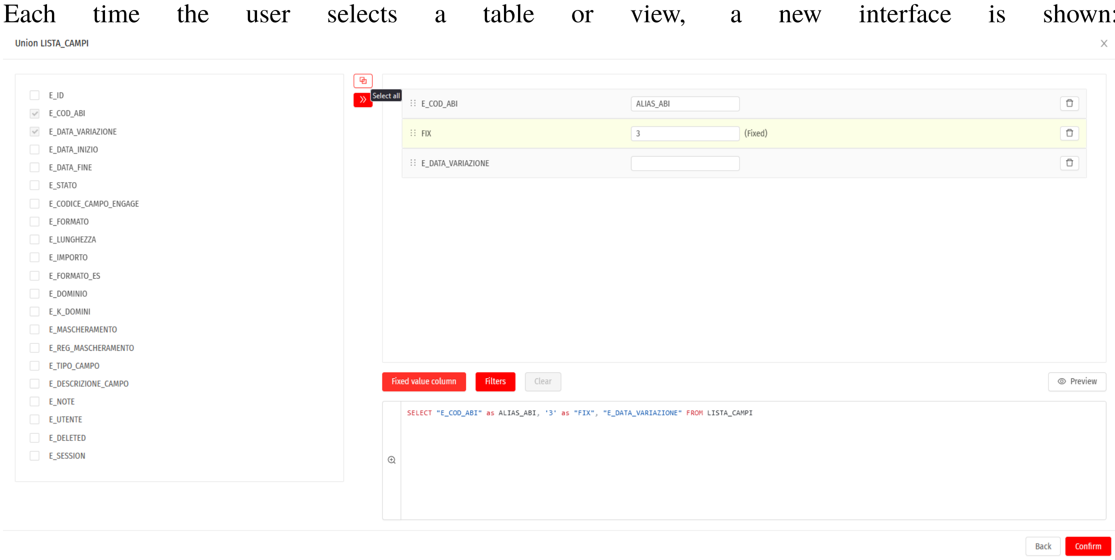


When the user has written a query in the text box, the "Next" button will become selectable. Before doing that, the user can choose to extract the columns of the query on the table below with the "Extract columns" button, or choose to clear the query and the columns with the "Clear query" button. The user can also preview the data of the SQL query with the "Dataset Preview" button. The button "Time/Period

alert" shows a modal that notify the user of the need to have the fields used for temporal transcoding with a fixed length.
If the user wants to build a more complex union SQL query, he can click on the "Union" button. A new modal will appear:

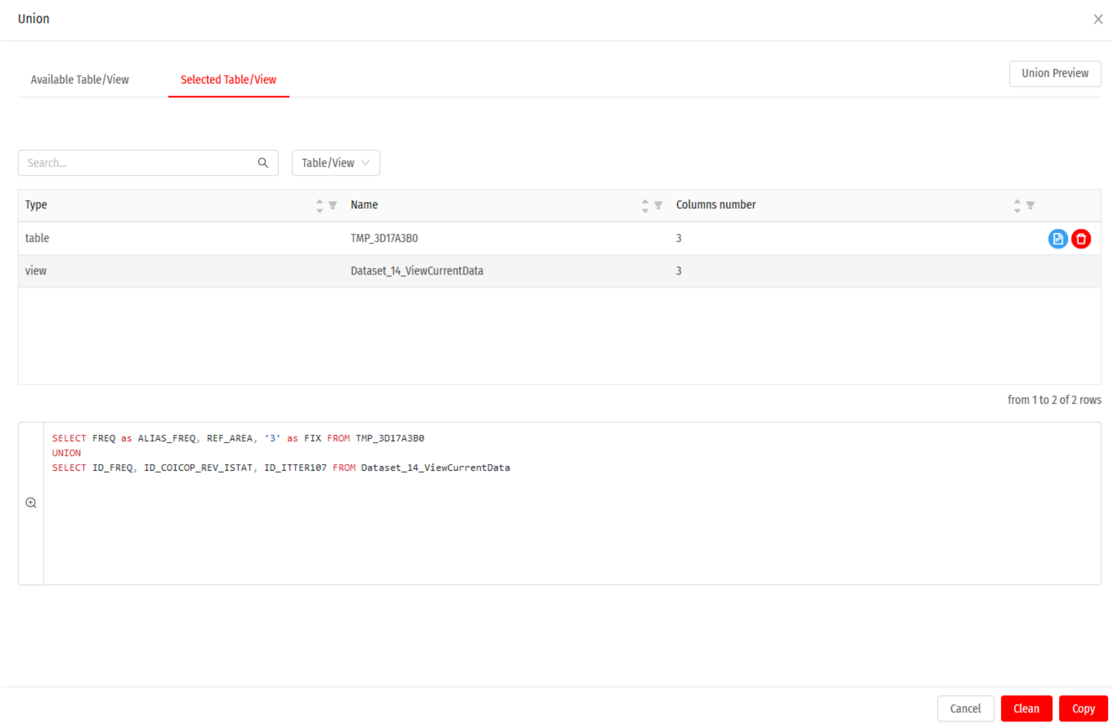


This new modal lets the user choose multiple tables or views of the connection in order to create a SQL query. If the connection is of type MDM, the MDM button is also available, which lets the user select datacubes or dataflows in order to extract the relative table/view and work on that for the union.
Each time the user selects a table or view, a new interface is shown:

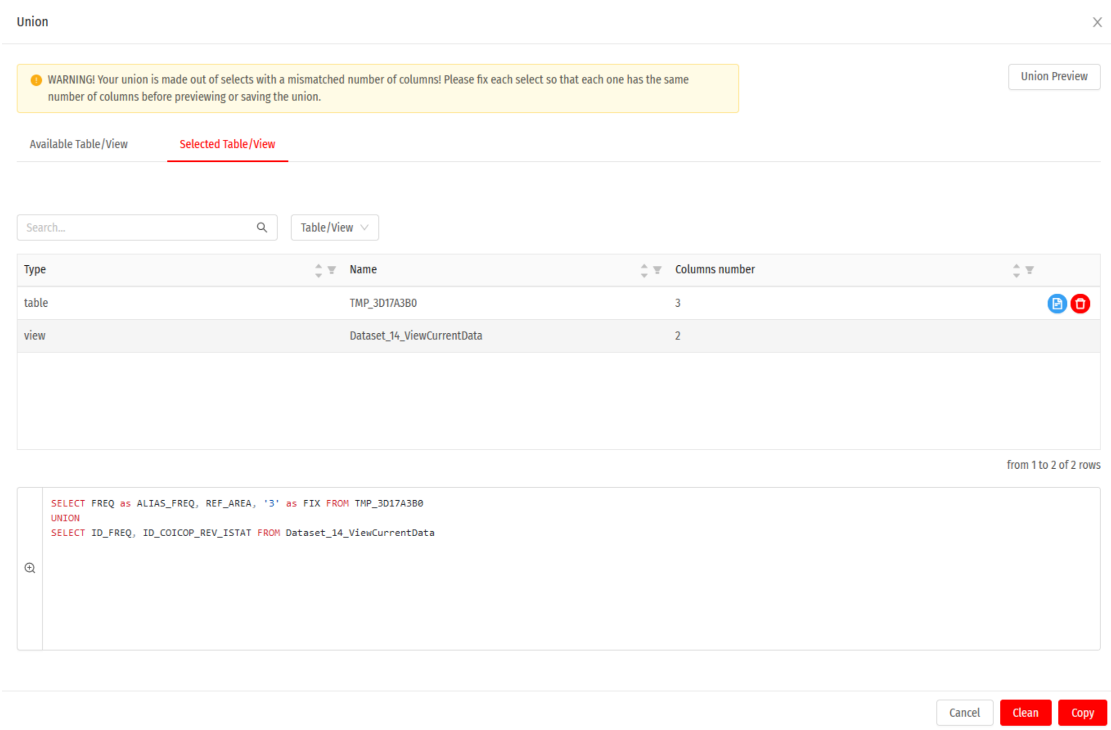


Here the user can select the columns he is interested in on the left. He can also select them all with the "Select All" button (which is the white button above the red one with the double arrows pointing right). When using the red button with the double arrows pointing right, the columns are moved to the right and are then included in the query. Here the user can choose aliases for each column or choose fixed value columns with the button on the bottom left. Once a fixed value column has been created, it will appear on the right among the others, but will be a different color and have a "(Fixed)" description beside the input

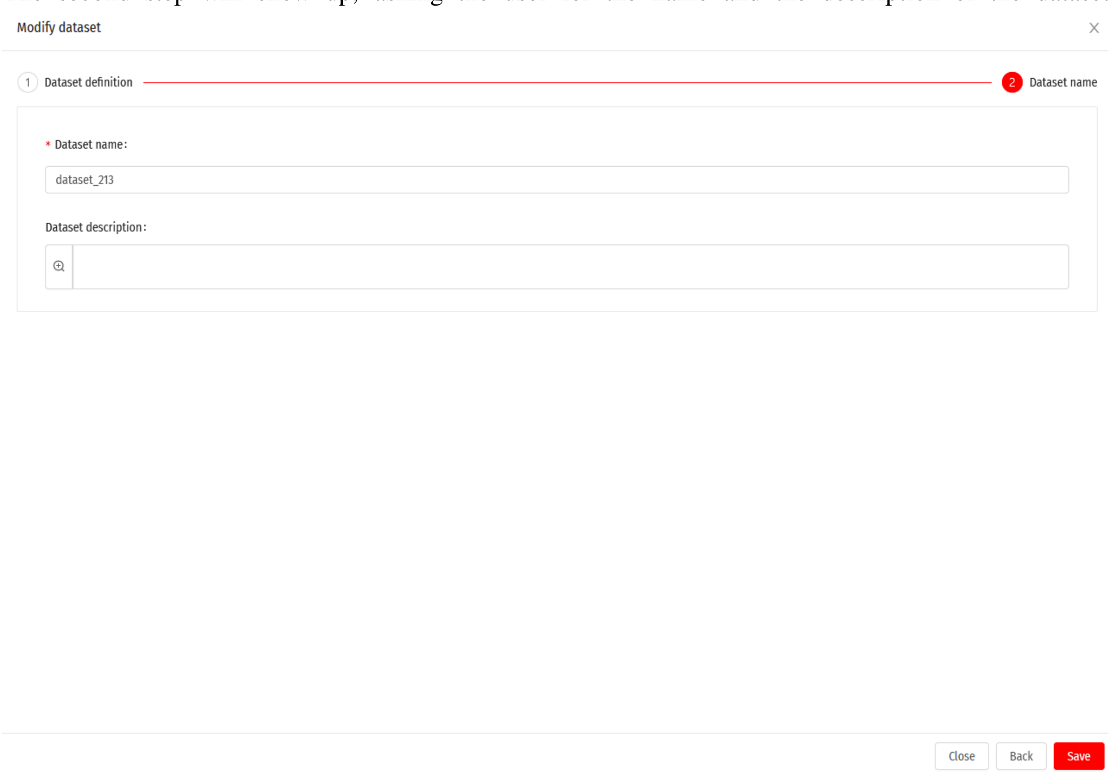
text. All chosen columns can be reordered and the change is reflected immediately in the query below. The columns can be deleted as well with the trash icon on the right. The user can also decide to clear all of them at once with the “Clear” button.
With the “Preview” button, the user can have a preview of the data. With the “Filters” button, the user can tweak the selection with filters on each column.
Once the query has been created, with the “Confirm” button the query is added to the previous view and will appear in the bottom text area. This time the user will not be able to change it directly.
Each time that a new table or view is selected, it will appear in the section “Selected Table/View”, which will list all the table and views that are being currently used in the query. The query will be visible on the bottom.
Here is an example of a union query built with two tables/views:



The single table/view usages can be deleted or modified (affecting only the relative part of the query) by clicking on the buttons on the right of each row. The user can also preview the result of the union with the “Union Preview” button.
In case the SQL selects have different number of columns, the modal will display an alert so that the user can rectify the error. The rows also display the number of columns each selection has, to make it easier to spot the one with the error.



The user can also click on the "Clear" button to remove all the union queries and start from scratch. If the user clicks on "Copy", the query overwrites the main query on the dataset modal.
Back on the main dataset modal, the user can click on "Next" once the query has been written. The second step will show up, asking the user for the name and the description of the dataset:

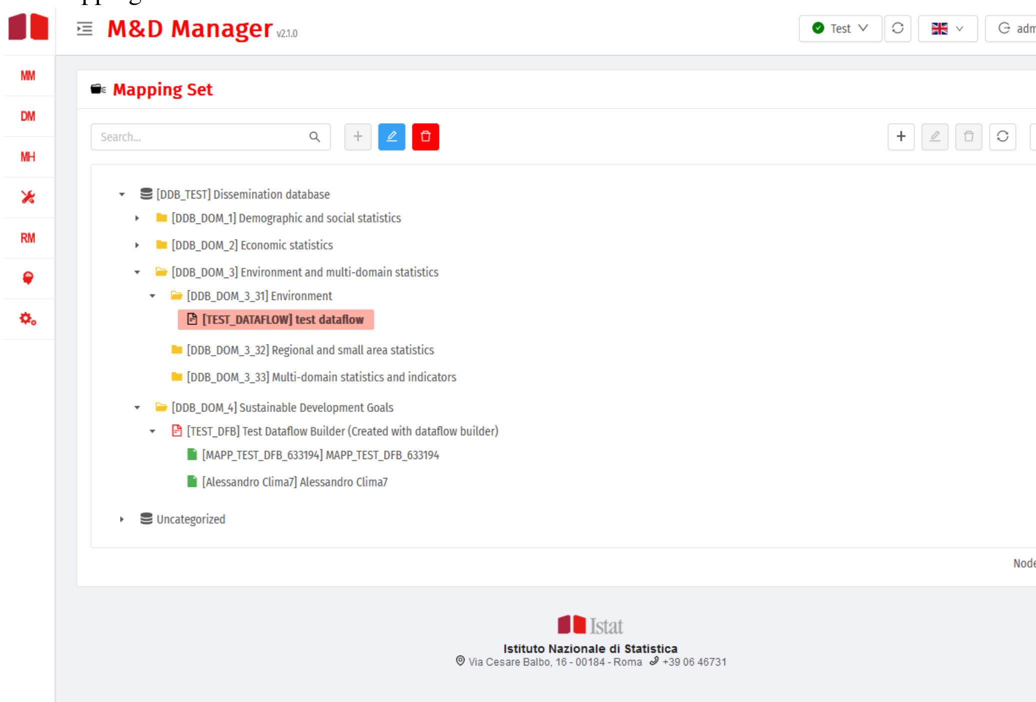


The user must bear in mind that a dataset cannot start with a prefix equal to the "Dataset Prefix" in the *Data Management* section of the Application Settings (which is "DS_" by default). Once the user clicks on "Save", the dataset will be saved under the currently selected connection.

When editing a dataset, changing the name will **clone** the dataset instead of modifying it. When the name stays the same, the dataset is modified.

11.3 Mapping set

The **Mapping Set** function is present in the left-hand side menu starting from the "Mapping Handler" item.
The mapping set feature allows the user to create, modify and delete dataflows under the relative categories and to create, modify and delete the mappings related to the dataflows.

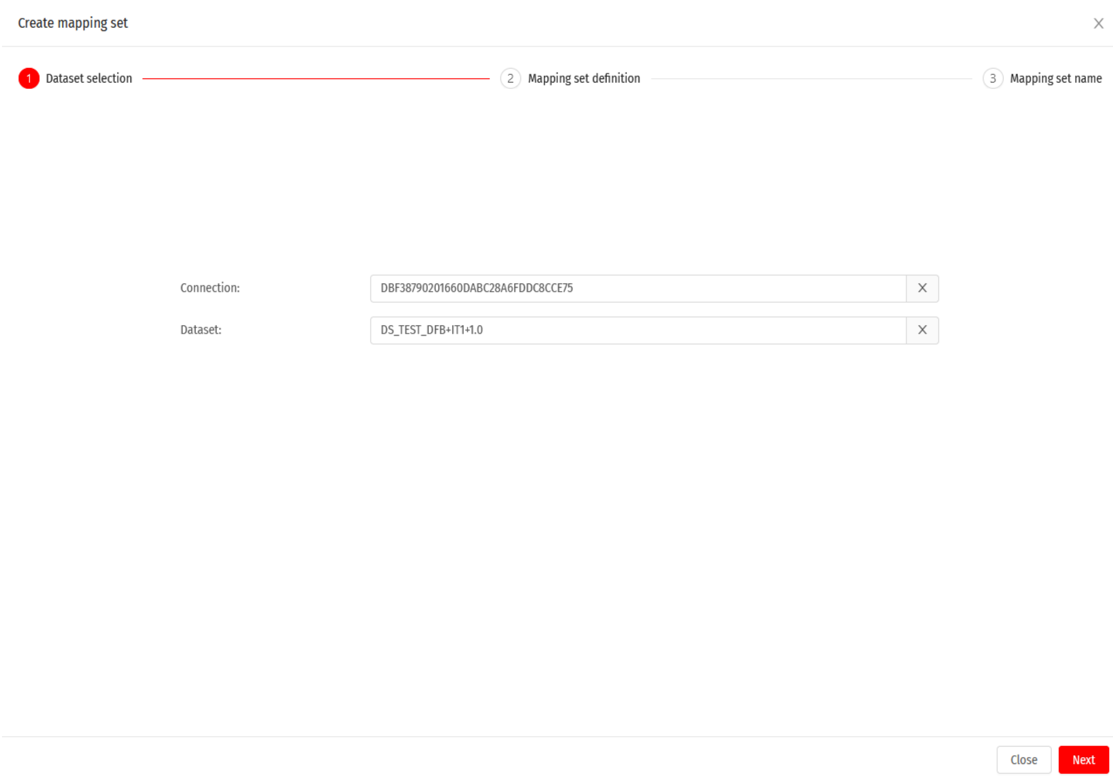


The main interface is as follows:
The categories are shown in the tree as folders. In each folder, dataflows can be created: they have a grey document icon. If the icon is red, it means that the dataflow has been created with the *Dataflow builder*. This means that it can only be changed there, and it will not be editable or deletable here (nor will be its mapping set). The mapping sets have a green document icon and each dataflow can only have one mapping set (and vice-versa, one mapping set cannot be assigned to more than one dataflow).

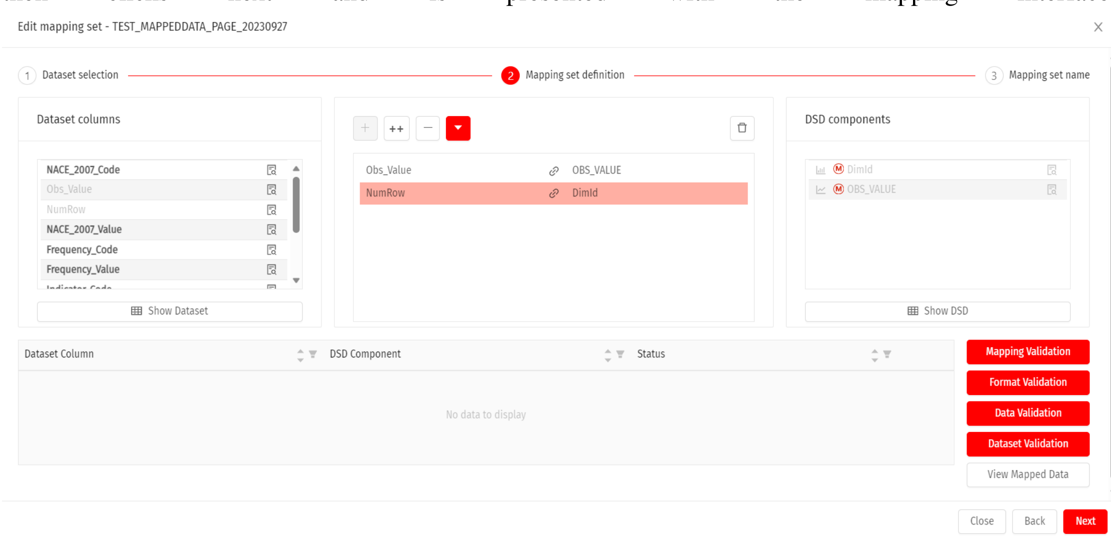
Dataflow actions
The three buttons on the left are used to create, modify or delete dataflows, while the ones on the right do the same actions for the mapping sets. On the left, the first button is enabled when a category is selected, the second and third ones are enabled when a dataflow is selected. If the dataflow has a red icon, the edit icon is only a read-only view and the delete icon gets deactivated. The actions of creating, modifying and deleting a dataflow are the same as in the *MetaManager Dataflows*.

Mapping set actions
On the right, the first button is enabled when a dataflow is selected and only if a mapping set has not yet been defined. The second and third buttons are enabled when a mapping set is selected and only if the dataflow is not from the dataflow builder. The last three icons on the right are used to reload the tree, collapse all the nodes and open all the nodes.

When the user clicks the button to create a mapping set, the following modal pops up:



In the first step, the user selects a connection and a dataset, then clicks next and is presented with the mapping interface:

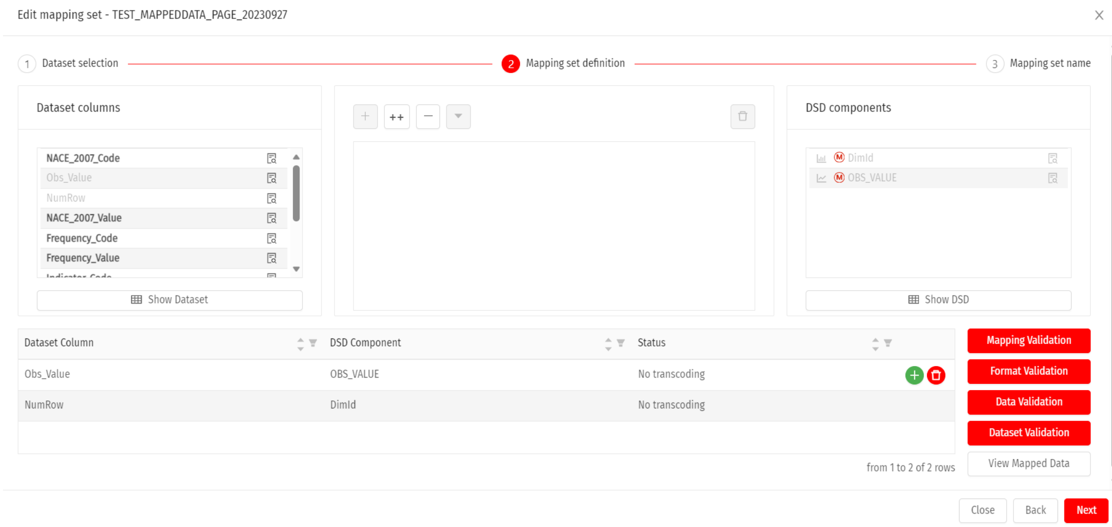


The user can select columns from the dataset on the left and from the dataflow's DSD on the right. By doing so he can perform:

- 0:1 mappings by selecting only a column in the right and then providing a constant value when pressing the "+" button on the center
- 1:1: mappings by selecting one column on the left and one on the right and pressing the "+" button on the center
- N:1 mappings by selecting many columns on the left and one on the right and pressing the "+" button on the center

Note that the dataset mappings can be used more than once, even if they are greyed out.

When the user has all the necessary mappings in the center, it can press the red button with the downwards arrow so that the mappings can be pushed to the bottom table. From there the user can add a transcoding to each mapping. In order to do so, the mapping set must first be saved, so a popup will appear asking the user for the name of the mapping set (if it was not already provided) before letting the user add a transcoding.



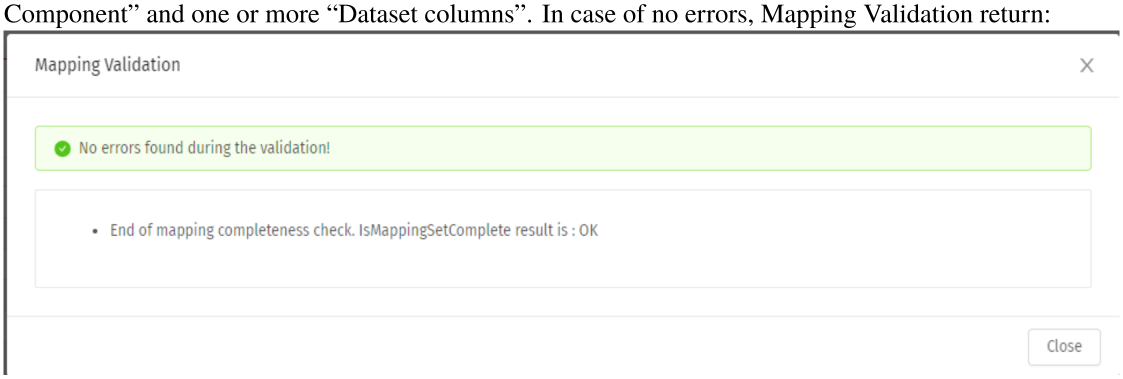
By clicking the green button present in each row, it is possible to create a transcoding for the mapping:

- A Constant transcoding, if the mapping is 0:1
- A Dimension transcoding, if the columns are of type dimension
- A Time Dimension transcoding, if the columns are of type time dimension

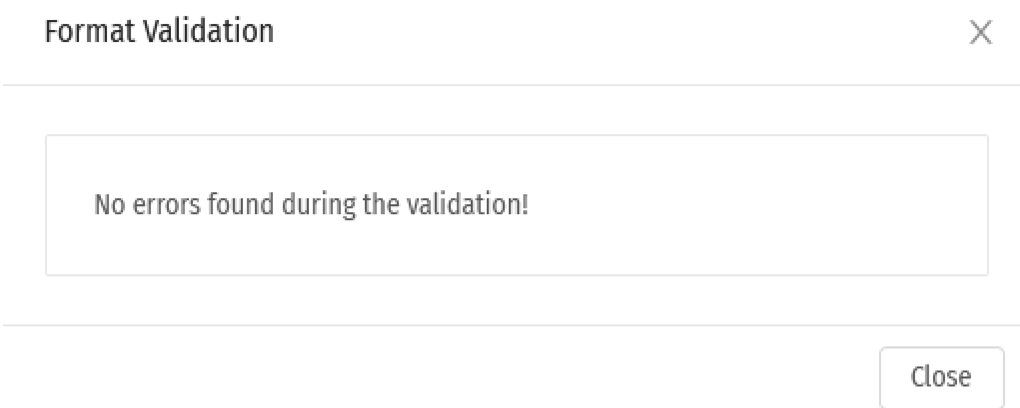
The user can perform 4 types of validation checks:

- Mapping Validation
- Format Validation
- Data Validation
- Dataset Validation

By clicking on “Mapping Validation”, the user can validate the individual association between a “DSD Component” and one or more “Dataset columns”. In case of no errors, Mapping Validation return:

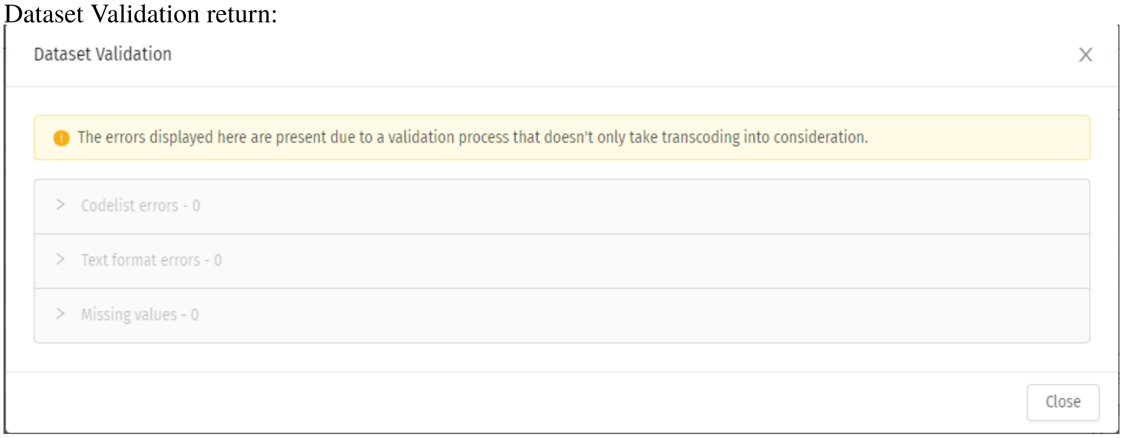


If the user is connected with the Struval library, the ‘Format Validation’ button is enabled. By clicking on it, the user can validate the format of the dataflow. In case of no errors, Format Validation return:

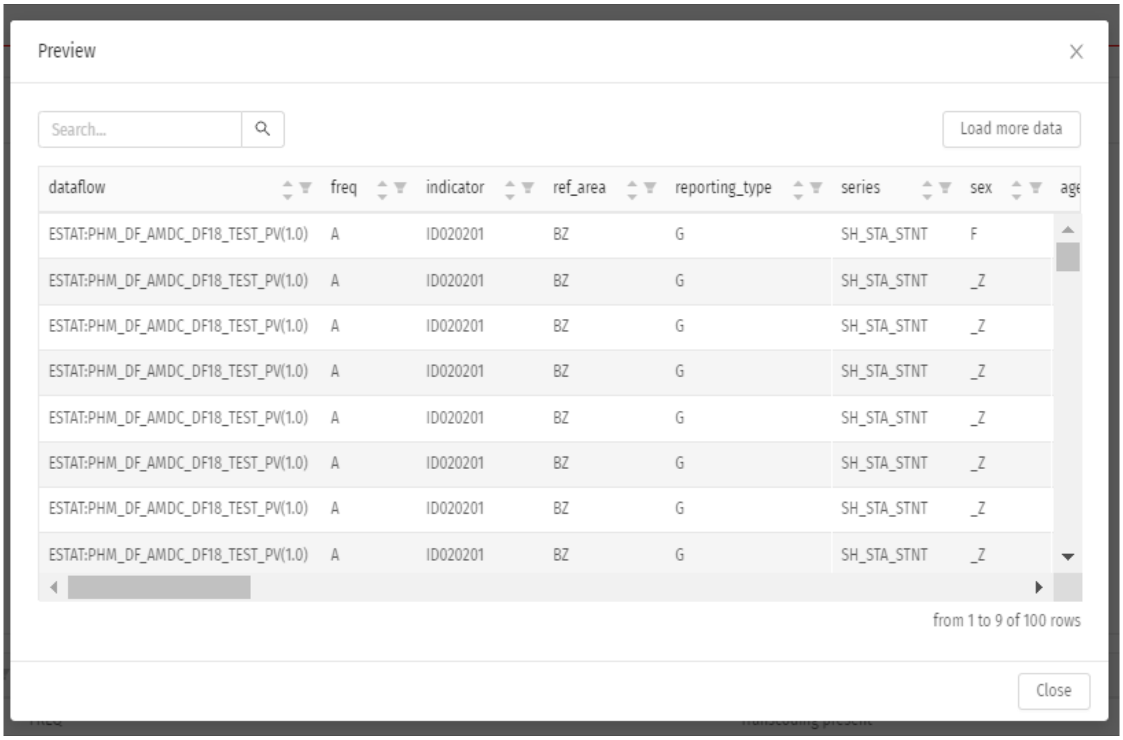


By clicking on “Data Validation”, the user can validate the data of the dataflow. In case of no errors, Data Validation return:

If there are some kind of errors, the user can download them by clicking on the “Download” button.
By clicking on “Dataset Validation”, the user can validate the data of the dataset. In case of no errors, Dataset Validation return:

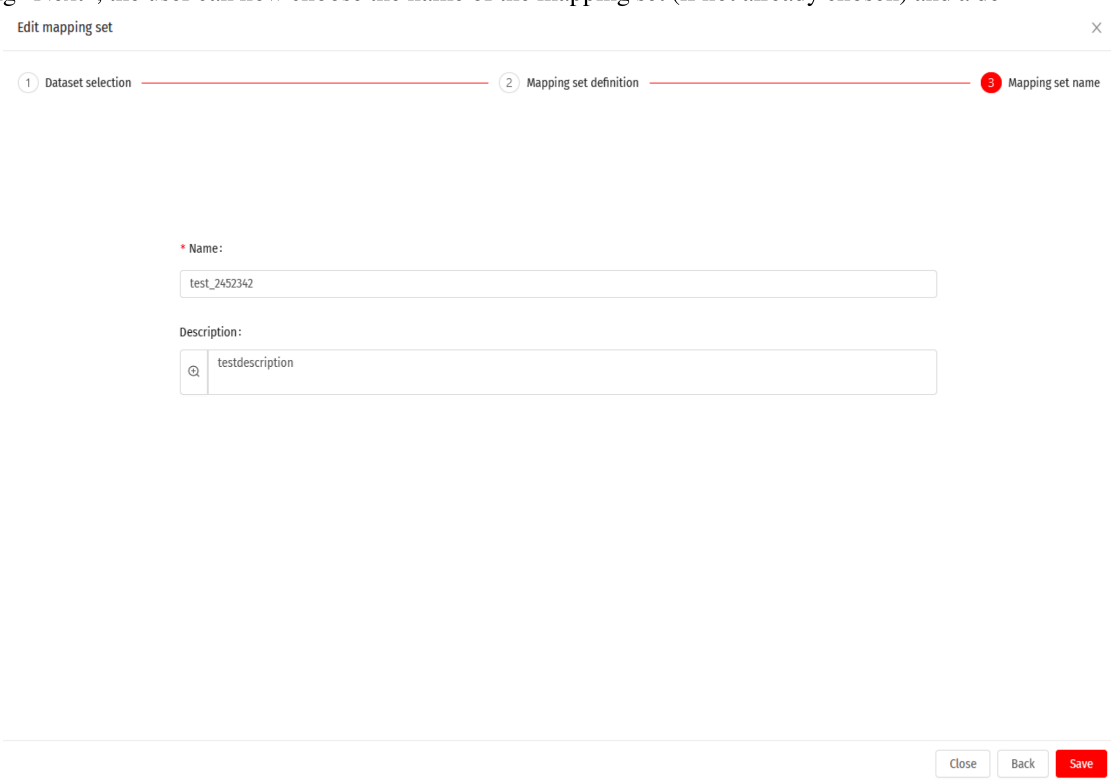


The button “View Mapped Data” provides a preview of the mapped data.
In case there is a need to load more data, the user can click on the “Load more data” button.



All these actions require the user to save the mapping set because the database needs to be in sync with the user interface.
Each time the mapping set is saved, if the validity checks are passed, the dataflow is enriched with a "ReadyForProduction" annotation; otherwise the annotation will be "NotReadyForProduction".

By clicking "Next", the user can now choose the name of the mapping set (if not already chosen) and a description:

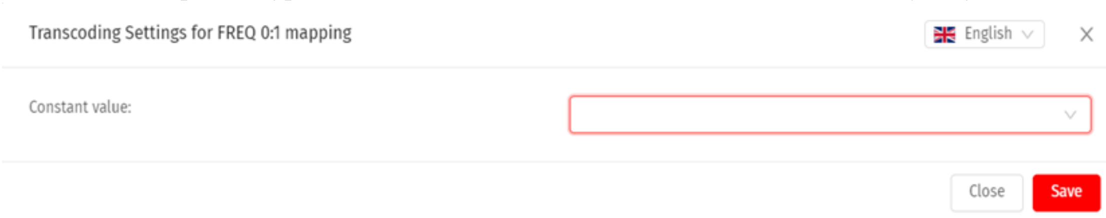


By clicking "Save", the mapping set is saved and the user is returned to the main tree.

11.3.1 Transcoding

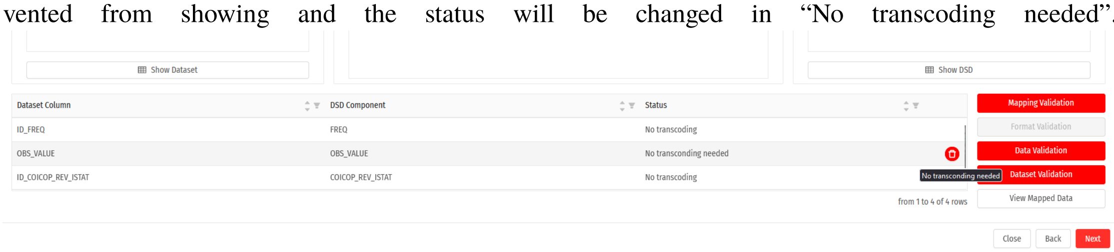
11.3.1.1 0:1 Transcoding

In the case of 0:1 transcoding, when defining the mapping, a popup will open which will allow you to enter a constant value chosen via a select. If the DSD is of a temporal type, the constant value can be defined directly by the user.

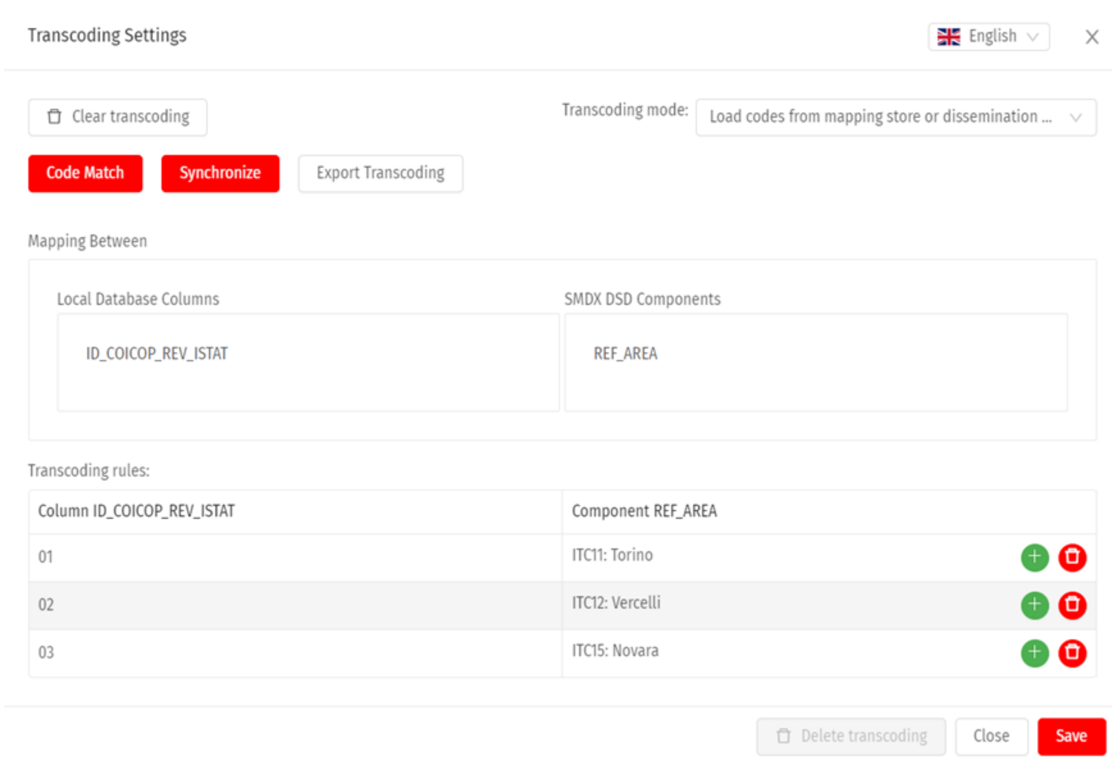


11.3.1.2 Dimension transcoding

If a mapped column is not a time-based dimension and is provided with a codelist, the user is allowed to add a dimension transcoding by clicking on the green "plus" button. In case the selected column does not have a codelist, the green button is prevented from showing and the status will be changed in "No transcoding needed".



In the dimension transcoding modal the user can add or modify existing dimension transcoding.



This window's features are listed below:

Transcoding Mode:

The Transcoding Mode select contains two options:

- "Load codes from mapping store or dissemination database" that allows the user to select a component from the "Transcoding Rules" table
- "Import transcoding rules from csv file" displays a file input through which the user has the possibility to import the component from a csv file. Once the file has been selected, the import button is enabled.

Export Transcoding:

By clicking the "Export Transcoding" button it is possible to export the transcoding rules into a csv file.

Clear Transcoding Button:

This button clears the component cell content.

Code Match:

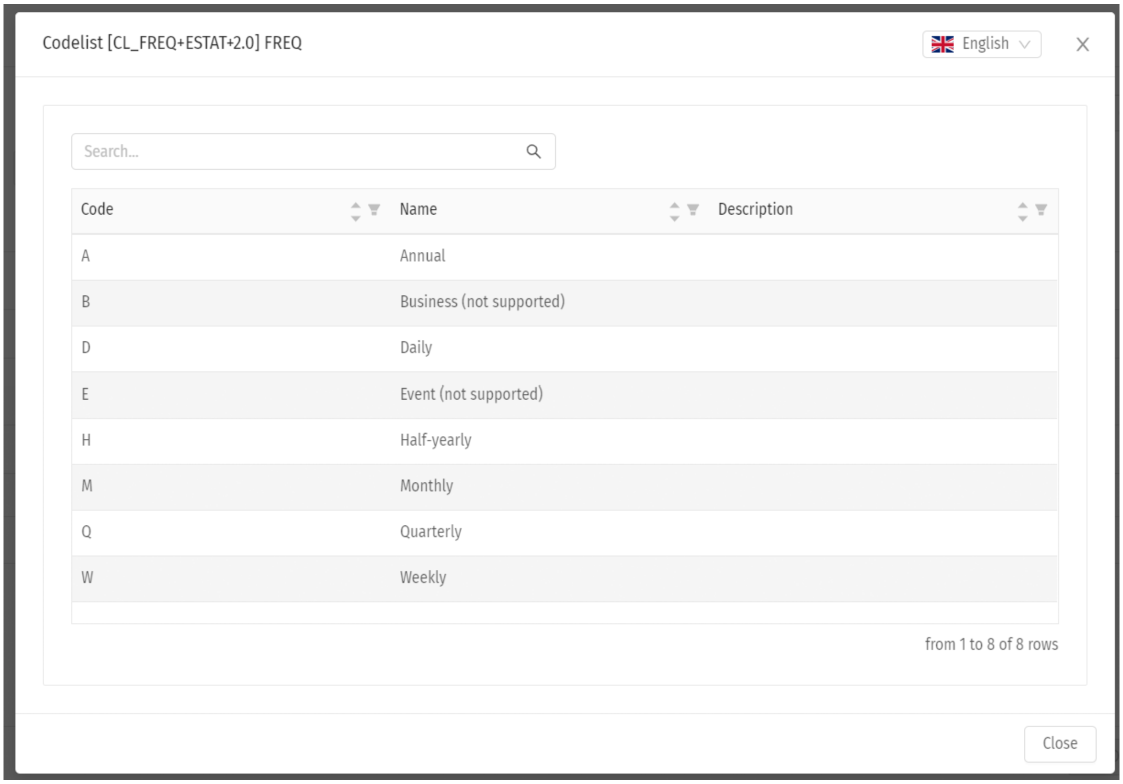
This button makes a combination of dataset column's values with compatible component codelist's values. It is only enabled when there is one selected dataset column, otherwise it is disabled.

Synchronize:

This button is used to reload the codelist to have an updated list.

Transcoding Rules:

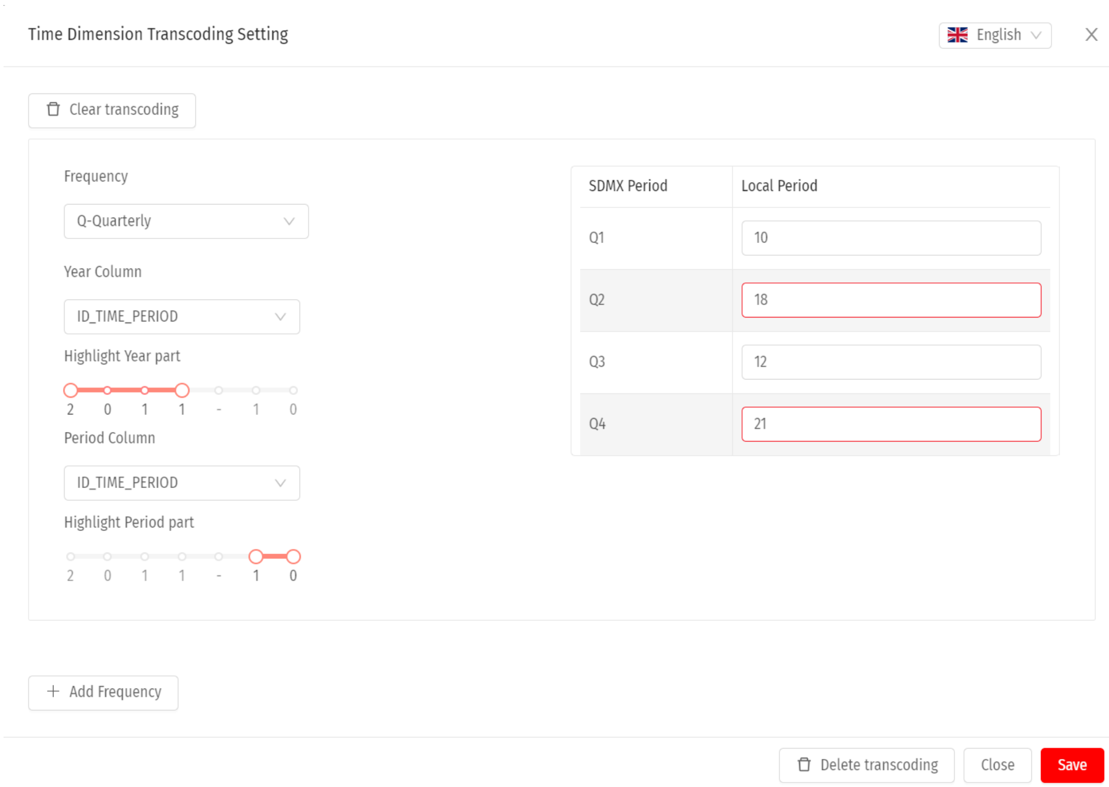
In this section the user can configure the transcoding rules for each value of the selected column. The table is made of rows which are a product of all dataset columns selected by the user. For each product the user is able to choose a component by clicking on the green button. That button opens a new modal which displays a list of codelist related to the component. The user can select one of them to associate it with the dataset column(s).



The chosen codelist will appear in the component cell. If the user wants to delete the component associated with the dataset column(s), he can click on the red button.

11.3.1.3 Time Dimension Transcoding

If a mapped column is a time-based dimension the user is allowed to add a time dimension transcoding. This window allows users to configure transcoding rules for temporal dimensions of a dataset. Temporal dimensions can be associated with various frequencies (e.g., yearly, half-yearly, quarterly or monthly) and can be configured to include year and period columns. Below there's a list of the features of the window and how to interact with them:



Frequency Selection:

- Use the dropdown menu to select the desired frequency for the temporal dimension (annual, semi-annual, quarterly, monthly).
- Frequency selection is mandatory.

Year and Period Column Selection:

- After selecting the frequency, you can choose the corresponding year and period columns from the dropdown menu.
- Selection of the year column is mandatory.
- Selection of the period column is mandatory only if the frequency is not "Annual."

Setting Temporal Bounds:

Sliders are interactive tools that allow to define desired time periods. Their configuration depends on the selected time frequency and chosen time columns.

- **Set Year Sliders:** Allows to define temporal bounds of interest using sliders based on the value contained in the selected annual column.
- **Set Period Sliders:** If the chosen frequency is different from "Yearly", it allows to define intervals of months or other time periods using sliders based on the value contained in the selected annual column.

Temporal Data Table:

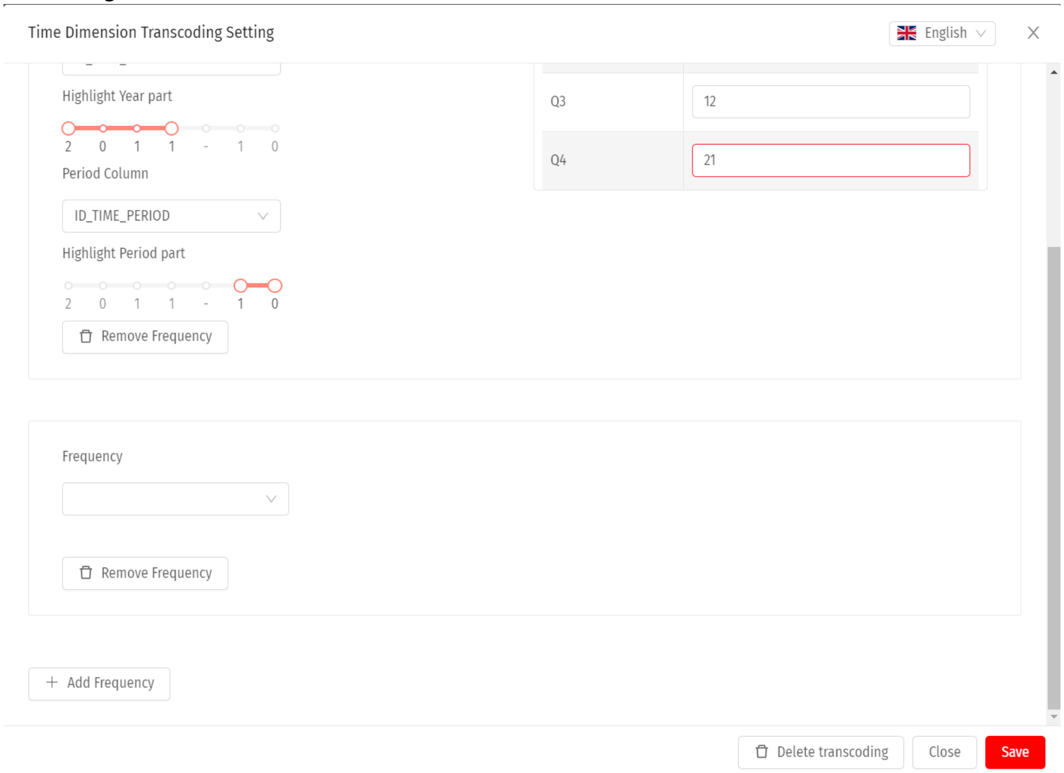
- If the selected frequency is different from "A-Yearly", a table of temporal data will be available next the frequency settings.
- You can input values into the "Local Period" column of this table. Input fields will be highlighted in red if they do not adhere to the temporal intervals defined by the sliders or if the entered value is not present in the selected column. If these rules are not met, transcoding can still be saved.

Field Validation:

- Fields will be validated to ensure that the frequency, year, and period columns are correctly selected, and that values in the temporal data table adhere to the defined temporal intervals.

Managing Multiple Frequencies:

- You can add multiple temporal transcoding rules by clicking the “Add Frequency” button. A new tab will appear below the previously compiled frequency.
- You can select only one frequency from the following options: yearly, half-yearly, quarterly, or monthly, with a maximum of four selections based on the number of options available in the “frequency” dropdown menu.
- You can remove a frequency by clicking the “Remove Frequency” button associated with that frequency.
- Before removing the frequency, the system will provide a confirmation message to the user.
- All changes will not take effect on the database until the user clicks the “Save” button.



Saving Settings:

- After configuring the temporal transcoding rules, click the “Save” button to confirm the settings.
- Before saving, the system will provide a confirmation message to the user.
- Ensure that all dropdown menu and slider values are valid before saving, or else an error message will be displayed.

Clearing Transcoding Settings:

- You can reset all current settings by clicking the “Clear” button.
- This action will restore default settings.
- Before clearing, the system will provide a confirmation message to the user.

- Changes will not take effect until the user saves them.

Deleting Transcoding:

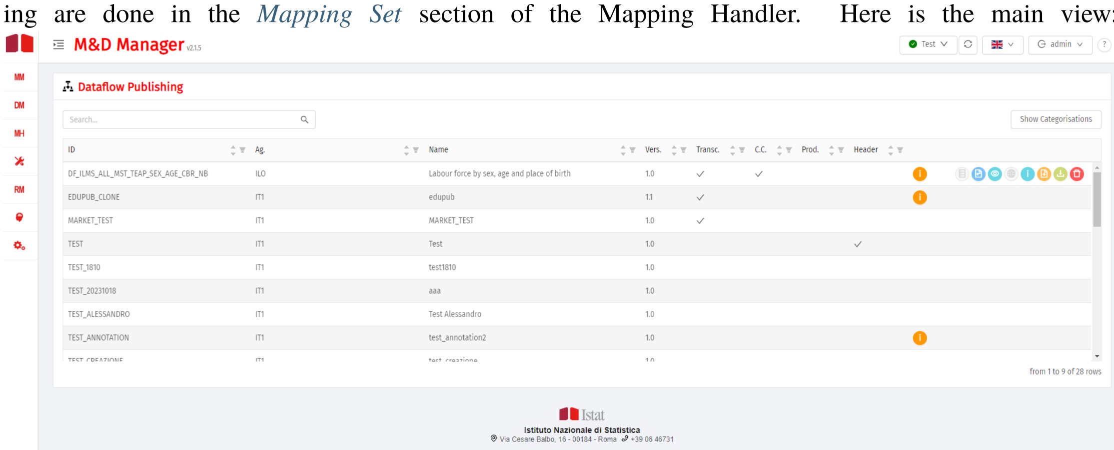
- You can delete the selected temporal transcoding by clicking the "Delete Transcoding" button.
- Before permanently deleting the transcoding, a confirmation message will be provided to the user.

Closing the Form:

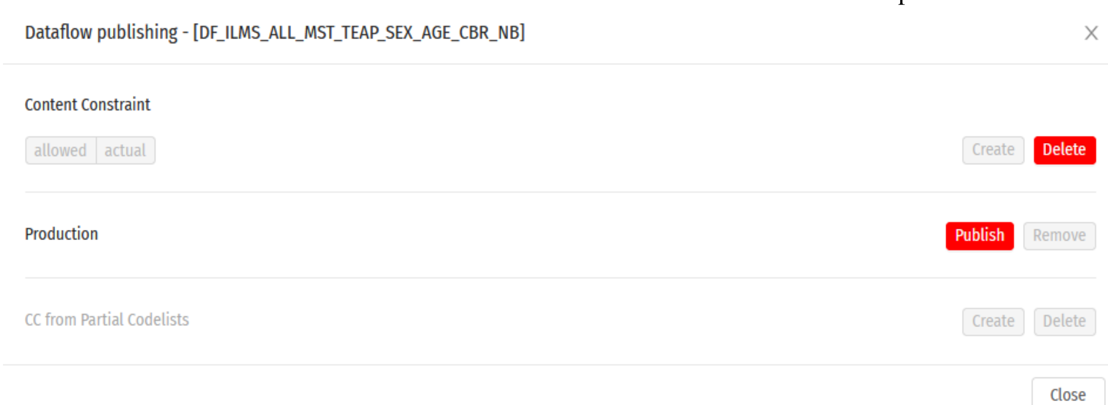
- Click "Close" or "X" button to exit the form without saving any changes.

11.4 Dataflow publishing

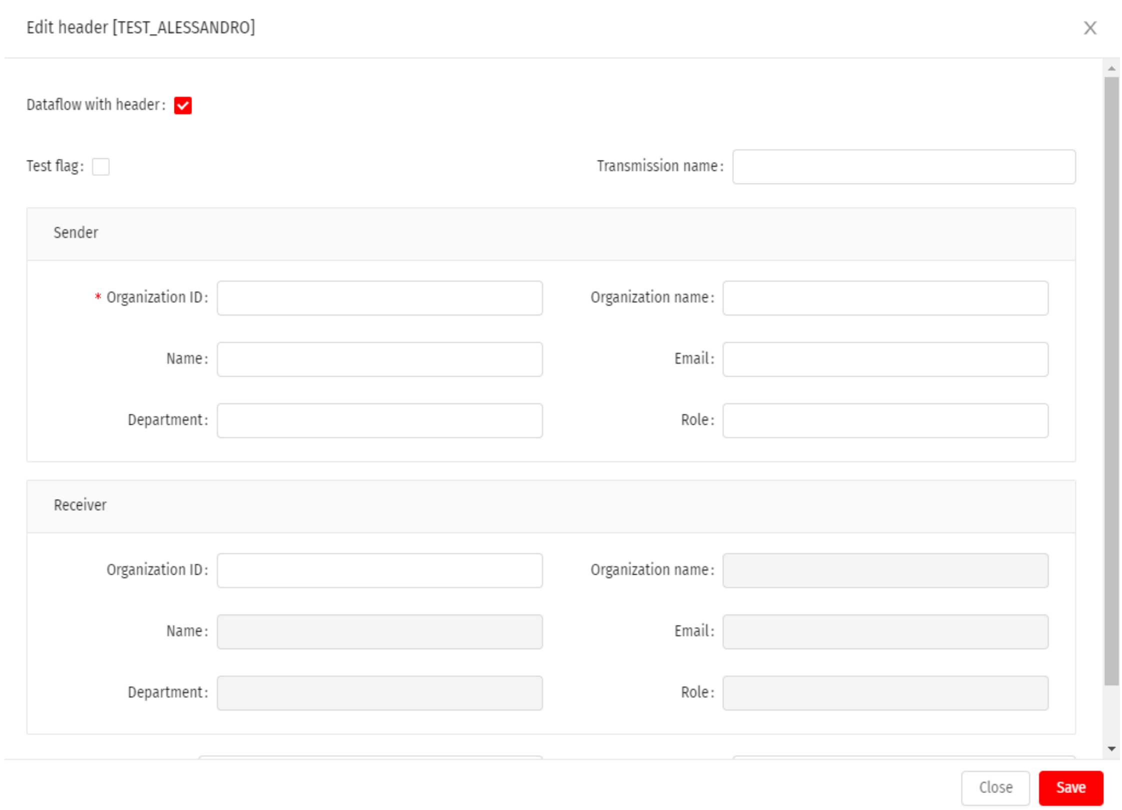
The **Dataflow Publishing** function is present in the left-hand side menu starting from the "Mapping Handler" item. The appearance of the page and the actions that can be taken on each row are similar to the *Dataflow builder* page, with the exception of the header action.
The publishing action is present only when the dataflow is ready, which means that it has a complete mapping set and transcoding and also has the "Ready for production" annotation. The publishing step here only have three sections instead of five, since the mapping and transcoding are done in the *Mapping Set* section of the Mapping Handler. Here is the main view:



The publishing modal lets the user create Content Contraints, publish the modal and create content constraints from partial codelists:

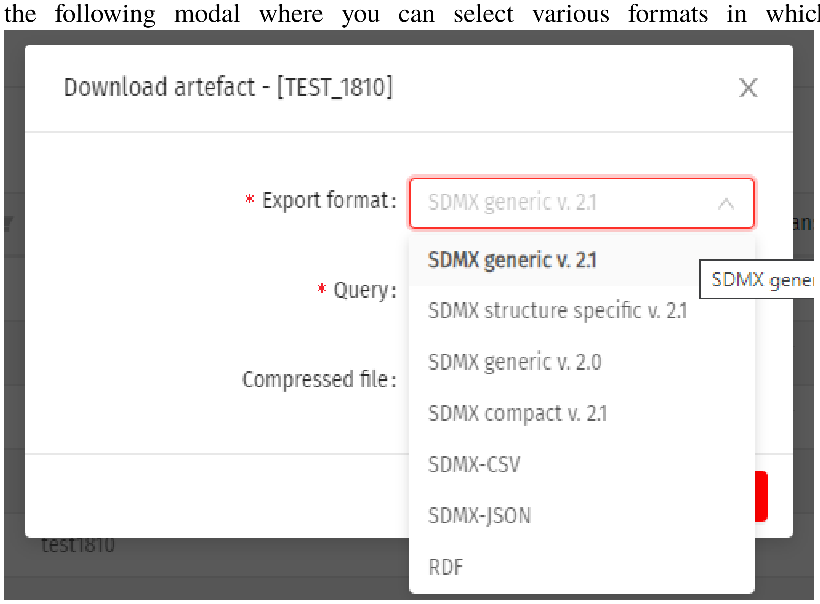


Clicking on the "Edit Header" button to create or edit the header of the related Dataflow through the follow-

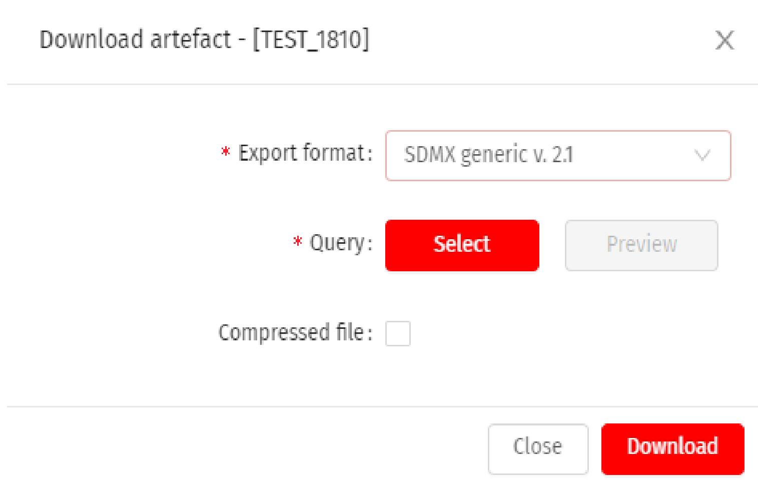


ing mask

Once the mask for entering the header is open, the user must flag the checkbox "Dataflow with header" to be able to fill in the form for entering the dataflow header. If a header is to be removed, simply deflate the checkbox. Clicking on the "Download" button opens the following modal where you can select various formats in which to export the Dataflow



The "Select" button allows you to write a query to filter the Dataflow data by displaying a preview of it


Download artefact - [TEST_1810]
* Export format:
SDMX generic v. 2.1
* Query:
Select
Preview
Compressed file:
Close
Download