Data Browser - User Manual

Release 1.3.3

ISTAT

Nov 24, 2021

CONTENTS

CHAPTER

ONE

ACRONYM AND GLOSSARY

.NET Core

Free and open source software development framework for different operating systems: Microsoft Windows, MacOS e Linux

IIS

Internet Information Services

Data Browser Hub WS

Data Browser Hub Web Services

CHAPTER

TWO

ABOUT

This User Manual describes the Data Browser project giving an overview of its functionalities from a technical and practical point of view.

All sections fully describe configuration, management and organization of each single component in order to make it easy for the reader to understand and know how to directly move inside the system.

2.1 License

European Union Public Licence V. 1.1

2.2 Product overview

The **Data Browser** project is a web portal for sharing, integrating and disseminating macro-data produced by Sistan or other statistical agencies which fulfill functions or services of public interest. The system implements a distributed data warehouse based on the **SDMX** standard ***(ISO IS-17369)*** which can be freely queried by external users via a web interface.

The strategic aim of this innovation is to:

- create a "network" of distributed databases;
- integrate datasets with the ones already available from the agencies;
- make sure that the created network contains statistical data of good quality;
- combine data and meta-data with a view to semantic interoperability;
- share international best practices on statistical dissemination systems.

The **"Hub"** architecture is based on the fact that the dissemination of data is carried out through the nodes of the system, each of which is managed by every individual entity participating in the network. Data publication on a system node, implies that the information entered is now available inside the network and easy to browse in the Hub.

The hub is, therefore, the only point from which data can be queried.

To get more specific about the application, from a more technical and implementation point of view, the Data Browser project is the result of the combination of the following functional components:

1. Administration component
2. Browsing component

3. Data Visualization component
4. Sharing component
5. Search component

This User Manual gives an overview of the functionalities of the Data Browser and explains in detail how to configure and manage the application and the nodes (Administration) and how to visualize, search and share data (Data Browsing) .

CHAPTER

THREE

INSTALLATION

In this section we will show the steps needed in order to install the application.

3.1 Pre-requisites

Operating system

The supported operating systems are the same ones supported by .NET Core. In the following table, the Microsoft Operating System supported are listed.

Windows Client	7 SP1+, 8.1	x64, x86
Windows 10 Client	Version 1607+	x64, x86
Windows Server	2008 R2 SP1+	x64, x86

IIS

IIS has to be installed in a version supported by the used Windows operating system. Make sure that the *.json* MIME type is available, by performing the following steps:

- click on the IIS Web Site under which the application has to be installed;
- double click the *MIME Types*;
- ensure that **.json** and **application/json** mime types are present;
- if they are missing, add them by right clicking and selecting "Add".

.NET Core

The ***.Net Core Framework ver.3.1.x*** has to be installed together with ***.NET Core hosting bundle for IIS***, in compliance with the necessary requirements as specified in https://docs.microsoft.com/en-us/dotnet/core/install/dependencies?pivots=os-windows&tabs=netcore31.
In order to check if these modules have been already installed, just access: *Control Panel/Programs/Programs and functions.*

If .NET Core has not been already installed, it is possible to proceed as follows: 1. Download the package from the following URL: https://dotnet.microsoft.com/download/dotnet-core/3.1 the last version of 3.1.x Hosting Bundle in the section ASP.NET Core Runtime 3.1.x.

Warning! The installation of the *Microsoft Visual C++ 2015 Redistributable Update 3* or higher is needed.
Check in **Control panel/Programs/Programs and functions** if it exists or, alternatively, download and install the package from: https://www.microsoft.com/en-us/download/details.aspx?id=52685.

SSL CERTIFICATE

In order to publish the web services in https, an SSL certificate is needed.
The instructions for creating such certificate depend on the certificate type and on the IIS version.
For IIS ver.10, instructions are available and can be followed at: https://www.digicert.com/csr-creation-ssl-installation-iis-10.htm.

3.2 Prevent IIS Tilde Enumeration

The IIS Tilde vulnerability consists in the threat caused by the tilde character. It could happen that a remote hacker discloses files and folder names because of leakage of elements containing sensitive data (such as credentials, configuration files, maintenance scripts, etc..) as a result of exploiting this vulnerability. However, some simple steps could defend againsts these attacks. Instructions to prevent IIS Tilde Enumeration are as follows.

On the server that hosts IIS, considering for example that installation is been made under the *wwwroot* folder:

- open command prompt with administrator permissions
- disable creation of file names in 8.3 format with the command:
  - *fsutil 8dot3name set 1*
- remove all the names in the 8.3 format present in folders and subfolders (/s) concerned.
  - *fsutil 8dot3name strip /s C:\inetpub\wwwroot* (WARNING: don't run the command on the whole filesystem e.g.: C:, you may have unpredictable side effects on already installed programs)
- verify that all 8.3 filenames have been deleted
  - *fsutil 8dot3name scan /s C:\inetpub\wwwroot*

scan and strip parameters generate a log file (the path is visible in the command output) in which is possible to check concerned files and possible problems (e.g. files not processed due to lock by other applications).

3.3 Software package

PACKAGE DESCRIPTION

The software package ***databrowser_x.x.zip*** contains two folders ***app_first_installation*** and ***app_upgrade***.

Both contain the following folders:

- databrowser: configuration for client side.
- databrowserhub: configuration for server side.

app_first_installation is used for the first installation and includes: configuration files, customizations and application files.

app_upgrade is used to upgrade an already configured application. In this case, the package contains also translations but configurations are removed. If translations have been previously customized, related files must be manually overwritten in the folder i18n (for example the it.json and en.json files).

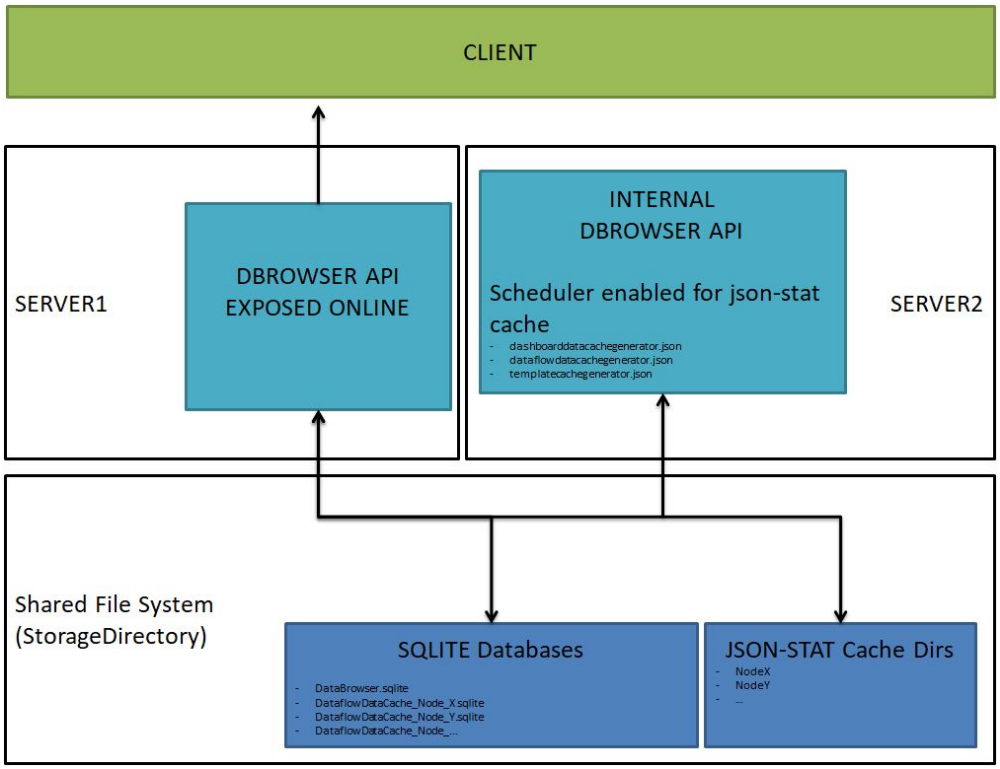
3.4 Configurations

The client application is already configured to communicate to the web service listening on http://localhost/databrowserhub/. The endpoint address can be set in the path [**ROOT_IIS**]\databrowser\config.json

The *json* file appsettings.json, included in the databrowserhub folder, contains all the information regarding paths and settings of the application. Let's take a closer look to all the possibile configurations.

Storage Directory

It is possible to configure the path of the base directory dedicated to data (databases and cache files (json-stat format)); this will potentially allow 2 installations of Data Browser, one dedicated to front-end and one dedicated to cached data update, that share database and cache files.



This configuration is optional: if no base directory for storage is set, the software will automatically create a directory in the installation root directory.

```
"StorageDirectory": "C:/StorageDirectory"
```

DataflowDataCache

In this part of the configuration, the user defines all settings regarding the storage of the cache files (json-stat format) for dataflow data.

```
"DataflowDataCache": {
  "Type": "NoSql",
  "ConnectionString": "Data Source=DB/DataflowDataCache.sqlite;",
  "IsEnable": true,
```

(continues on next page)

(continued from previous page)

```
  "SaveDataOnFile": true,
  "SavedDataFilePath": "_DataflowDataFiles",
  "Expiration": 604800 //Second
},
```

Specifically:

- *Type* is the server storage's type (supports only "NoSql").
- *ConnectionString* specifies the path of the sqlite database.
- *IsEnable (true/false)* indicates if the Data Browser can use (or not use) the cache.
- *SaveDataOnFile* specifies if the jsonstat will be saved in a separated file.
- *Expiration* indicates the default value, in seconds, for the validity of the jsonstat in cache. If value is 0 (zero), cache is never generated. If the value is -1 cache never expires.
  At application level, it is possible to disable cache by setting the flag IsEnable = false. By default Expiration is always set to -1

CatalogCache

In this part of the configuration, the user defines expiration for catalogs.

```
"CatalogCache": {
    "Expiration": -1
  },
```

It indicates the default value, in seconds, for the validity of the jsonstat in cache. If value is 0 (zero), cache is never generated. If the value is -1 cache never expires.
By default Expiration is always set to -1.

Database

In this part of the configuration, the user defines all settings necessary for the storage of information regarding the Data Browser (nodes, dashboards, views, templates, users and so on).

```
"Database": {
  "DbType": "SQLite",
  "ConnectionString": "Data Source=DB/DataBrowserDB.sqlite;",
  "UseMigrationScript": true
},
```

Specifically:

- *DBType* is the server storage's type.
- *ConnectionString* is the path of the sqlite database concerning the storage directory.
- *UseMigrationScript true/false* indicates if the new version of the software can automatically update the used database (recommended value is true).

Geometry Database

In this part of the configuration, the user defines the path of the database needed for the default geometries.
In the package provided, for example purposes, there are geographical data referring to Eurostat NUTS and ISTAT Italian municipalities.

```
"GeometryDatabase": {
  "DbType": "SQLite",
  "ConnectionString": "Data Source=DB/Geometry.sqlite;"
},
```

It is also possible to customize the geometries that overwrite the default configuration, simply by adding in the same directory some databases in the same format as "GeometryDatabase", whose name must respect the following format: Geometry_Node_{NodeID}.sqlite

The geographical database in sqlite is structured through a dedicated database, containing the table "Geometry" with the following fields:

- *UniqueId*: unique numeric identifier (mandatory)
- *Id*: unique identifier, used to make the join with codelist codes (mandatory)
- *Label*: territory label; this information is optional and not used by the application
- *Country*: country identifier; this information is optional and not used by the application
- *NutsLevel*: NUTS level; this level is used by the application. It is a string which can be translated by the client, whose key must be in the format "nutsLevel{NutsLevel}". Here a possible example:

```
        .
        .
        .
"nutsLevel0": "Country",
"nutsLevel1": "Territorial divisions",
"nutsLevel2": "Regions",
"nutsLevel3": "Province",
"nutsLevel4": "Municipality",
        .
        .
        .
```

- *Source*: data source; this information is optional and not used by the application
- *WKT*: territory geometry in WKT format; for more details about the format see: https://en.wikipedia.org/wiki/Well-known_text_representation_of_geometry
- *AlternativeIds*: alternative identifiers for the current territory. In case the same territory is present in the codelist with different identifiers, in order to avoid adding more rows with the same geometry, in this field it is possible to insert the concatenation of the identifiers so that the system can receive them.

If the user needs to specify more than one field, the "pipe" | separator must be inserted between the values.

CORS policies

In this part of the configuration, the user decides if CORS policies must be enabled or not (for more information on CORS check https://en.wikipedia.org/wiki/Cross-origin_resource_sharing).

```
"General": {
  "CORS": {
    "Enable": true
  },
```

External and internal rules

```
"EndPointResponseLogForDebug": false,
```

If true and log level is "debug", all responses from NSI will be saved on log file.

```
"InternalRestUrl": "",
```

Specifies the URL used for calling the DataBrowserAPI from the installation server.

```
"ExternalRestUrl": "http://localhost/databrowser/api/",
```

Specifies the URL used for calling the DataBrowserAPI from outside the installation server.

```
"ExternalClientUrl": http://localhost/databrowser
```

Specifies the URL used for calling the Data Browser frontend from outside the installation server.

Authentication

Some functionalities, exposed via API, are accessible by all types of users (anonymous and authenticated), others, on the other hand, are subject to profiling. In this part of the configuration, settings regarding login for authenticated users are defined.

```
"Authentication": {
  "IsActive": true,
  "Key": "8CF07358F9BB4CA98C0EE4D26A97858C",
  "Issuer": "DataBrowserIssuerApi",
  "Audience": "DataBrowserApiUser",
  "JwtTokenLifeTime": 15, //Minute
  "EnableRefreshToken": true,
  "EnableAuditLogin": false,
  "TryLoginMax": 3,
  "TryLoginTime": 60,
  "DelayLogin": 3000,
```

Specifically:

- *IsActive* if set to false, all APIs will be accessible from anonymous users.
- *Key* represents the symmetric security key which is used for the generation of the hash token needed for login. The usage of the key is necessary for security encryption.
- *Issuer* identifies the principal that issued the token.
- *Audience* identifies the recipients that the token is intended for. If the principal processing the claim does not identify itself with a value in the "audience" claim when this claim is present, then the JWT MUST be rejected.
- *JwtTokenLifeTime* specifies the time to live of the generated token.
- *EnableRefreshToken* tells the system to regenerate a new token from cookie instead of recalling token with user and password.
- *EnableAuditLogin* saves all attempts to login by all user on database. Information are saved in the AspNetUsersAudit table of the applicative database.
- *TryLoginMax* sets the maximum number of login before activating the system of delay response.
- *TryLoginTime* sets time (minutes) in which incorrect logins are considered.

- *DelayLogin* (ms) specifes the time of delay of login response (time is incremented by this value for each invalid login).

Refresh cookie options

In this part of the configuration, the user defines the cookie's settings to refresh the token.

```
"RefreshCookieOptions": {
  "HttpOnly": true,
  "SameSite": "Lax", //null, "Unspecified", "None", "Lax", "Strict"
  "RefreshTokenLifeTime": 14400 //Minute
```

Specifically:

- *HttpOnly* is recommended to be always set to true.
- *SameSite:* *Lax* is recommended. If it is necessary to use crossdomain cookie, set it to "None" and use an Https connection. For all possibile values of the SameSite parameter, check https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Set-Cookie/SameSite
- *RefreshTokenLifeTime* defines validation time (in minutes) of the cookie.

User policy for password

In this part of the configuration, the user defines settings for password policy and mail in order to enable the user to create or change password.

```
"UserPolicy": {
  "PasswordRequiredLength": 8,
  "PasswordRequireNonAlphanumeric": true,
  "PasswordRequireLowercase": true,
  "PasswordRequireUppercase": true,
  "PasswordRequireDigit": true
}
```

For the creation/change password configuration, it is necessary to set SMTP information important for email exchange from DataBrowserAPI.

```
"Mail": {
  "Smtp": {
    "Host": "",
    "Port": 1234,
    "Secure": false,
    "Username": "user",
    "Password": "pass"
},
"DefaultMail": "info@databrowser",
"Templates": {
  "ResetPassword": {
  "Sender": "",
  "Subject": "Recovery Password",
  "Message": "config/Template/recoveryPassword.html"
}
```

In particular:

- *Mail* contains the information needed to configure the SMTP server for email exchange in case of recovery/change password necessity. In particular, the "Secure" field specifies if there is a SSL protocol or not.

- *DefaultMail* is the default sender's email address.
- *Template* contains all html templates used for sending email. The only available template is ResetPassword. In this section, "Subject" is the subject of the email sent to reset the password. "Message" is the body of the email sent in html format. Usually the path to the html is specified. In this case the folder must be inside of DataBrowserAPI folder. This message can be overwritten by UserLang used from request. If the folder contains the config/Template/recoveryPassword.{UserLang}.html, that will be used as file for the message. The file is in HTML/CSS format and it can be modified as needed from the one delivered in the installation package.

Swagger

In case this option is set to true the user can see the requests defined in Data Browser and that can be sent to the application. If this option is missing or set to false swagger is disabled.

```
"Swagger": true,
```

Special cache management

When talking about *cache* we refer to data stored so that future requests for that data can be served faster; the data stored in a cache might be the result of an earlier computation or a copy of data stored elsewhere. In this application, cache is very important especially when considering filterable dashboards that hold territorial dimensions and allow users to query results by changing these dimensions. For this reason timing is very important and a key component for a responsive and functional application.
In order to manage these requests, we consider services for handling cache regeneration depending on whether we are considering filterable or non-filterable dashboards (see section *Dashboards* for more information).
In particular, in the appsettings.json file a scheduler is defined in which a timer is set to specify the interval to check if there is any work to do (usually five minutes is the recommended interval 00:05:00). This scheduler picks the four files present in the config folder (config\dashboarddatacachegenerator.json, config\dataflowdatacachegenerator.json, config\cataloginmemorycachegenerator.json, config\templatecachegenerator.json) and launchs services.

```
"Scheduler": {
  "IsEnable": true,
  "Timer": "00:05:00",
}
```

In particular:

DashboardDataCacheGenerator refreshes all dataflows assigned to static not-filterable views in a dashboard.

The service settings are specified in file: config\dashboarddatacachegenerator.json which contains the following lines:

```
{
  "DashboardDataCacheGenerator": {
    "IsEnable": true,
    "StartTime": "20:30:00",
    "Days": [ 0, 1, 2, 3, 4, 5, 6],
  }
}
```

In particular:

- *IsEnable* is true if the service is enabled.
- *StartTime* defines the starting time.
- *Days* ndicates the day the worker runs (0 = Sunday and 6 = Monday).

CatalogInMemoryCacheGenerator refreshes cache's catalog for all active nodes.

The service settings are specified in file: config\cataloginmemorycachegenerator.json which contains the following lines:

```
{
  "CatalogInMemoryCacheGenerator": {
    "IsEnable": true,
    "StartTime": "10:15:00",
    "Days": [ 0, 1, 2, 3, 4, 5, 6 ]
  }
}
```

In particular:

- *IsEnable* is true if the service is enabled.
- *StartTime* defines the starting time.
- *Days* ndicates the day the worker runs (0 = Sunday and 6 = Monday).

TemplateCacheGenerator regenerate the cache on the data for all templates present.

The service settings are specified in file: config\templatecachegenerator.json which contains the following lines:

```
{
  "TemplateCacheGenerator": {
    "IsEnable": false,
    "StartTime": "13:20:00",
    "Days": [ 0, 1, 2, 3, 4, 5, 6 ]
  }
}
```

In particular:

- *IsEnable* is true if the service is enabled.
- *StartTime* defines the starting time.
- *Days* ndicates the day the worker runs (0 = Sunday and 6 = Monday).

DataflowDataCacheGenerator refreshes all dataflows configured in the DataflowDataCache section config.

The service settings are specified in file: config\dataflowdatacachegenerator.json which contains the following lines:

```
{
  "DataflowDataCacheGenerator": {
    "IsEnable": true,
    "StartTimer": "20:30:00",
    "Days": [ 0, 1, 2, 3, 4, 5, 6],
    "DataflowsRefresh": [
      {
        "Id": "Agency+Id+Version",
        "Dimensions": [ "DimensionId" ],
```

(continues on next page)

(continued from previous page)

```
        "NodeCode": "NodeId",
        "GruopByNumber": {
          "GroupSize": 20
        }
      }
   }
}
```

In particular:

- *DataflowsRefresh* is an array that contains the list of all dataflows that need to be refreshed in cache.
- *Dimensions* can have only one value and defines the dimension's ID to refresh.
- *NodeCode* defines the node's ID that contains the dataflowid to refresh.
- *GroupByNumber* contains the configuration for grouping the number of the items of the dimensions' codelist to get the data that needs to be inserted in cache.
- *GroupSize* defines the number of codes that will be send for each request to the endpoint.

Logconfig

In order to configure logs it will be sufficient to set the parameters in the file "datarowser-hub/config/base/logconfig.xml". This configuration allows the user to distinguish web application logs from those of the scheduled services. Possible log's level that can be set are: Debug, Information, Warning, Error.
Here an example of part of a log configuration file:

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <appSettings>
        .
        .
        .

        <!-- Web Application Logger (DataBrowser.log) -->

        <add key="log:serilog:using:File"
             value="Serilog.Sinks.File" />
        <add key="log:serilog:filter:ByIncludingOnly.expression"
             value="RequestId is not null and Length(RequestId)
                 &gt; 0" />
        <add key="log:serilog:write-to:File.path"
             value="DataBrowser.log" />
        <add key="log:serilog:write-to:File.restrictedToMinimumLevel"
             value="Error" />
        <add key="log:serilog:write-to:File.formatter"
             value="Serilog.Formatting.Compact.CompactJsonFormatter,
                 Serilog.Formatting.Compact" />
        <add key="log:serilog:write-to:File.outputTemplate"
             value="[{Timestamp:HH:mm:ss.fff}
                 {Application} {Level:u3}]
                 [{RequestId}]: {Message:lj}
                 {SourceContext} {NewLine}{Exception}" />
        <add key="log:serilog:write-to:File.fileSizeLimitBytes"
             value="4000000" />
        <add key="log:serilog:write-to:File.buffered"
```

(continues on next page)

(continued from previous page)

```
            value="false" />
        <add key="log:serilog:write-to:File.rollingInterval"
            value="Day" />
        <add key="log:serilog:write-to:File.shared"
            value="false" />
        <add key="log:serilog:write-to:File.retainedFileCountLimit"
            value="31" />
        <add key="log:serilog:write-to:File.rollOnFileSizeLimit"
            value="true" />
        .
        .
        .

  </appSettings>

</configuration>
```

Export Excel

In this part of the configuration, the user can define all the settings required for the correct and useful functioning of the export to Excel for multidimensional tables.

```
"Export": {
"MaxExcelSheets": 50
"MaxColsPerSheet": 100,
"MaxRowsPerSheet": 1000
},
```

Specifically:

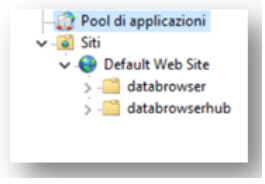
- *MaxExcelSheets* is the maximum number of sheets to be created.
- *MaxColsPerSheet* specifies the maximum number of columns per sheet.
- *MaxRowsPerSheet* specifies the maximum number of rows per sheet.

Having to manage the occupation of resources on the server and waiting times for the end user, these configurations make it possible to limit the number of objects that can be exported by the user.
The limitations set are by default but can be modified according to specific needs.

3.5 Application deployment

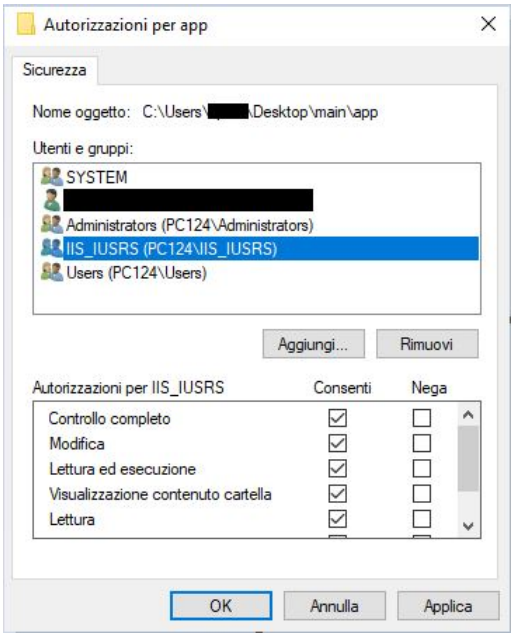
IIS CONFIGURATIONS

First of all, the user needs to copy the two folders from the software package (*databrowser* and *databrowserhub*) in the IIS root directory which from now on we will refer to as **[ROOT_IIS]** (i.e. C:\inetpub\wwwroot\ ) and open the ISS Manager where these folders will now appear.

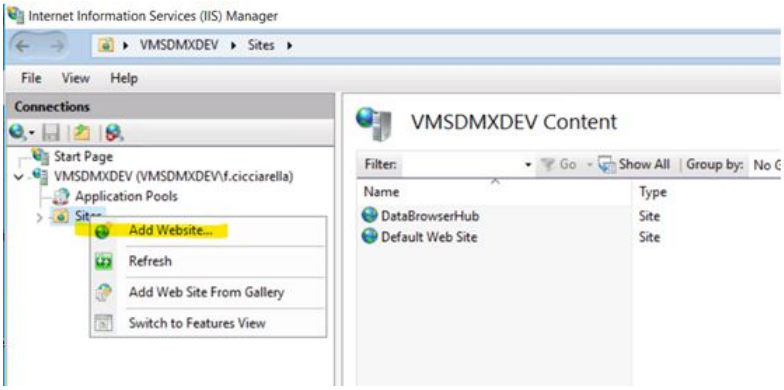


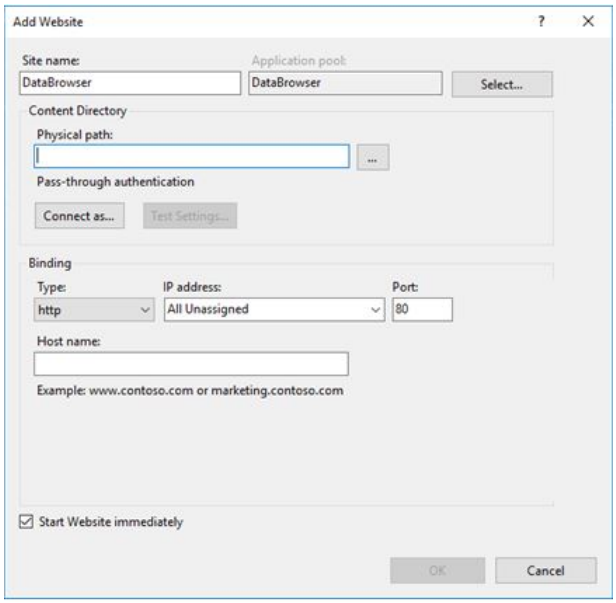
The users IIS_IUSRS e IUSR must have the suitable permissions on the web applications, therefore on the folder:

- right click on the folder;
- select *Property/Security*;
- click on *Edit/Add*;
- in the section *"Locations"*, select the local computer;
- in the section *"Enter the object name to select"* write IIS_IUSRS;
- click on *"check names"* and then OK;
- in the section *"Permission for IIS_IUSRS"* include *"full control"*;
- repeat steps from 3 to 6 for user IUSR
- in the section *"Permission for IIS_IUSRS"* include *"write/read"* permissions.



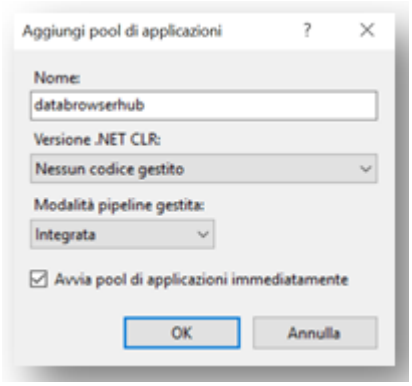
The user can now create a website for the Data Browser



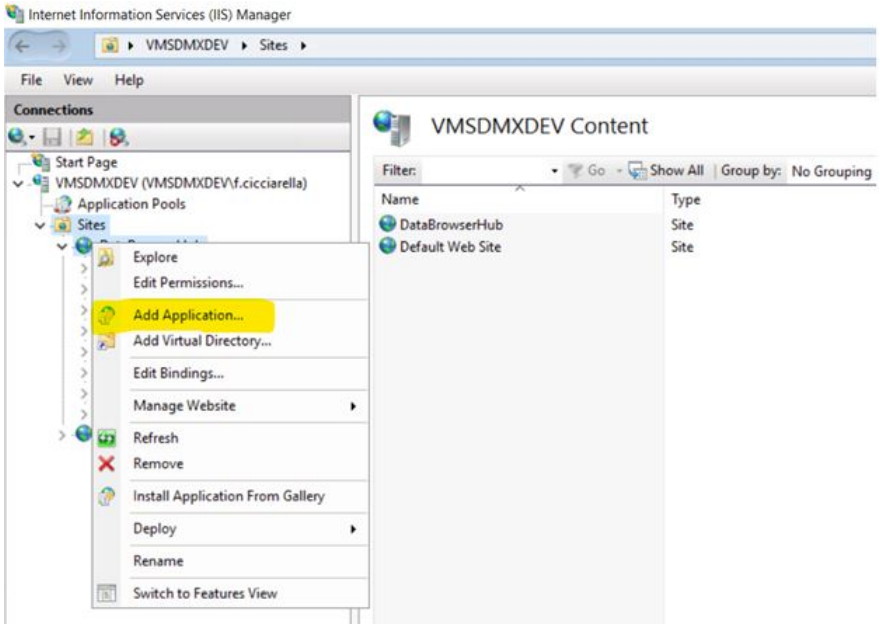


The user selects the DefaultApplicationPool or creates a new one with type **.NET CLR Version 4** and selects the psysical path of the databrowser's folder. Finally, he inserts the hostname of his machine (or name created by the network administrator).

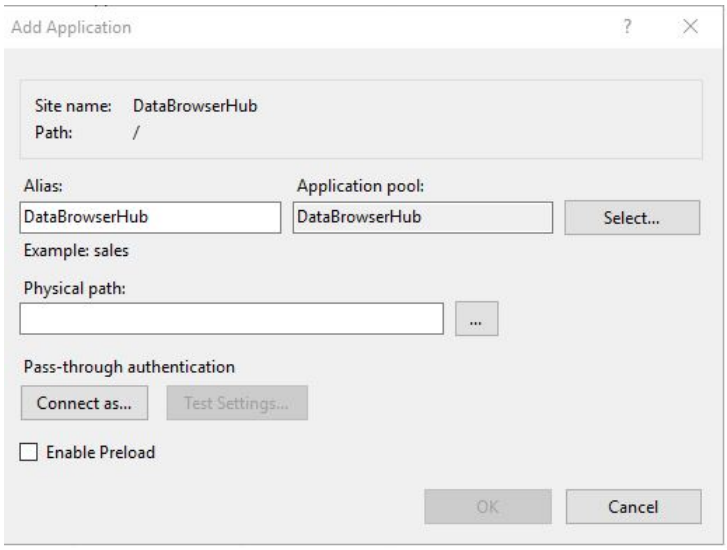
At this point the user needs to create an application pool for the web service "databrowserhub". This operation can be made by right-clicking on "Application Pool" and by selecting "Add Application Pool" item



Afterwards, the user needs to create a new application (as shown in picture):



and configure it



Once the previous operations are completed, the user selects the application pool created for the DataBrowserHub and inserts the path to the databrowserhub's folder into the *Physical path* field.

Most recent browsers use aggressive caching techniques that tend to contact the web server only as necessary. The client application implements a mechanism to minimize requests to the web server while keeping its latest version in the browser cache. In order for this mechanism to work, it is necessary to configure the web server so that caching of the index.html file is denied. Such operation is made possible by adding some custom headers in the web.config file, stored in the IIS root directory, indicating that the location path file (in this case index.html) must include these just mentioned headers in order to avoid caching.
The following part of code is the one that was added in the web.config file:

```
<?xml version="1.0" encoding="UTF-8"?>
  <configuration>
```

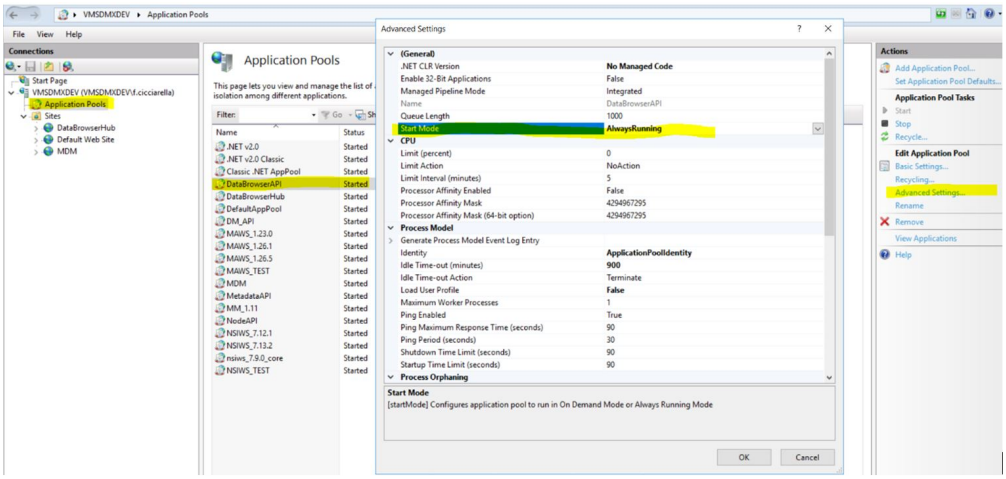
(continues on next page)

(continued from previous page)

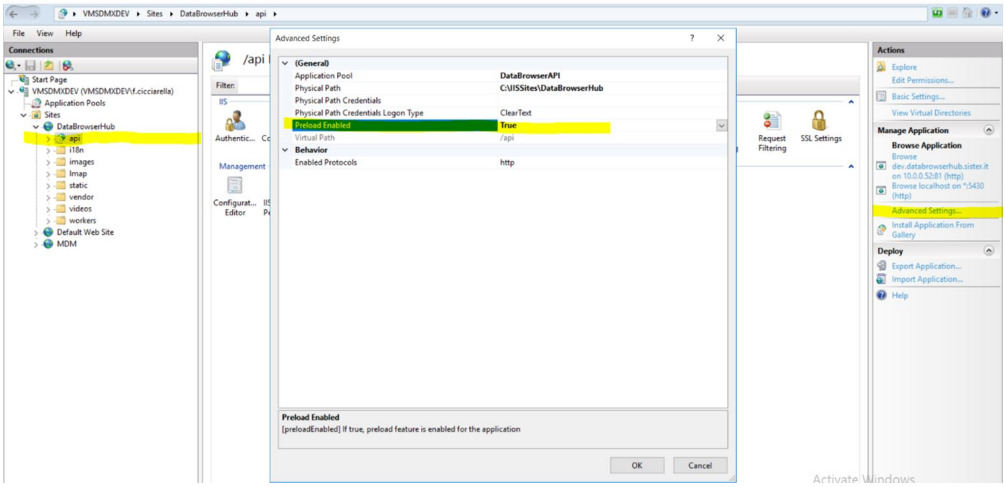
```
  <location path="index.html">
   <system.webServer>
    <httpProtocol>
     <customHeaders>
      <add name="Cache-Control" value="no-store, must-revalidate"/>
      <add name="Pragma" value="no-cache" />
      <add name="Expires" value="0" />
     </customHeaders>
    </httpProtocol>
   </system.webServer>
  </location>
</configuration>
```

IIS Configuration for Application Initialization

Initialization needs to be applied on the Application Pool as well as the IIS Application level. For all possibile IIS' configurations, please check the official Microsoft documentation on the matter: https://docs.microsoft.com/.
Here are some recommended configurations, to leave the IIS pool always active.



On the Site/Application level user can specify whether the site should pre load; in this case the Preload Enabled flag needs to be set to true.



HTTPS BINDINGS

It is needed to create a binding for https. The task can be performed as follows:

- click on *Default Web Site*;
- click on *Binding* in the *Actions* menu on the top-right;
- click on *Add*;
- select *http sas type*;
- select an available SSL certificate;
- click on *OK*.

OTHER POSSIBLE CONFIGURATIONS

It is also possible to set other parameters (time-outs, lenght content, etc.) that can be useful according to the users' needs. Here are some examples:

- ***Maximum allowed lenght for the content***
  - Click on the IIS Web Site under which the application has to be installed;
  - double click on the *Requests filtering* menu item;
  - click on *Edit feature settings*;
  - modify the *Maximum allowed content length (byte)* to the desired value.
- ***Connection time-out***. The connection time-out parameter has to be set in order to allow delayed responses by the web services. The suggested value for this parameter is 6000 seconds (100 minutes).
  - Click on the IIS Web Site under which the application has to be installed;
  - select the *Advanced Settings* menu;
  - click on *Limits*;
  - modify the *Connection Timeout* parameter to the desired value.
- ***Request time-out***. This parameter allows to increase the time interval after which a time-out error is launched (blocking the execution) during the waiting of a response by a web service. It is suggested to increase this parameter to 120 minutes.
  - Click on the IIS Web Site under which the application has to be installed;
  - select *Configuration Editor*;
  - access the *system.webServer/aspNetCore* section;
  - modify the *requestTimeout* parameter.
- ***Execution time-out***. This parameter, similar to the previous, allows to increase the time after which a timeout is launched (blocking the execution) after the execution of a web service that doesn't modify its execution status. Is suggested to increase this parameter to 120 minutes.
  - Click on the IIS Web Site under which the application has to be installed;
  - select *Configuration Editor*;
  - access the *system.webServer/httpRuntime* section;
  - modify the *executionTimeout* parameter.

- ***Session state***. In order to increase the application session duration, the *Session State* parameter has to be set. It allows the maintenance of the session cookies without constraining users to re-login to the application.
  - In IIS manager, click on the *Default Web Site*;
  - click on the *Session State* menu;
  - set the option *TimeOut* (in minutes) to a suitable value (e.g. 60 minutes)
- ***Idle time-out***. This parameter determines the time after which an idle web service is stopped. It allows to eliminate the waiting time for restarting the web service in case of a very long session. It has to be set for each pool involved in long duration tasks.
  - Click on the pool;
  - select Advanced settings;
  - modify the Idle TimeOut parameter i.e. by setting it to 120 minutes.

3.6 Super Administrator

Superadmin user is generated automatically when the database is created and initialized during installation with username: admin@databrowser.com and empty password. It is strongly recommended, but not mandatory, to change the password at first login by following the instructions in the paragraph "Manage user password".

3.7 Application Upgrade

To update the application, the user must follow the following steps:

- stop the application pools
- overwrite the already published databrowser and databrowserhub folders
- restart the application pools

3.8 Quick steps

This paragraph contains the synthetic summary of the steps needed to install and configure the application, considering that the prerequisites have been already satisfied.

1. Download the software package ***databrowser_x.x.zip***
2. Extract the two folders from the package(*databrowser* and *databrowserhub*) and copy them into the IIS root directory
3. Set the ISS configurations
   - Assign to the IIS_IUSRS and IUSERS users read/write grants to the folder
   - Create an IIS application pool for the web service "databrowserhub"
4. Deploy the web service
   - Create under the IIS Default Web Site a new application associated with the just created application pool **"databrowserhub"**

5. Start the navigation
    - Go to http://localhost/databrowser

3.9 Customizing Hub and nodes styles

In the client folder, when starting configuration of the application, it is also possible to set a personalization of the styles of the application itself or a single node inside the application. User can perform this operation by changing the custom.css file inside the DataBrowserHub-Client folder.
The following sections show examples on how to customize parts of the application modifing the css file.

ENTIRE HUB PERSONALIZATION

It is possible to customize colors for principal structures (Header banner background, principal application buttons, text color in databrowsing windows) and/or for secondary structures (background color in buttons in hub's main page, rendering messages when opening files or applying configurations, loading bars).
Example:

```
/*Color for principal structures which include:
  - Header banner background color
  - principal application buttons (in data visualization and windows)
  - text color in databrowsing windows (criteria, layout, ...)*/

.theme__palette-primary-main {
        color: #00295a
}

.theme__palette-primary-light {
        color: #3b5187
}

.theme__palette-primary-dark {
        color: #000030
}

/*Contrast text color for structures
having palette-primary background color*/

.theme__palette-primary-contrastText {
        background-color: #ffffff
}

/*
Secondary structures colors which include:
  - background color in buttons in hub's main page
  - rendering messages when opening files or applying configurations
  - loading bars
*/

.theme__palette-secondary-main {
        color: #f8a81e
}
```

(continues on next page)

(continued from previous page)

```
.theme__palette-secondary-light {
        color: #ffd956
}

.theme__palette-secondary-dark {
        color: #c07900
}

/*Contrast text color for structures having palette-secondary
background color (information-nodes-dashboards buttons)*/
.theme__palette-secondary-contrastText {
        color: #00295a
}
```

SINGLE NODE PERSONALIZATION

The following css code sets style's personalization to a node which ID is "CUSTOM". In particular considers the main and secondary colors of the structures. As in the previous section, structure customized are always the same but in this case, changes are applied only to a specific node. So new colors impact the Header banner background color, the principal application buttons, the text color in databrowsing windows for the main structures, and background color in buttons in node's main page, rendering messages when opening files or applying configurations, loading bars on the secondary structures.

```
/*Color for principal structures which include:
        - Header banner background color
        - principal application buttons (in data visualization and␣
↪windows)
        - text color in databrowsing windows (criteria, layout, ...
↪)*/

.theme__node__CUSTOM__palette-primary-main {
        color: #bf360c
}

.theme__node__CUSTOM__palette-primary-light {
        color: #f9683a
}

.theme__node__CUSTOM__palette-primary-dark {
        color: #870000
}

/*        Contrast text color for structures
having palette-primary background color*/

.theme__node__CUSTOM__palette-primary-contrastText {
        background-color: #ffffff
}

/*
Secondary structures colors which include:
        - background color in buttons in node's main page
        - rendering messages when opening files or applying␣
↪configurations
        - loading bars
```

(continues on next page)

(continued from previous page)

```
*/

.theme__node__CUSTOM__palette-secondary-main {
        color: #7cb342
}

.theme__node__CUSTOM__palette-secondary-light {
        color: #aee571
}

.theme__node__CUSTOM__palette-secondary-dark {
        color: #4b830d
}


Contrast text color for structures having palette-secondary
background color (information-nodes-dashboards buttons)
.theme__node__CUSTOM__palette-secondary-contrastText {
        color: #000000
}
```

It is also possible to set custom colors to tables of a specific node. In the next code example, colors of borders and background of cells are modified for the node with ID "CUSTOM". Every line of code modifies a specific part of the table (borders, rows and columns).

```
/*****EXTERNAL TABLE BORDER (red border) ************/
/* Top border first row of the table */
#node__CUSTOM thead tr:first-child th.c{
        border-top-color: #da0d14 !important;
}

/* Left border first column of the table (head section) */

#node__CUSTOM thead tr th.c.cfm.ch.cl0{
        border-left-color: #da0d14 !important;
}

/* Left border first column of the table (body section) */

#node__CUSTOM tbody tr th.c.cfm.csh.cl0{
        border-left-color: #da0d14 !important;
}

/* Right border last column of the table */

#node__CUSTOM .c.c-rb{
        border-left-color: #da0d14 !important;
}

/* Top border of the last row (not visible with data) of the table */

#node__CUSTOM .c.c-bb{
        border-top-color: #da0d14 !important;
}
```

(continues on next page)

(continued from previous page)

```
/***** END EXTERNAL TABLE BORDER ****************/

/***** TABLE'S BODY ROW BACKGROUND *************/

/* Alternating rows color (grey even and white odd)*/
#node__CUSTOM tbody tr:nth-child(even) td.c.cfm{
        background-color: #dcdcdc !important;
}


/******* END TABLE'S BODY ROW BACKGROUND *********/

/******* TABLE'S LAYOUT **************/

/*Background color and text color
for dimensions set in rows (dimensions' titles)*/

#node__CUSTOM thead tr[data-row-key="hh"] th.c.cfm.ch{
        background: #fff !important;
        color: #000000;
        text-decoration: underline;
}

/*Background color for dimensions set in rows
(cells componing rows with no titles)*/

#node__CUSTOM thead tr[data-row-key="hh"] th.c.cfm.csh{
        background: #fff !important;
}

/*Background color and text color
for dimensions set in sections */

#node__CUSTOM tbody th.c.cfm.cs{
        background-color: #da0d14 !important;
        color: #ffffff;
}

/*Background color and text color for dimensions
set in columns (dimensions' titles)*/

#node__CUSTOM .c.cfm.ch{
        background: #aaa !important;
        color: #000000;
        text-decoration: underline;
}

/*Background color and text color for dimensions
set in columns (dimensions' single items)*/

#node__CUSTOM thead th.c.cfm.csh{
        background: #aaa !important;
        color: #ffffff;
}
```

Last but not least, user can also change the map colors in the application or just in a specific node. In case of node customization, remember to always write the node's ID (like in the

example “CUSTOM”).

```
.map__start-color {
    color: white;
}
.map__end-color {
    color: black;
}
.map__node__CUSTOM__start-color {
    color: orange;
}
.map__node__CUSTOM__end-color {
    color: red;
}
```

Another important functionality contained in the Data Browser application is the web accessibily. It is also possible to configure a personalized style of the page when this functionality is enabled.

```
/*
-------------
ACCESSIBILITY
-------------

you can apply colors for accessibility (a11y) mode too, like this:*/
.a11y-theme__palette-primary-main { color: white }

/*and for specific nodes too:*/
.a11y-theme__node__NODE_ID__palette-primary-main { color: white }
```

FOOTER PERSONALIZATION

It is possible to personalize the footer, modifying the file present in the client installation in the directory: “databrowser/footer/index.html”.

3.10 Resolving Chrome downloading issues: integration of proxy to download files from HTTP Origin

In case a dataflow has an annotation containing an attached file URL, the ability to do the download will be added in the Data Browser.

In order to allow maximum flexibility and support for several different usage scenarios, the URL entered into that annotation by the user is never altered, i.e. it arrives to the user’s client exactly as it was entered into the annotation. This approach also guarantees the possibility of completely decoupling the file repositories from the Data Browser, with particular reference to the fact that users who want to proceed to download the file will forward their HTTP requests directly to the server where it resides, transparently to the server where the Data Browser is installed. In other words, all the load resulting from these downloads is completely delegated to the server that hosts the files, which may or may not be the same one that hosts the Data Browser.

In case this mechanism is used in a Data Browser installed in HTTPS, but the resources reside on a server that does not use the HTTPS protocol, the Chrome browser detects this download as “not secure” and blocks it. On the contrary, on Microsoft Edge and Firefox this problem does not occur with the current versions and the download is allowed. This security limitation

was introduced by the Chrome browser in August 2020 and is documented at this link: https://blog.chromium.org/2020/02/protecting-users-from-insecure.html

In order to overcome this problem, which does not depend on the application, it is possible to act at an infrastructural level by introducing a proxy on the server where the Data Browser is installed, whose only purpose is to link the URLs of these files to those of the same HTTPS domain of the Data Browser. Example: let's assume that a proxy is configured which can be reached at the url "https://www.databrowser_domain.com/proxy" and which, for security reasons, only manages the domain "unsecuredomain.com". At this point it would be possible to modify the current annotations so that they do not point directly to "http://unsecuredomain.com/" but to "https://www.databrowser_domain.com/proxy?http://unsecuredomain.com/". In this way the download would refer to the same HTTPS domain as the application, so it would no longer be blocked by Chrome.

The use of this infrastructural solution would also allow to better manage the case in which the fileserver is not accessible from outside but only from the server of the Data Browser through appropriate permissions.

Obviously, as with any other approach in which the URLs for the download refer to the application server and not to the original one, there are the following limits:

- the download time needed by the user to download the file must be added to the time needed by the proxy to download it from the file server
- the activity of download management weighs entirely on the application server, so in the case this is particularly important, all application performances can suffer, starting from a potential saturation of the band.

However, we highlight how the proposed solution is in our opinion to be considered optimal compared to others because:

- it operates exclusively on an infrastructural level, without impacting/complicating the application logic with potentially very complex functions that are logically not its own responsibility
- it acts exclusively on the proxy configurations, for which various cases/scenarios can be managed with simplicity
- since it doesn't foresee any configuration at the level of the whole application, it allows to manage every single file in a different way, i.e. it's possible to distribute the load deriving from the download operations of the files deciding for every single file if it has to pass from the application server using the proxy or it can point directly to an external URL. In other words, the limits indicated above are manageable for each single file, and you are not obliged to have them for the entire application.
- allows you to set on the proxy any control logic and download limitation.

As software for the proxy can be used any of those found on the network. As an example we indicate this one developed in .NET: https://github.com/Esri/resource-proxy. The same result could be obtained also using a reverese proxy.

CHAPTER

FOUR

ADMINISTRATION

The “Administration” functional component allows super administrator type user to manage the whole application. In particular the following sections will properly explain all types of interventions that can be done.

4.1 How to configure the application

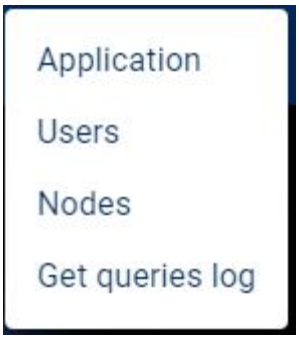
4.1.1 General settings

On the main page of the application, for user logged ad Super Administrators, there are the following icons:

- Flag: to modify the language of the Data Browser.
- Man: to choose between “Classic version” and “Accessible version”.
- Settings: to configure Application, Nodes etc.
- User: to change user information, to manage view and dashboard and logoff.
- Question mark: to retrieve information about application version.

The super administrator user has the authority to configure the application settings (consisting in managing nodes, users and so on). It is possible to configure specific settings by clicking the setting icon:

and choosing the option related to the operation the user wants to perform from the list that appears:



In the "Application" configuration window it is possible to configure a set of application parameters (which will be described in the next paragraph).

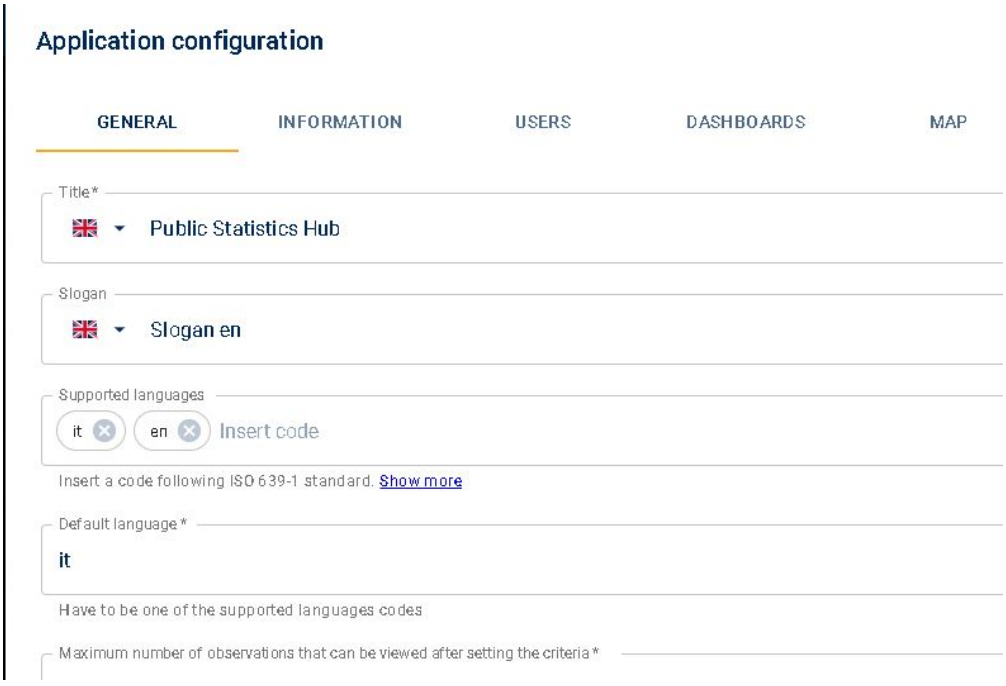
In the "Users" configuration window it is possibile to create, edit or delete users.

By clicking the "Node" link, it is possibile to create new nodes, manage node's cache and templates, delete nodes and, most importantly, set user's permissions on nodes.

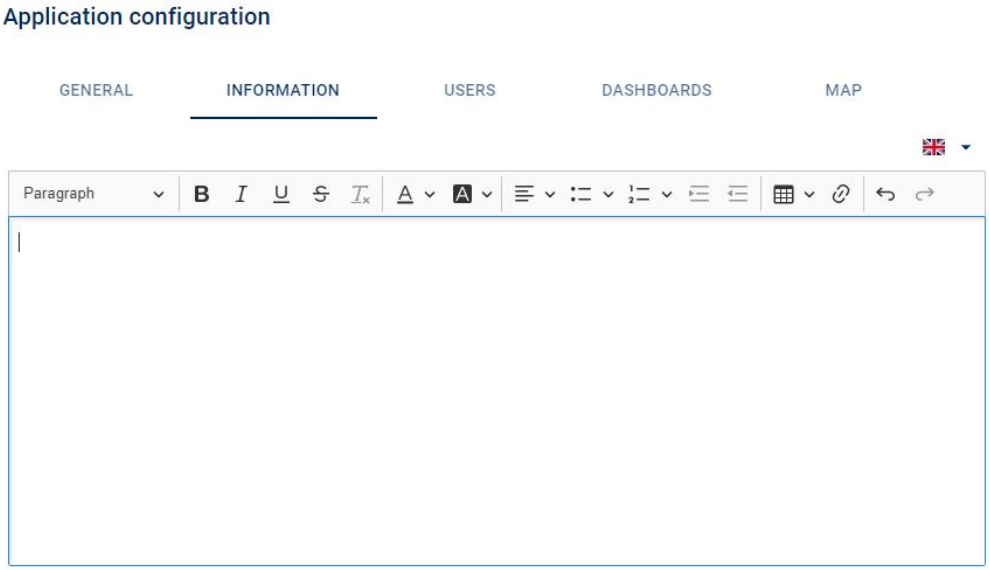
The "Get queries log" link allows the administration user to download the last *n* SDMX-queries requested in the page.

4.1.2 Home page settings

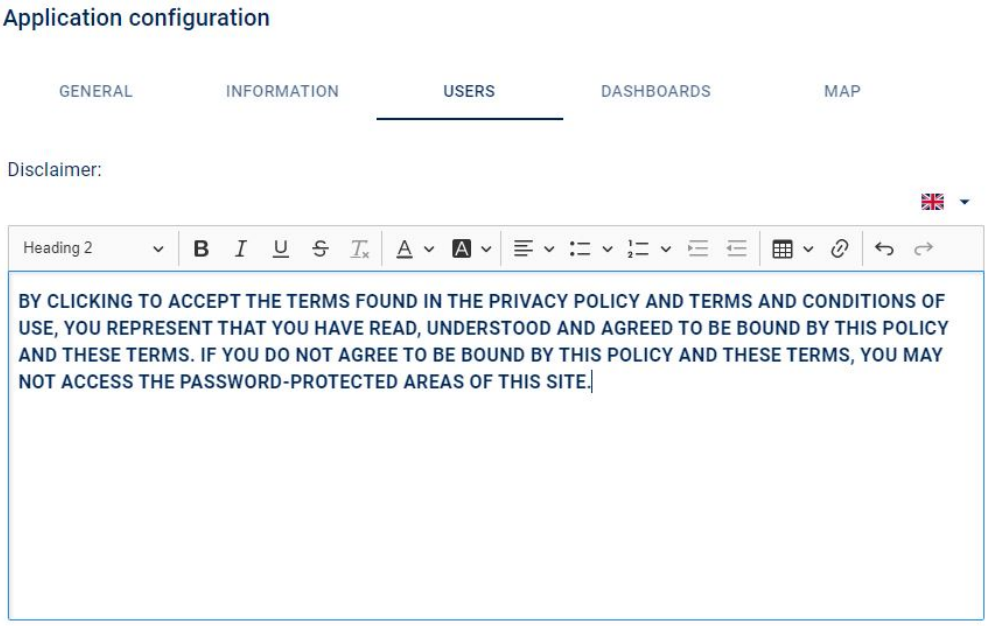
Generally, the home page must have at least a title, a slogan, a section on information and nodes, and also a possible section dedicated to dashboards. It could have a welcome image or short video for the background and different images for the logo on the main page and header. These settings appear in the "application configuration" window:



It is possibile to set some information text, in the home page, describing the purpose and aim of the network. This text must be inserted in the "INFORMATION" tab.

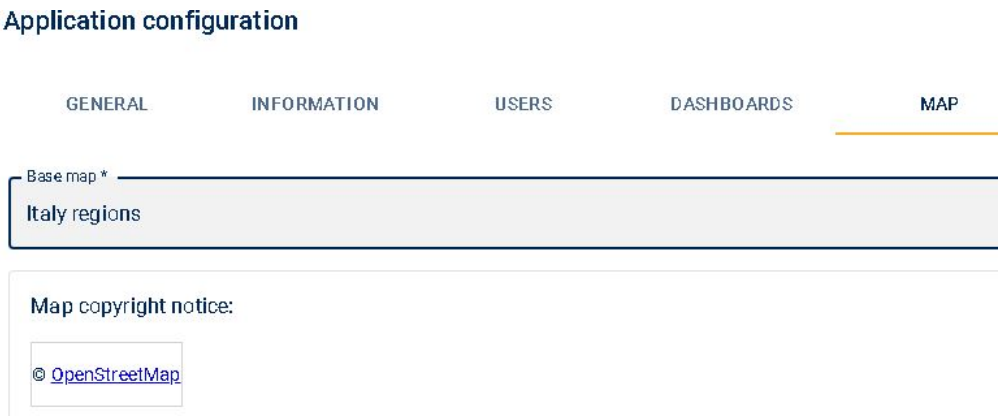


In this window, the super administrator user, can find the "USERS" tab where he can set the disclaimer text that will appear in the "Sign Up" window when a new user decides to sign up to the application (see section *How to add a registered user* for detailed infomations).



In the configuration panel, the super administration user can choose to insert dashboards that will be publicly available to all users once the home page is visited (see section *Dashboard management* for detailed infomations). It is also possible to set a title for each language supported by the application allowing the user to benefit of the multilingual functionality (indicated by the presence of the flag in certain fields).

In the tab "Map" the user can choose which background map to use in geographical report, choosing between "Open Street Map" and "Italy regions". For both maps it is recommended to insert the copyright information.



If only a node is configured, then home page and node home page will be the same; otherwise if more than one node is configured (beside the default one), then the home page will show a section, named "Nodes", containing all nodes contained in the Hub which can be browsed.

4.1.3 Accessible version

The Data Browser application also contemplates web accessibility allowing all kind of people to benefit of the application's functionalities.
Activation of this option is very simple.
On the main page of the hub, click on the little man icon and choose "Accessible version" from the list that appears.

Once the functionality is enables, a green check will appear near the icon  meaning that the application now is in accessible mode (this means that the system will apply special CSS styles that can be defined in the "custom.CSS" file using the special CSS class "a11y", that will allow to treat in a dedicated way any element of the interface in order to increase the

contrast or other properties useful for the purpose. At the moment, the most important operation that this functionality provides, is that when clicking on a data the system will not show the viewing/navigation page of the dataset because this functionality requires an advanced and interactive human-machine interaction, but it will guarantee the fruition of the information contained in the data through the CSV download of that dataset itself.

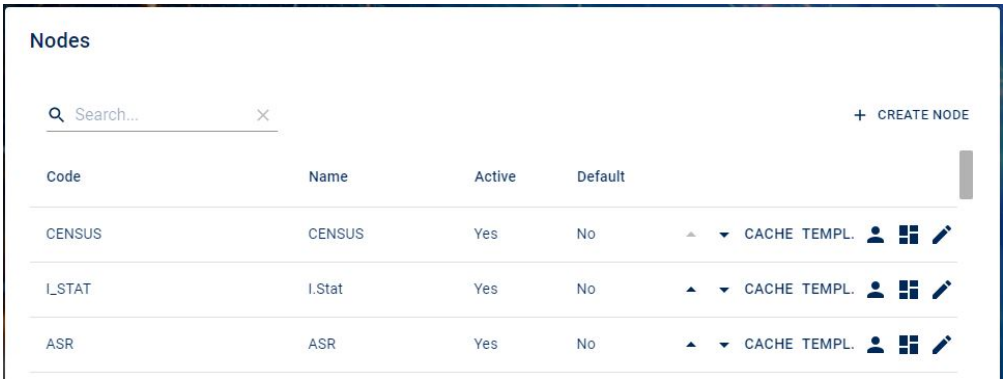
4.2 Hub management

In this section we explore basic operations of managing and configurating nodes.

4.2.1 How to manage data providers (nodes)

From the setting icon, choosing the "Nodes" option, the super administrator user can manage and configure all the present nodes. Moreover in this section he can set the configurations focused on:

1. Ordering the appearance of nodes
2. Setting a default node
3. Checking if the node is active or not
4. Adding/Removing nodes
5. Editing nodes
6. Editing/Deleting Dataflows and catalog cache
7. Viewing/Deleting data templates
8. Setting permissions to node administrator users
9. Editing/Deleting dashboards



The above figure shows all the possible settings. In particular

- the "+ CREATE NODE" adds a new node;
- the small arrows allow the user to set the sorting method for the nodes to appear;
- by clicking on "CACHE" it is possible to edit or delete cache for each dataflow and catalog of the node.
- by clicking on "TEMPLATES" it is possible to view or delete a dataset template

- the person icon allows the super administrator user to set permissions for the other users (such as node administrator users)
- the mosaic icon manages the dashboards for that specific node
- the pencil symbol allows to edit the node.
- the trashcan symbol deletes the selected node;

Node administrator users cannot delete or change position of the nodes they are allowed to manage. Most of all, based on what kind of permission they have on the nodes, not all the above mentioned icons will be present in their node configuration window.

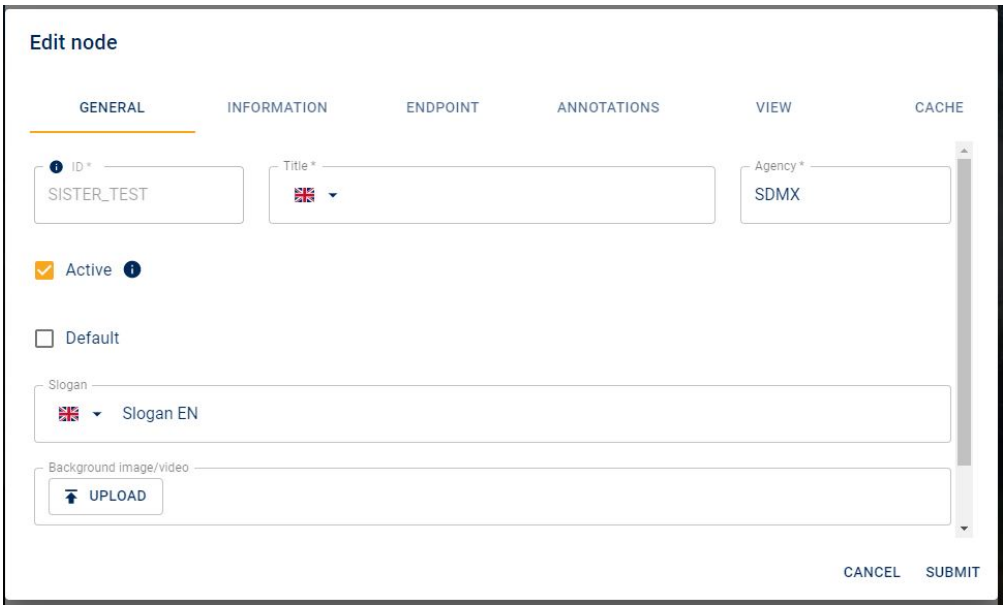
4.2.2 How to configure a data provider

Editing an existing node or adding a new one, will bring to another window where it is possible to configure the fields that will afterwards be the settings for that node. In general these parametes can be customized in the normal node configuration set-up, with a data annotation setting or with the creation of a template. In particular, the parameters relate to:

- Decimal separator: symbol used to separate the integer part from the fractional part of a number written in decimal form. Possible values . or ,
- Number of decimals: number of digits that appear after the decimal point. Usually is set to 1.
- Empty cell character: value set which appears when data has an empty cell
- Default views: default visualization setting (table, chart or map)

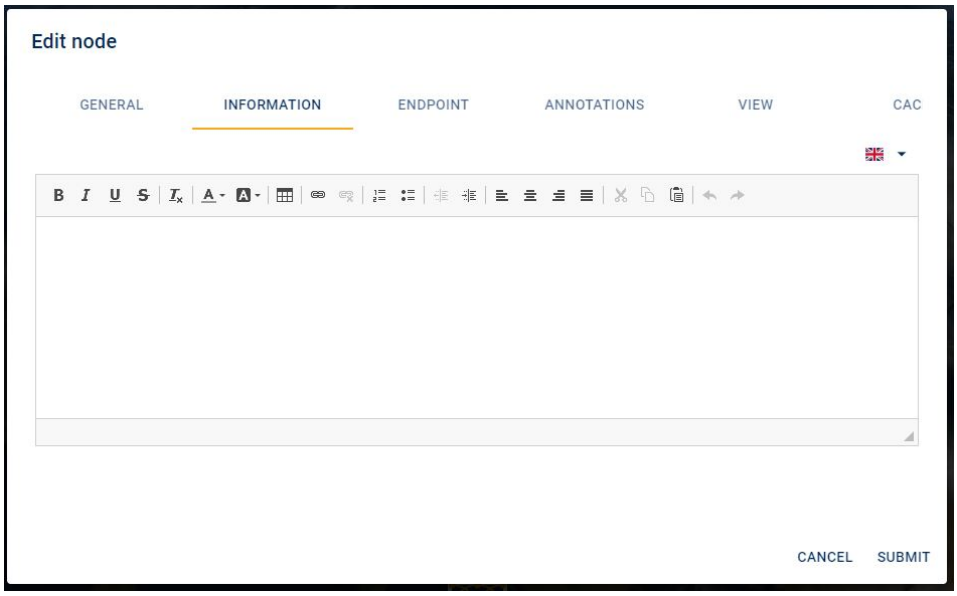
All elements will be described in the following paragraphs when talking about the related configuration levels.

In some tabs the multilingual funcionality is enabled (indicated by the presence of the flag in certain fields). This allows the user to set a title for each language supported by the application. More specifically, the configuration window appears like this:



It contains different tabs, each one configuring specific settings related to the node itself. Getting a closer look to each tab, it is possible to notice that:

- in the ***"General"*** tab, the mandatory fields are the ID of the node, the TITLE and the AGENCY. The user can decide to check the ACTIVE box if the node is ready to be browsed otherwise it must be left unchecked (it might happen that a node is part of the hub but still under construction). The "Default" checkbox allows to set the node as default node of the application. "Slogan", "Background image/video" and "Logo" are all components of the node.
- In the ***"Information"*** tab, the user can give a brief description of the node which will appear once it will be selected.

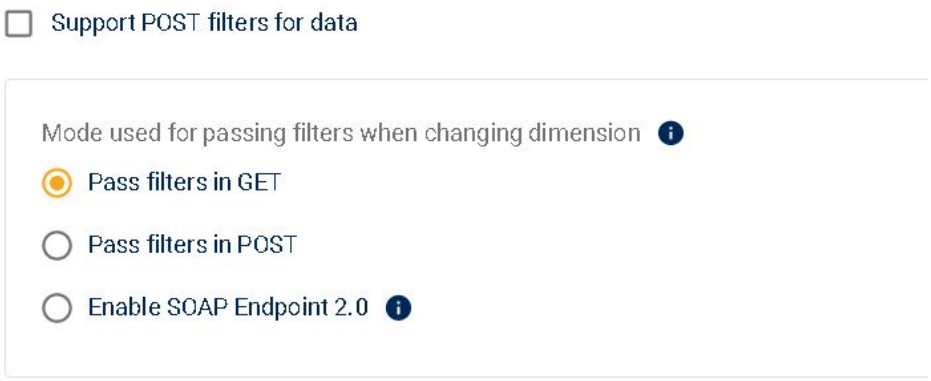


- The ***"Endpoint"*** tab specifies the configurations needed in order to manage the system's capability to connect and query the SDMX web service.

This functionality allows the system to correctly query the data and also to recover any SDMX artefacts useful to the application (for example the category scheme and the categorizations useful for the Node Catalog). It is possible to set a HTTP Authentication and a Proxy, check the specific box to support POST filters and also the possibility to enable a SOAP endpoint. User can also select the method to access the codes of the dataflow

dimensions when defining criteria. There are five possibile choices regarding the criteria selection mode:

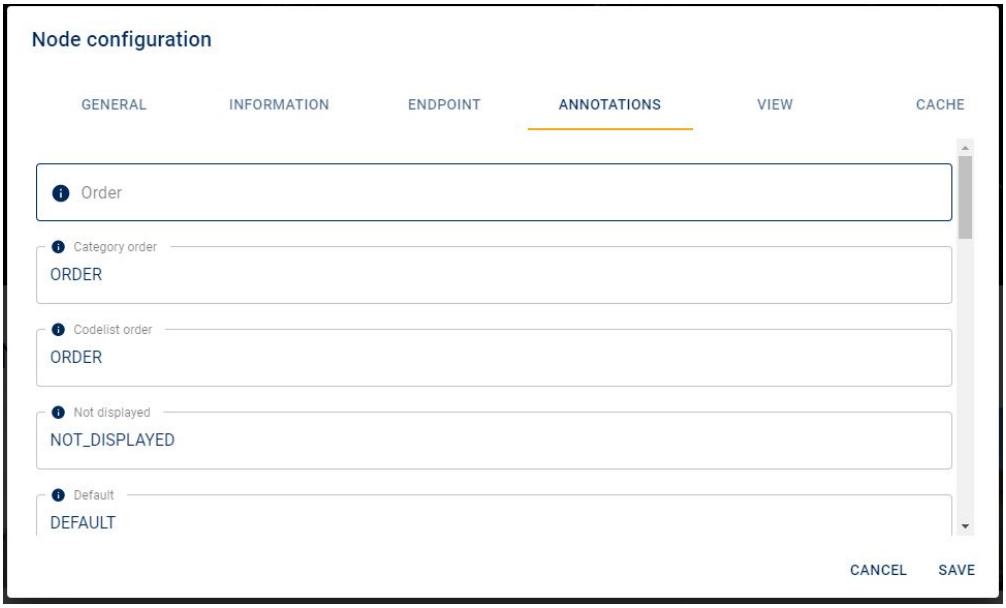
- *ALL_FULL*: in this case all dimensions' items are simultaneously loaded (even those that are not contained in the data) and for each dimension the number of elements will be present.
- *ALL_PARTIAL*: in this case only dimensions' items contained in the data are simultaneously loaded and for each dimension the number of elements will be present.
- *STEP_BY_STEP_FULL*: in this case all dimensions' items are loaded (even those that are not contained in the data) while moving from one tab's dimension to another. For each dimension the number of elements will not be present at the beginning.
- *STEP_BY_STEP_PARTIAL*: in this case case only dimensions' items contained in the data are loaded while moving from one tab's dimension to another. For each dimension the number of elements will not be present at the beginning.
- *STEP_BY_STEP_DYNAMIC*: in this case case only dimensions' items contained in the data are loaded while moving from one tab's dimension to another. Furthermore, choices made in a tab affect next selections acting as filters. For each dimension the number of elements will not be present at the beginning.



If criteria selection mode is: *STEP_BY_STEP_DYNAMIC*, the user can choose the mode used for passing filters when changing dimension between:

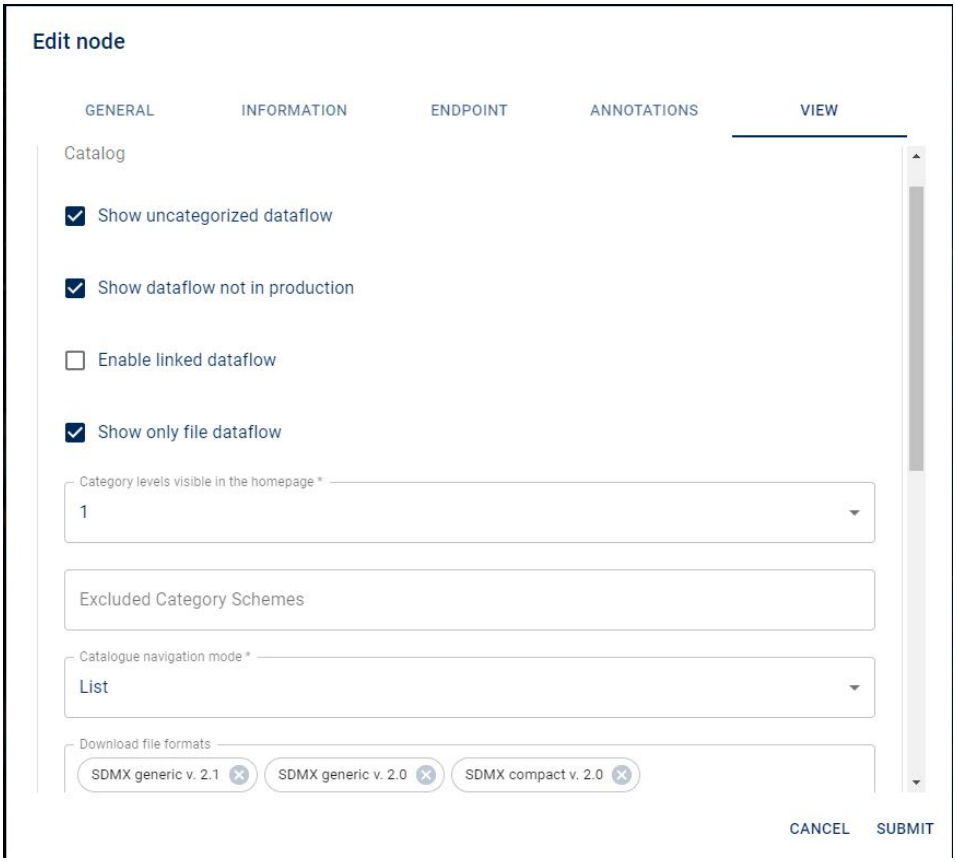
- Pass filters in GET
- Pass filters in POST
- Enable SOAP Endpoint SDMX 2.0

- The *"Annotations"* tab specifies the name of the components used to configure some specific visualization aspects of the node defined through the Annotation mechanism.



- The ***"View"*** tab specifies some other visualization aspects of the node and of all its components. In particular it is possible to select configurations on how to :
  - show uncategorized dataflows.
  - Show dataflow not yet disseminated.
  - Enable linked dataflows.
  - Show only file dataflow.
  - Show all the category levels or just stop to the first one.
  - Exclude particular category schemes from the main page.
  - Choose what kind of visualization the catalog navigation mode should have by default (CARD or LIST for example).
  - Set the download file format.
  - Set the symbol used to separate the integer part from the fractional part of a number written in decimal form (decimal separator) and the number of digits to show after the decimal separator when visualizing data in table view.
  - Insert territorial dimension IDs (if this param is set, the application understands when to activate maps).
  - Choose last N periods of time (dynamic time-range) in order to set a default time period.
  - Configure two fields (not mandatory):
    start day (dd/mm/yyyy, via datepicker)
    end day (dd/mm/yyyyy, via datepicker)
    which will be displayed by default in the absence of the actual time range on the dataflow, i.e. when the SDMX endpoint is unable to retrieve the time coverage of the data.
  - Insert attributes' ids that will be hidden in the data visualization.
  - Show SDMX query info option when visualizing data.

When creating a new node, this tab will already have the following default values:
category levels visible: *first*
decimal number: *1*
decimal separator: *comma*
navigation mode: *list*

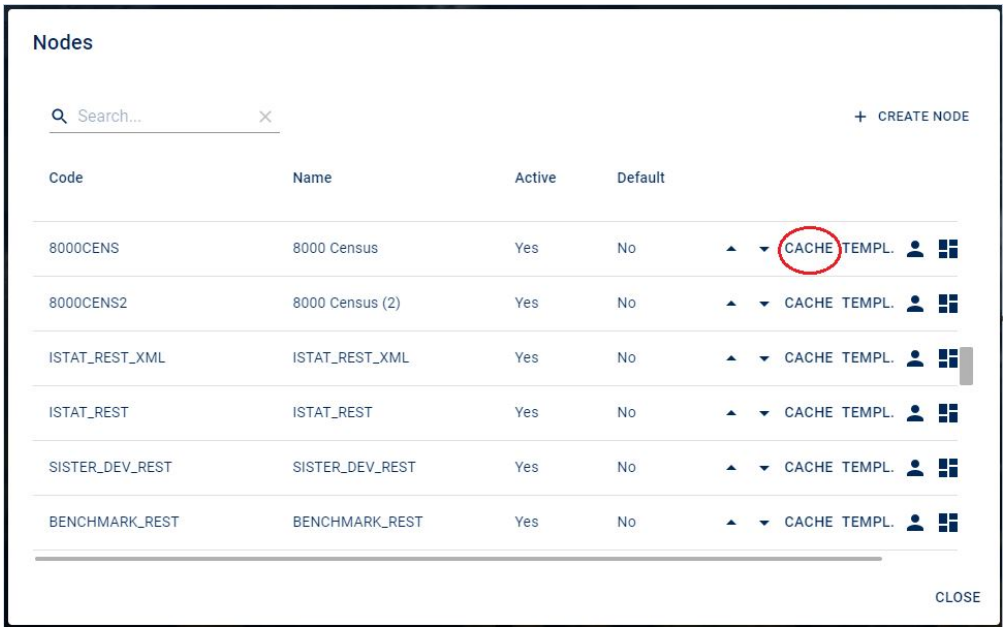


- In the ***"Cache"*** tab, the user can manages cache. In particular it is possible to set cache validity (TTL - time to leave) for:
  - Catalog cache: it concerns the cache of data exposed by a single node, it speeds up the search on catalog, node preview, etc..
    The user can decide cache time duration (in seconds) for all Catalog elements.
  - Dataflows cache: it is related to queries on individual dataflows
    The user can decide cache time duration (in seconds) for all Dataflows.

  If time duration is not set means that cache never expires.

4.2.3 How to configure Dataflows cache

By clicking on "CACHE" on Nodes configuration frame the user opens a new window where it is possible to edit or delete cache for each dataflow.



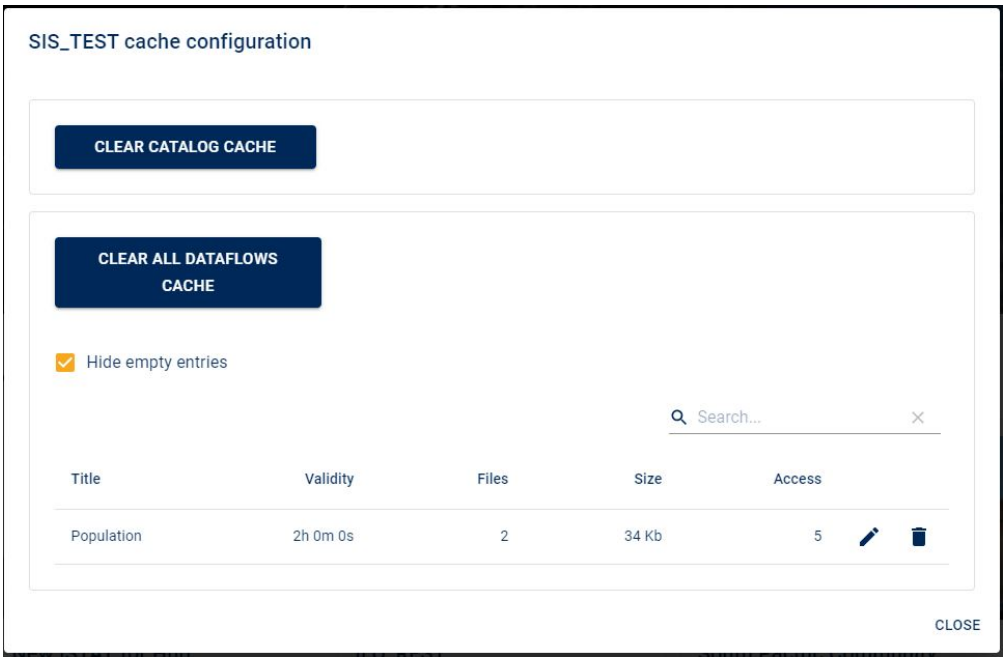
In Dataflows cache configuration window, for each dataflow these attributes are listed:

- cache duration time (in seconds)
- number of cached files
- cache size

The super administrator user, after selecting a dataflow, can change cache duration time (pencil icon) or delete cache (trashcan icon).

Moreover the user can:

- clear catalog cache
- clear dataflow cache for all the dataflows in the node
- show/hide not cached dataflows. By checking the "Hide empty entries" box, if data is not cached, it is not show in the list of cached data.



Node administrator users are allowed to manage cache only if the super administrator user sets them the right permission otherwise the "CACHE" clickable link won't even be present in the node configuration window.

4.3 User management

In this section we will show and describe all possible users that this application allows.
There are different types of scenarios based on the permissions that the superadmin user sets to each other user present.

Let's take a closer look to the user roles and functionalities in the following paragragraphs.

4.3.1 Application roles

There are four types of possible users allowed in this application:

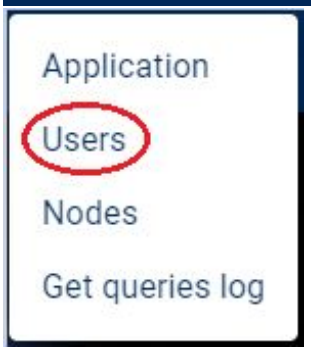
1. **Super administrator**: he is the only one that can access all configuration sections of the application, manage nodes and, most importantly, manage users and roles.
   He can configure the dashboards that can be shown in the Application's home page.
   Of course, the super administrator will have the same permissions of the node administrator and the registered user. (See *How to add a super administrator* paragraph for more details)

2. **Node administrator**: he can create templates and dashboards for the nodes he owns and manages, accessing their specific sections. He can configure the node's home page and, in particular, which dashboard show in it.
   However, it is possible that the node administrator does not have all permissions regarding templates management, cache management and so on. The super administrator sets permissions for the node administrator by checking one or all the options present in the node configuration section. (See *How to add a node administrator* paragraph)
   Permission options relate to:
   - *Cache management*
   - *Template management*
   - *Configuration management*
3. **Registered user**: he has its own account but does not have management permissions. He can visualize and download files of all nodes. He can create, visualize or remove views on the dataflows he can access. (See *How to add a registered user* paragraph for more information)
4. **Anonymous user**: does not have his own account and he has read-only permissions on nodes.

4.3.2 How to add a registered user

A registered user is a common user that has an account which enables him to login the application but does not have permissions regarding management of any kind. He can create views, visualize them and delete them if necessary.
In order to create a new registered user, a super adminstrator user must enter the user setting menu by accessing the main menu of the page

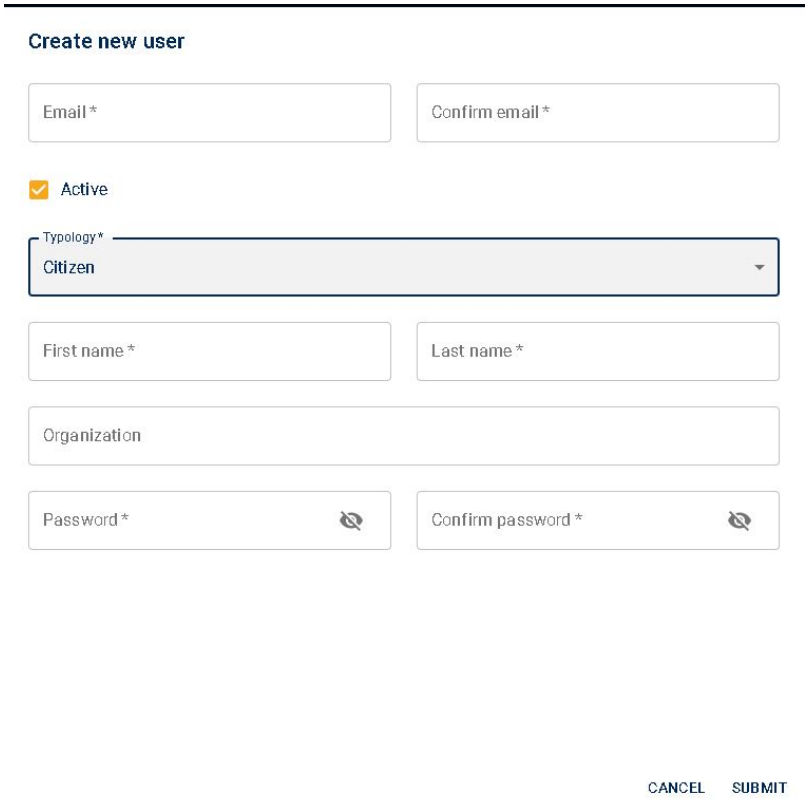
=>



and select the "Create user" button that will bring to a new window which defines the following mandatory and optional fields:

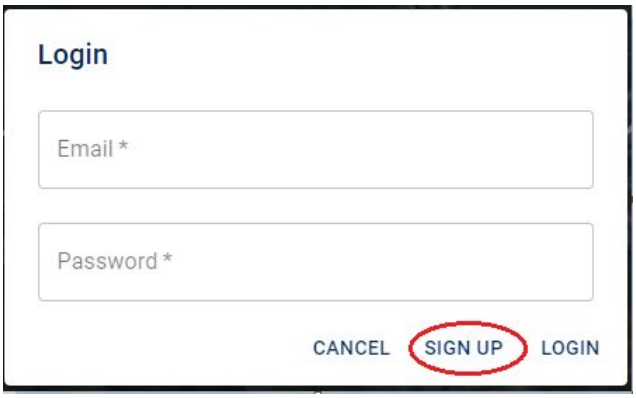
- Email: mandatory. It represents the username necessary for logging in to the application and it cannot be changed.
- Confirm email: mandatory. Field for email confirmation.

- Active checkbox: if checked means that the user is active. The super administrator user can, in any time, deactivate a user.
- Typology: mandatory. It defines if the new user is a citizen or a public authority.
- First name: mandatory. User's first name.
- Last name: mandatory. User's last name.
- Organization: optional. This field is filled if the user is part of some organization.
- Password: mandatory. In this field the new user sets his password.
- Confirm password: mandatory. Field for password confirmation.

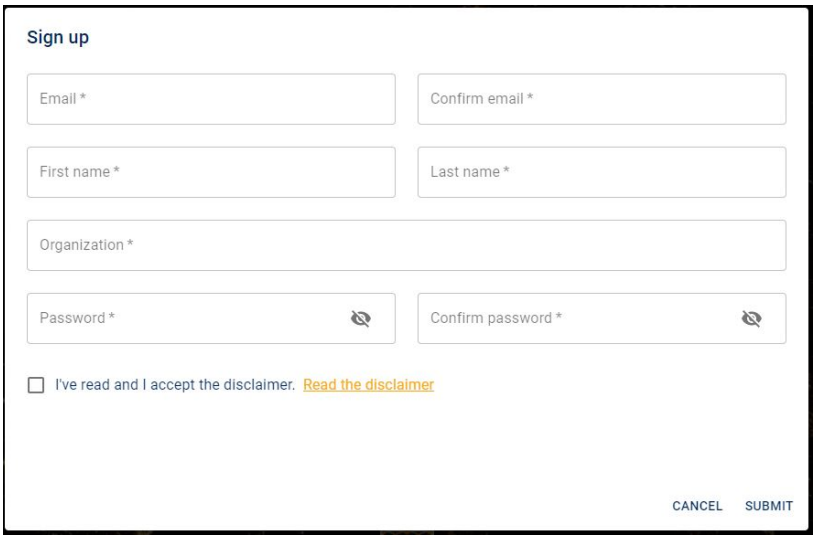


4.3.3 How to sign up to the website

It is also possibile to create a new user by accessig the login link and by clicking on "SIGN UP"



This opens a new window in which the new user defines his information. It is necessary to check the disclaimer box to continue the operation otherwise an error message will appear.

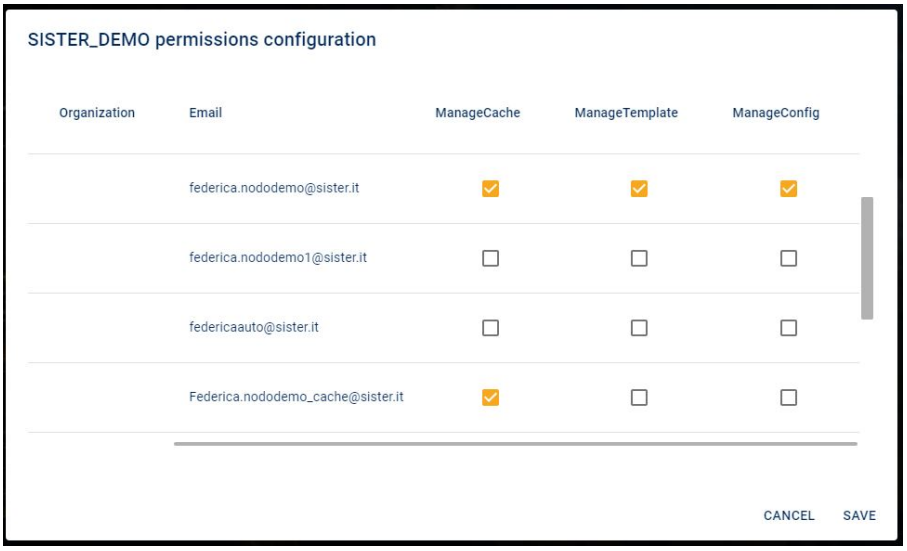


4.3.4 How to authorize a user as node administrator

In order to add a new node administrator, the super administrator user can create a new user (as described in section *How to add a registered user*) and give him permission or he can use one of the already existing users.
From the "Users" configuration panel, the super admininistrator can edit or erase the users. At this point, permission must be given to the new user regarding management of the node (this management relates to node configuration, templates and cache settings).
From the main settings menu, the super administrator selects the node configuration option and by clicking on the person icon of a specific node, enables permissions to the new user for that particular node.

The window that allows the setting, contains all users (because more than one user can manage the same node) and three checkbox that enable, respectively, cache management, template management and configuration management. If all checkbox are selected, the user has all permissions on nodes otherwise, some options might not be present in the node configuration window or data visualization.

For example, considering the following permission given to users federica.nododemo@sister.it and federica.nododemo_cache@sister.it



federica.nododemo@sister.it has access to everything regarding cache, templates, node configuration and dashboard management for the node itself. This user can edit the node settings but cannot delete the node itself.
On the other hand, federica.nododemo_cache@sister.it, can only manage cache and won't even be able to create or view templates.

4.3.5 Manage user's password

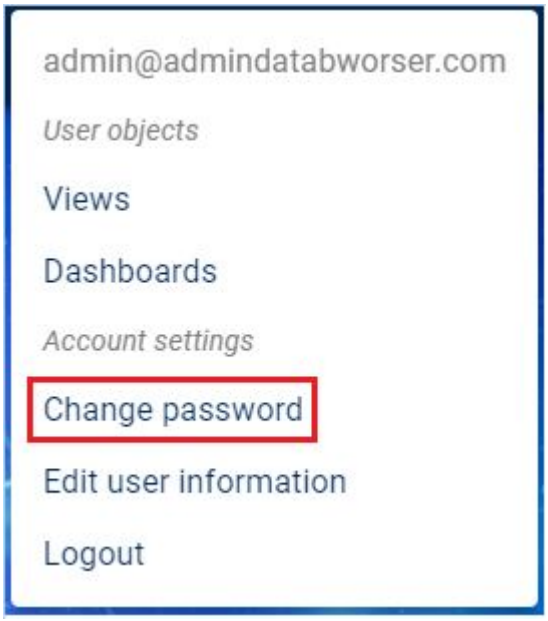
It might happen that the user doesn't remember his password or, in some cases, he would like to change it. In this application, the before mentioned operations are possible and easy to perform.

RECOVER PASSWORD

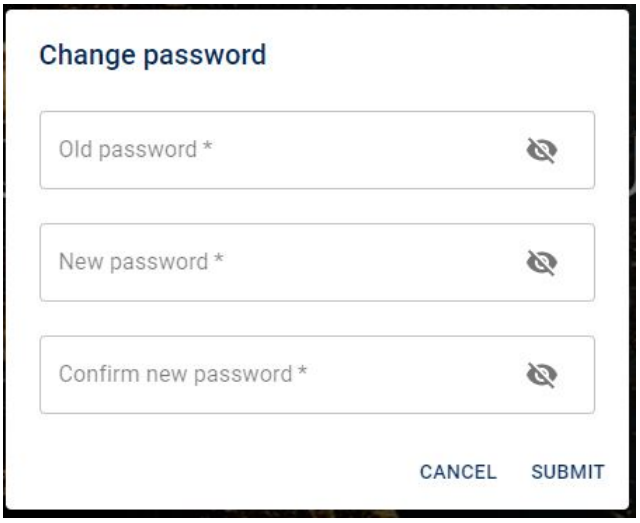
In the login window, right under the User/Password boxes, there is the *"Recover password"* link that allows the user to retrieve his password. Once the user clicks this link, a new window opens in which the email address must be inserted. The user will receive an email containing a restore password link that helps the user to redefine the password once again.
So, no password is sent via email, this operation guides the user to create a new one.

CHANGE PASSWORD

The change password operation is also very easy to perform. Once the user is logged in, by clicking on the user icon on the top right of the page, will show him a small window where the *"Change password"* link is available

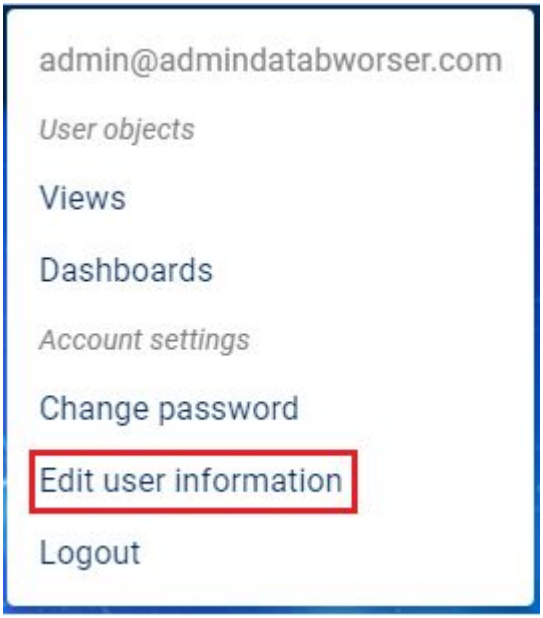


Once the link is clicked, a new window appears in which the user defines his new password

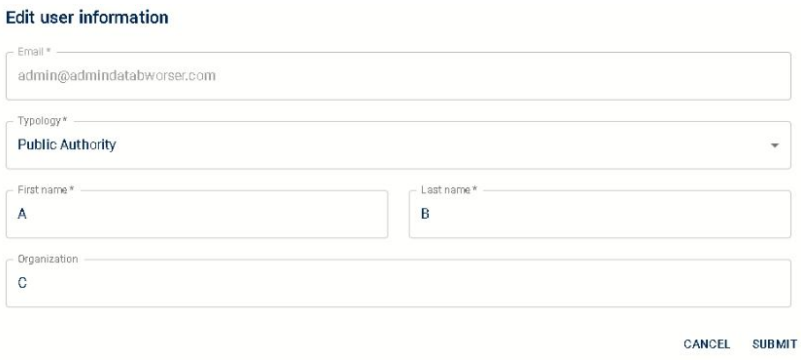


CHANGE INFORMATION

The operation of editing the user's information is also very simple to perform. Once the user is logged in, clicking on the user icon at the top right of the page, it will be shown a small window where the *"Edit user information "* link is available.

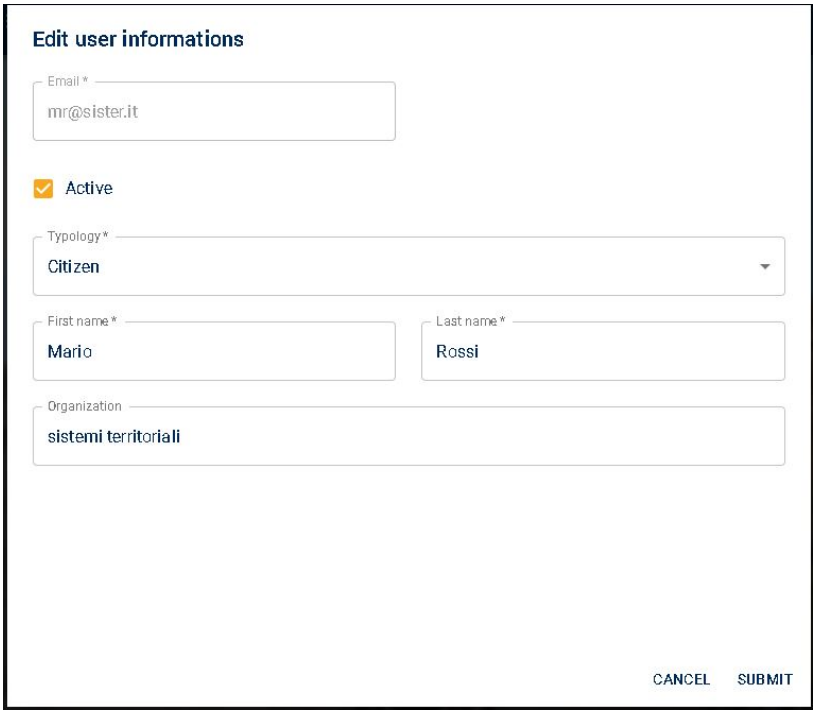


Once the link is clicked, a new window appears in which the user defines his new information

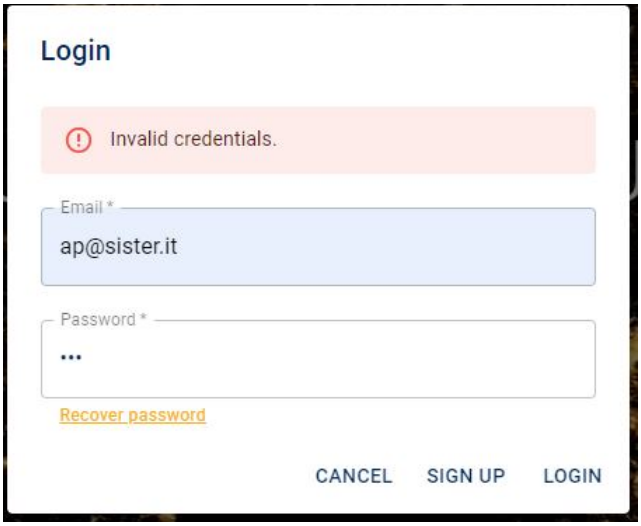


4.3.6 Enable/Disable user

It might happen that a user needs to be disabled, in this case the super administrator user can perform the disable operation.
In the user setting menu (accessible from the main menu of the page), the list of all the registered user appears. Every line of the list refers to an user and his information (name, surname, organization) and one specific column in the lines tells if the user is active or not. If the user's account is active, there will be a YES, otherwise there will be a NO. To change the activeness of the account, the super administrator user needs to click the edit button (pencil icon) and select/deselect the active box.



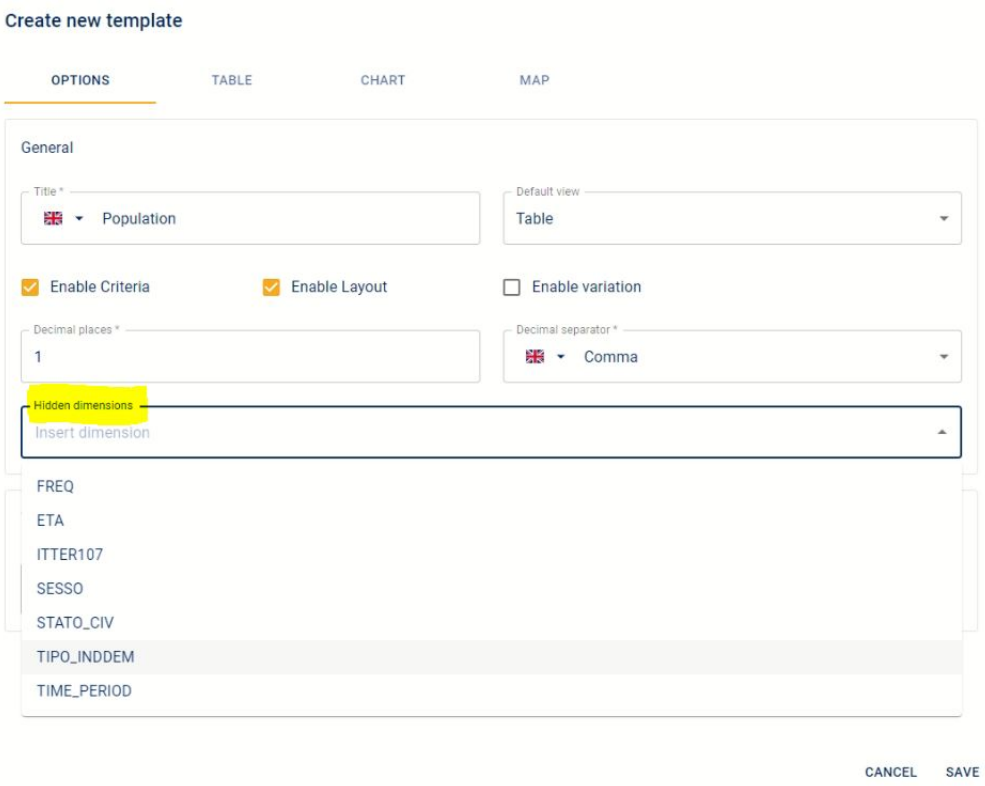
If disabled users try to login the application, an error message will describe the inability to connect.



4.4 Template management

The template is a "default visualization" of the data chosen by the super administration user (and node administrator user if allowed by the super administrator). There can be only one template for each dataset contained in a node. The super administrator user sets the configurations for criteria and layout and saves the template by clicking the save button on the top-right panel above the table. The multilanguage functionality allows users to set different titles depending on the selected language. This is made possible by selecting the flag related to a language and by defining a title for that language.

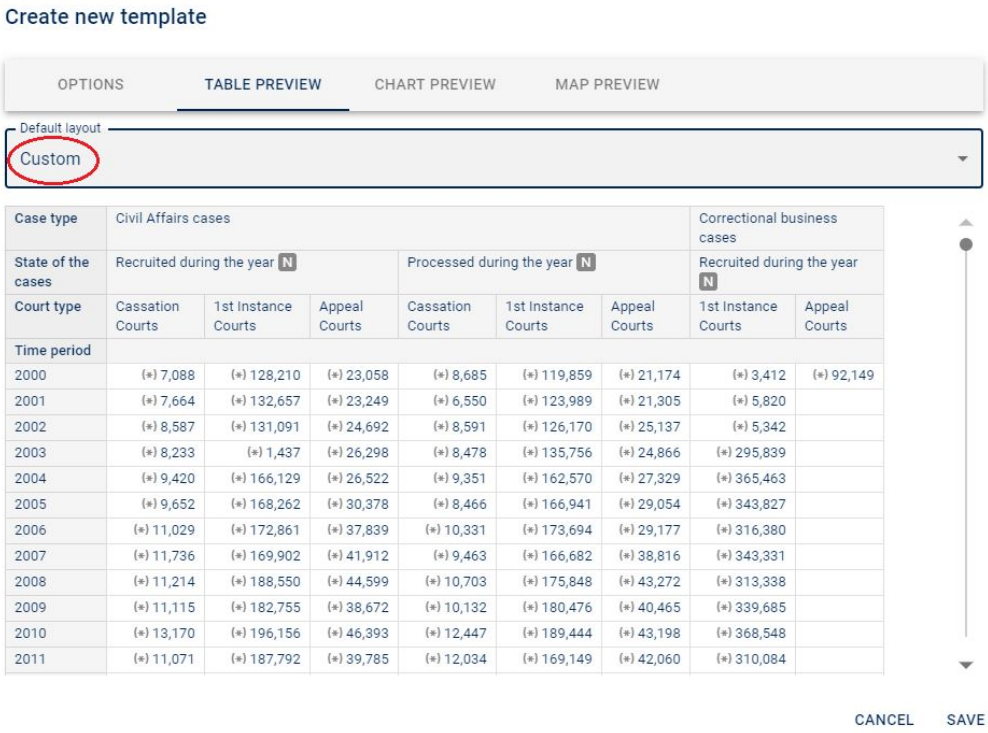
Once "Create new Template" is selected, this action will open a new window that allows to set other configurations.



The super administrator user, in the general tab, can choose to:

- enable/disable the criteria and layout when accessing this data (this means that the buttons and windows regarding the criteria and layout will no longer be shown);
- enable variation;
- Set the symbol used to separate the integer part from the fractional part of a number written in decimal form (decimal separator) and the number of digits to show after the decimal separator when visualizing data in table view;
- set one or more dimensions as hidden, they will be hidden in the visualization phase only if they are composed of only one element but they will be removed from the criteria anyway,
- set the value of the empty cell in case of empty cells.

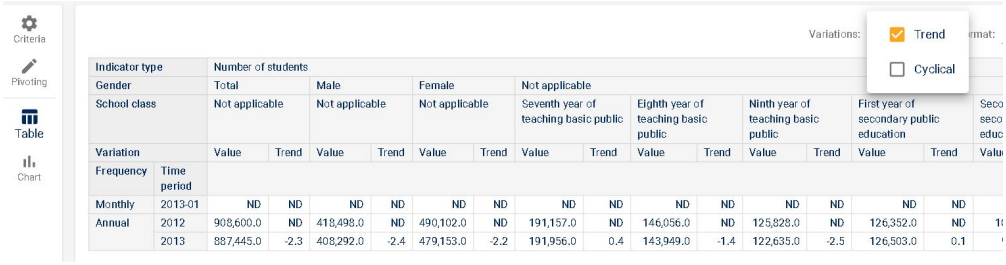
In the table tab, the super administrator user can choose to keep a default view of the table (this means that visualization will depend on what it is set at node level configuration or at data annotation level) or show the custom view that the user has set modifying the criteria and/or pivoting, during the visualization itself.
In the following image, the dafault layout chosen for the template's table is CUSTOM, this means that the user has changed the default layout configuration while visualizing data and wants to save his configuration's choice.



Case type	Civil Affairs cases						Correctional business cases	
State of the cases	Recruited during the year N			Processed during the year N			Recruited during the year N	
Court type	Cassation Courts	1st Instance Courts	Appeal Courts	Cassation Courts	1st Instance Courts	Appeal Courts	1st Instance Courts	Appeal Courts
Time period								
2000	(*) 7,088	(*) 128,210	(*) 23,058	(*) 8,685	(*) 119,859	(*) 21,174	(*) 3,412	(*) 92,149
2001	(*) 7,664	(*) 132,657	(*) 23,249	(*) 6,550	(*) 123,989	(*) 21,305	(*) 5,820	
2002	(*) 8,587	(*) 131,091	(*) 24,692	(*) 8,591	(*) 126,170	(*) 25,137	(*) 5,342	
2003	(*) 8,233	(*) 1,437	(*) 26,298	(*) 8,478	(*) 135,756	(*) 24,866	(*) 295,839	
2004	(*) 9,420	(*) 166,129	(*) 26,522	(*) 9,351	(*) 162,570	(*) 27,329	(*) 365,463	
2005	(*) 9,652	(*) 168,262	(*) 30,378	(*) 8,466	(*) 166,941	(*) 29,054	(*) 343,827	
2006	(*) 11,029	(*) 172,861	(*) 37,839	(*) 10,331	(*) 173,694	(*) 29,177	(*) 316,380	
2007	(*) 11,736	(*) 169,902	(*) 41,912	(*) 9,463	(*) 166,682	(*) 38,816	(*) 343,331	
2008	(*) 11,214	(*) 188,550	(*) 44,599	(*) 10,703	(*) 175,848	(*) 43,272	(*) 313,338	
2009	(*) 11,115	(*) 182,755	(*) 38,672	(*) 10,132	(*) 180,476	(*) 40,465	(*) 339,685	
2010	(*) 13,170	(*) 196,156	(*) 46,393	(*) 12,447	(*) 189,444	(*) 43,198	(*) 368,548	
2011	(*) 11,071	(*) 187,792	(*) 39,785	(*) 12,034	(*) 169,149	(*) 42,060	(*) 310,084	

The same will be for the chart and map tabs in the following implementations.

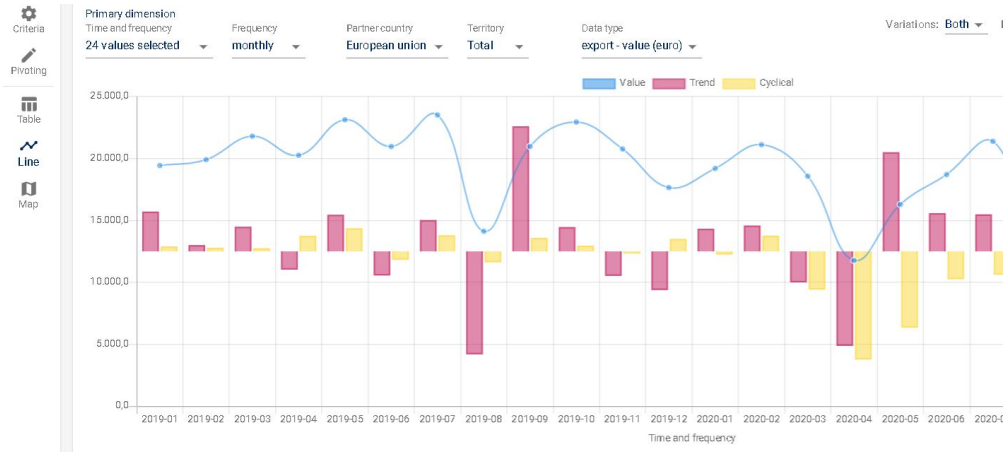
If the user chooses to enable variations, on the top right page, a small panel appears where the use can choose whether to show variations as trend, as cyclical or not show them at all.



Indicator type		Number of students														
Gender		Total		Male		Female		Not applicable								
School class		Not applicable		Not applicable		Not applicable		Seventh year of teaching basic public		Eighth year of teaching basic public		Ninth year of teaching basic public		First year of secondary public education		Seco seco educ
Variation		Value	Trend	Value	Trend	Value	Trend	Value	Trend	Value	Trend	Value	Trend	Value	Trend	Valu
Frequency	Time period															
Monthly	2013-01	ND	ND	ND	ND	ND	ND	ND	ND	ND	ND	ND	ND	ND	ND	
Annual	2012	908,600.0	ND	418,498.0	ND	490,102.0	ND	191,157.0	ND	146,056.0	ND	125,828.0	ND	126,352.0	ND	1
	2013	887,445.0	-2.3	408,292.0	-2.4	479,153.0	-2.2	191,956.0	0.4	143,949.0	-1.4	122,635.0	-2.5	126,503.0	0.1	

If the user chooses to visualize the data with a bar graph, the variations will be shown as lines. Two lines will be drawn if we choose to visualize both variations, or if we have only one variation but there is also a secondary dimension in the graph.

Otherwise, if the user chooses to visualize the data with a line graph, the variations will be shown as bars with the same logic described above.



If a specific data already has a template, if the administration user decides to save new configurations, these will overwrite the previously saved ones. There will not be a new template saved for the same data.
It is possible to see saved templates by selecting the “Template” item from the configuration node settings.

The new window shows the information about the saved template (which node is part of, the data ID, description) and also actions the user can perform (visualize the template, delete it).

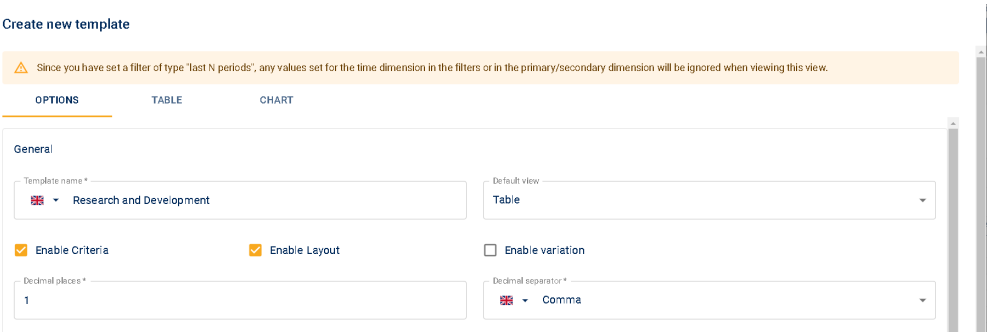
It is important to notice that if a template is set for a specific data, once this data is opened, the user won't see the criteria window and most of all there will be an information icon, on the top right of the table, saying that there is a template applied

If the template needs to be changed and modified, this is also possible. The user with template management permissions, can modify the table and then click on the save button and select "Update template".

If a filter of type “last periods” is set in the criteria on the temporal dimension, then the templates will always show:

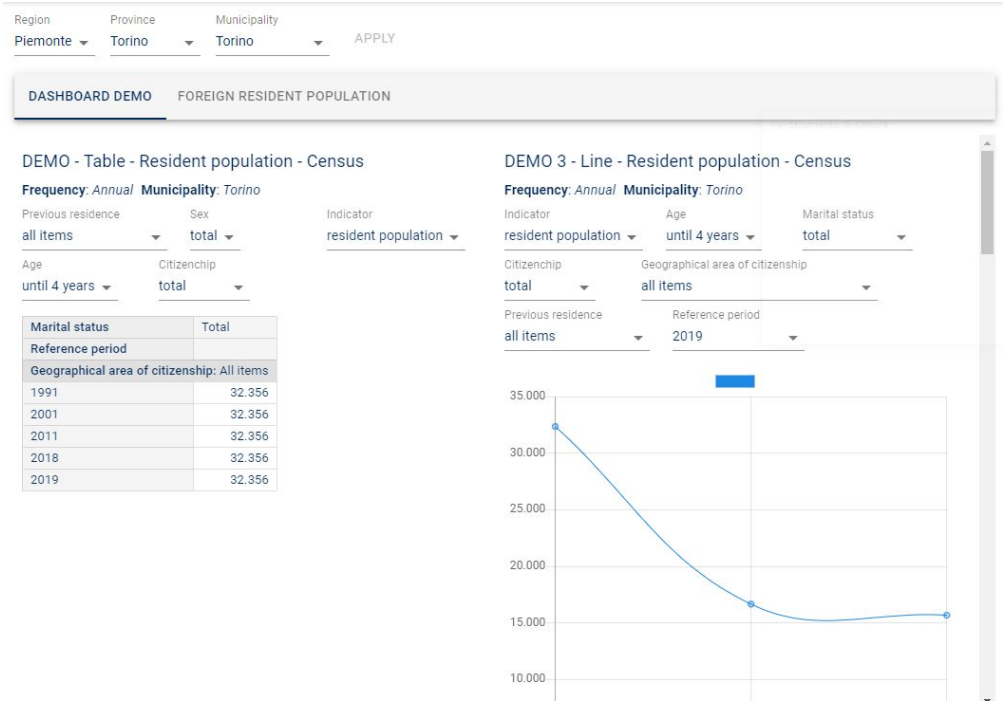
- the last available period in the data, if the temporal dimension is inserted in the filters of the multidimensional table
- all the values present for the data, considering the last N periods, if the time dimension is set as primary or secondary dimension of the graph

even if the current view that the user is saving is different (e.g. a period other than the last one is filtered).
The user who creates this template will be warned at save time that the values of the filter or the primary/secondary dimension he has set will be ignored at display time and will be asked to set a “custom range” policy if he wants to preserve those values.

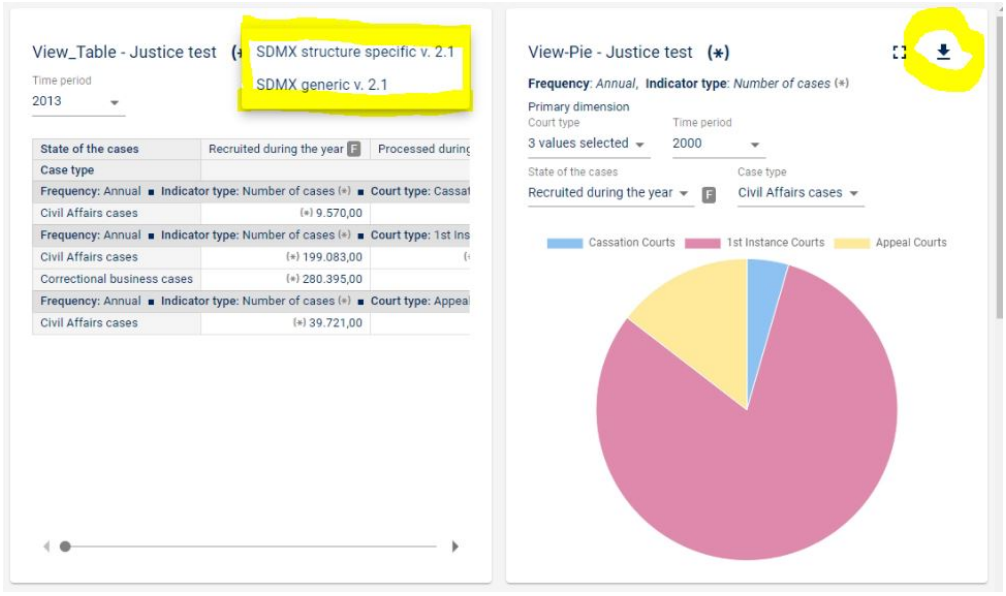


4.5 Dashboard management

Dashboards are groups of views and text that the user can put together in the same page. For example:



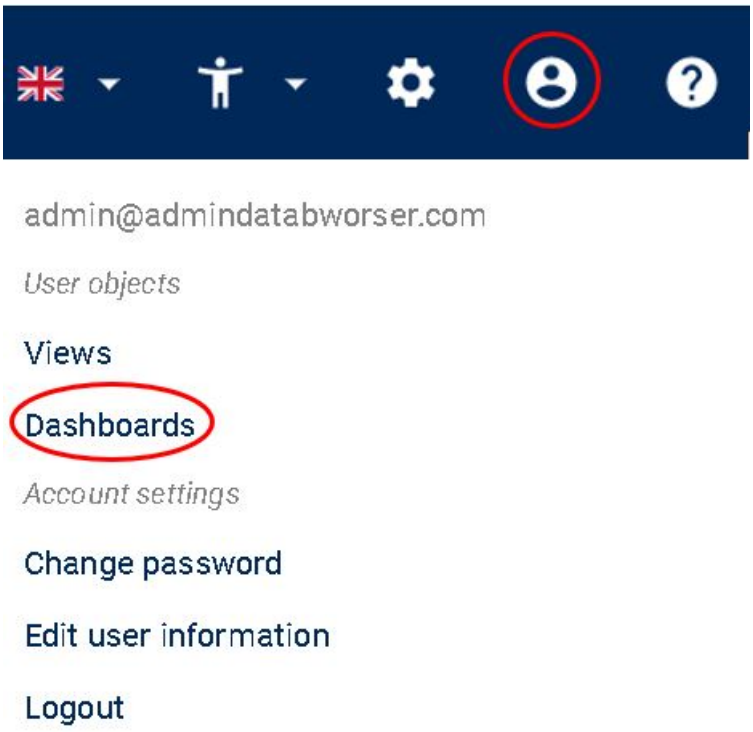
Most important operations on dashboards are ***creation*** and ***management***. The super administrator user and also a node administrator user with configuration permission, have all powers on creating dashboards on node level. They can change dashboards' order or delete them. However, only the super administrator can set dashboards on the application level. All elements in dashboards can be independently downloaded (by clicking on the export icon) and also viewed in *fullscreen* mode.



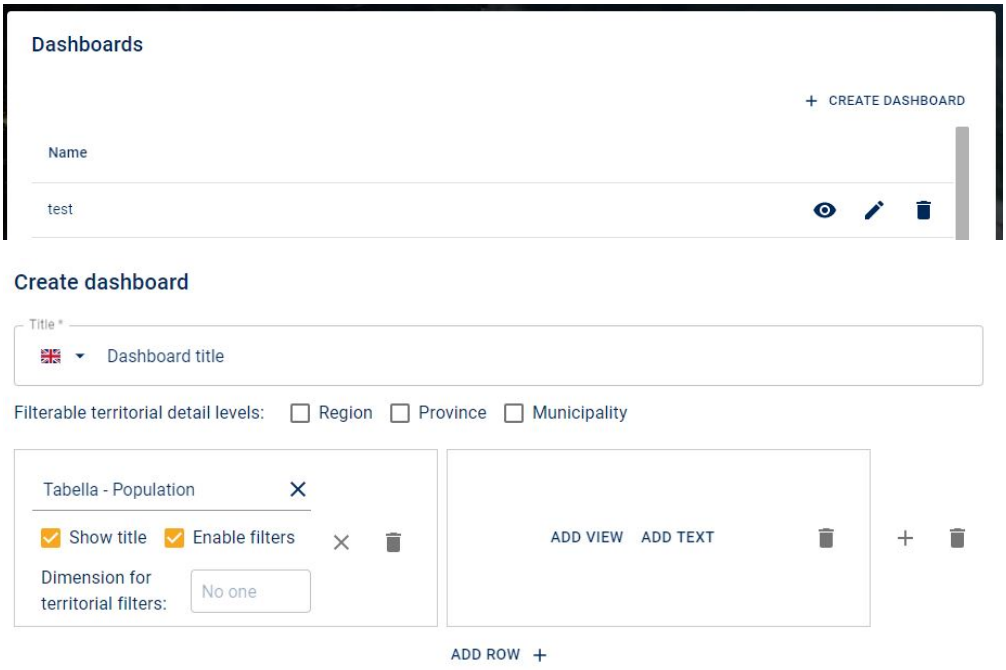
Let's give a more detailed description of what happens when dealing with dashboards.

Dashboard creation

To create a dashboard, the user must click the user icon and select "Dashboards"



From the window that appears, there will be the list of the existing dashboards which can be visualized, edited or deleted, and on the top right by clicking "+ CREATE DASHBOARD" it will be possible to create the new object.



There are two types of dashboards that can be created: ***filterable*** and ***non-filterable***. The difference between these two consists in the possibility to set a territorial filter (by writing in the text box the respective dimension contained in the data table, e.g. ITTER107) which allows the user to change the territorial dimension once visualizing the dashboard.This makes the dashboard dynamic and flexible. Of course specific cache management is necessary in order

to retrieve short response time since we are dealing with territorial dimensions which consider a big amount of items and information. For more detailed information on this matter check the section *Software package* under the **SPECIAL CACHE MANAGEMENT** paragraph.

Another important funcionality is the multilingual functionality which allows users to set different titles depending on the selected language. This is made possible by selecting the flag related to a language and by defining a title for that language.

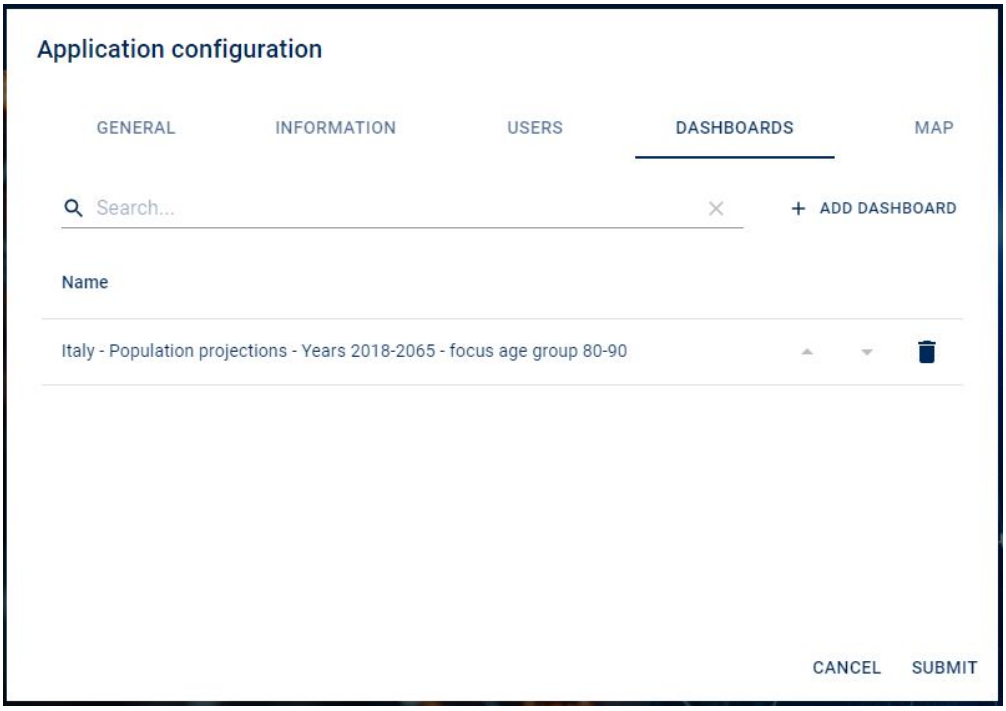
If a view is inserted in a dashboard in which a filter of type "last periods" is set in the criteria on the temporal dimension, then the view within the dashboard will always show

- the last available period in the data, if the temporal dimension is inserted in the filters of the multidimensional table
- all the values present for the data, considering the last N periods, if the time dimension is set as primary or secondary dimension of the graph.

Dashboards on application level

The super administrator user has permissions on adding dashboards on the landing page of the application. These dashboards will also be present at node level.
If dashboards already exist, they will be visible to the user by clicking on the "Dashboard" button on the landing page:

To add new dashboards in this section, the super administrator user opens the "Application" configuration window from the main menu (settings icon) and, under the "Dashboard" tab, adds one or more of the created dashboards.

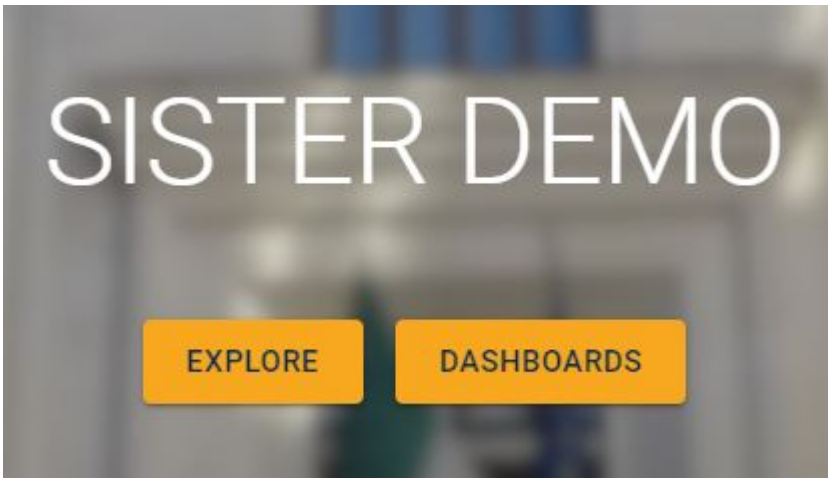


From the "Dashboard" tab he can always delete the dashboard from the landing page or change the order of appereance.

Dashboards on node level

Both the super administrator user and the node administrator user with configuration permissions can manage dashboards at node level. If dashboards are already set for a specific node, the mosaic icon will appear on the main menu of the page,

and also on the node home page (like it happens in the landing page)



Otherwise in order to add an existing dashboard to the node, the user must open the "Node" configuration window, click on "+ ADD DASHBOARD" on the top right and select the particular dashboard he wants to add.

CHAPTER

FIVE

DATA BROWSING

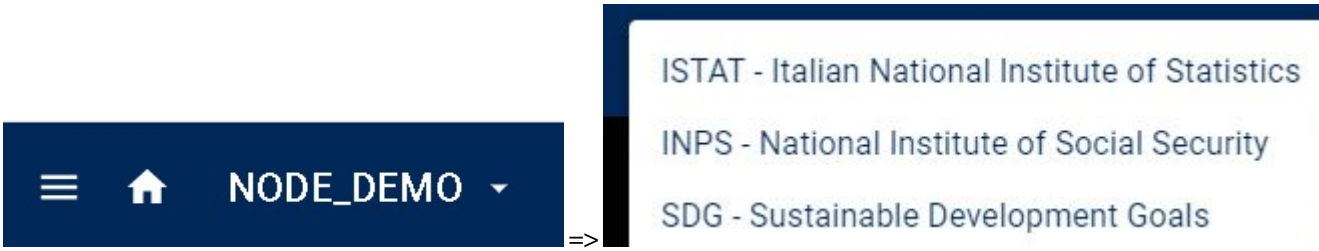
In this chapter we'll see how to:

- select a data provider (node)
- explore node's contents through categories tree
- find specific data
- display data in tabular, graphic and map format
- configure visualization and modify display layout

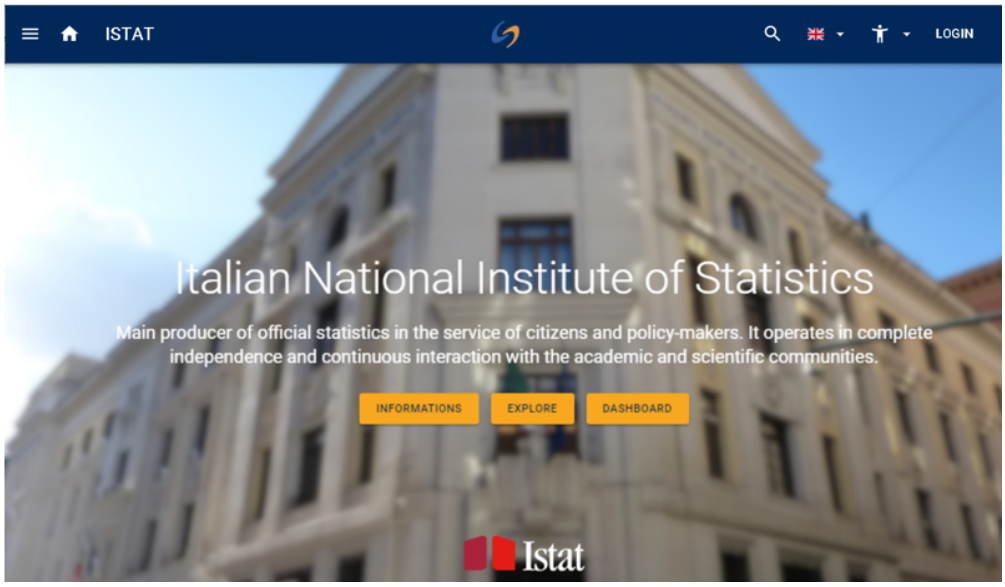
5.1 How to browse different data providers

The landing page shows information about default node (if set). From the landing page the user can choose to browse a different node by selecting the data providers in the section **Other nodes** both by clicking on the images or the descriptions.

Another way of reaching a different node from the default one, is by selecting the node from the top left menu of the page and choosing from the pop-up list.



Once the node is clicked, the Public Statistics Hub will show the node's homepage including: information, data grouped by categories and dashboards.



From a node's homepage it is possible to browse a different data provider by clicking again on the top left menu and selecting a different node from the list.

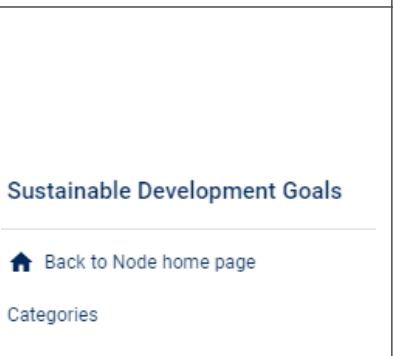
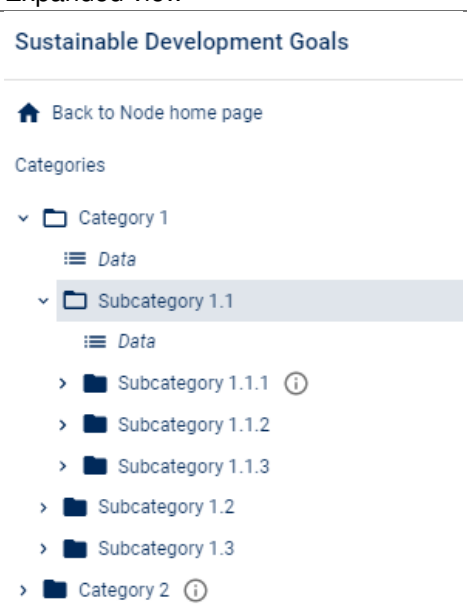
The "Home" icon will always be visible in the header of the application so the user can always easily return to the node's homepage.

5.2 How to browse a data provider

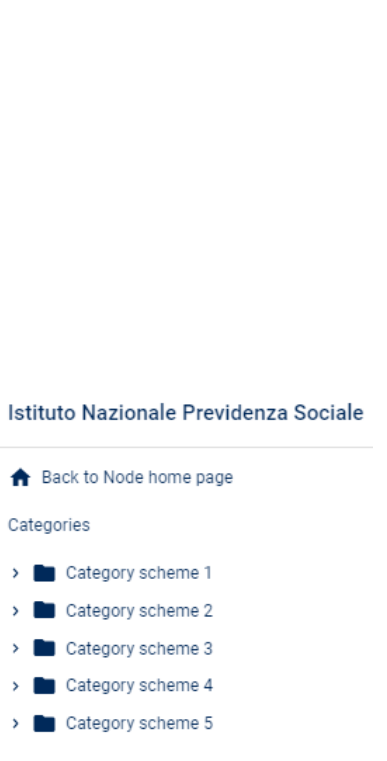
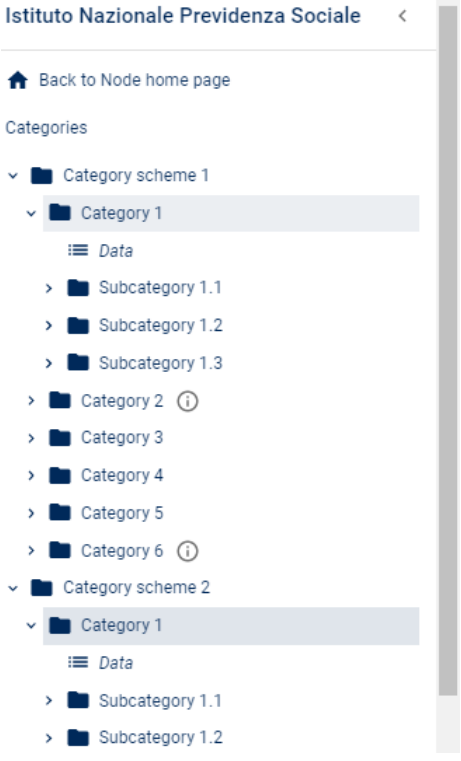
In this section we'll explain data's organization and how to search specific data.

5.2.1 Data catalog

All data contained in a node is grouped by categories organized in a catalog. The catalog of a node can be displayed by clicking on the burger button on the top left . Categories have an hierarchical structure: a category can contain one or more subcategories; a subcategory may in turn contain other subcategories, and so on. It is possible to expand or collapse a category or a subcategory by clicking on it.

Initial view	Expanded view
	

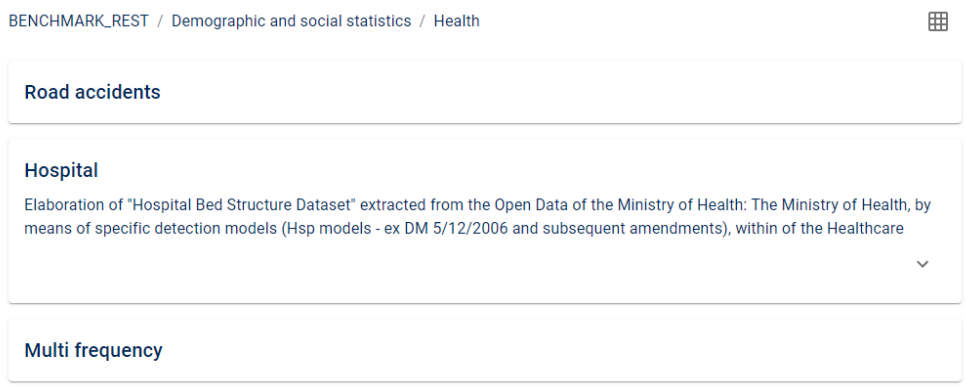
All categories belong to a category scheme and one node can have more than one category scheme; in this case the same data can belong to more than one category in different category schemes. Moreover the same data can belong to different categories in the same category scheme. If a node has just one category scheme, it isn't shown in the tree visualization of the catalog. If there are more than one category scheme in the node, only the root nodes (i.e. category scheme names) will be shown without showing the categories.

Initial view	Expanded view
	

By clicking the information icon next to a category, it is possible to see notes associated with the category. By clicking the **m** icon, instead, the user can see metadata as well. The leaves of the catalog are nodes of Data type. By selecting this kind of node on the right side of the panel, all the data belonging to the selected navigation path will be shown. Once data is shown, the user can decide to display the list of data in two different modes: by rows or by cards changing the icon on the top right.

For each data the user can see the title and the description. By clicking on the information icon next to the data it is possible to see notes associated with that data and by clicking on the **m** icon the user can see metadata as well. By clicking on the data title, results can be visualized in tabular, graphic and geographic format. Attachments might be present in the data visualization list, this means that data attachaments are available regarding that information. Depending on its native format, not all data can be displayed.

Example of list of data for a specific category:



5.2.2 Textual search

It is possible to find specific data by textual search. Even if the user doesn't know the exact name of the data, he can type some text in an input box and the system will search for all data containing the typed text in titles, descriptions or keywords. The application will complete text for data titles; all data belonging to the search will appear to the result set. Search text can be inserted by clicking the magnifying glass on the top bar menu:

In *future releases* it will be possibile to also search by using the search input text box on the top left of the page where all data belonging to a category or a subcategory is displayed.

When the searched data appears, the user can click on the title and the default visualization will open.

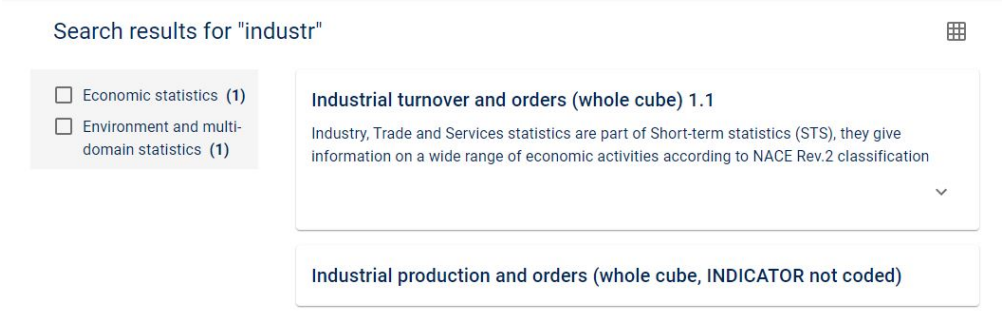
KEYWORDS

Keywords are specific words linked to the content of the data which are included in the uploaded dataset in a node. It is no mandatory to have keywords, so they might not be present in all data contained in a node.
You can search data using keywords but they are not listed in the pane that describes the specific data.

User can search for keywords by clicking the magnifying glass on the top-right of the page, and all related datasets containing that specific word will appear in the page.

FILTERING FOR CATEGORY

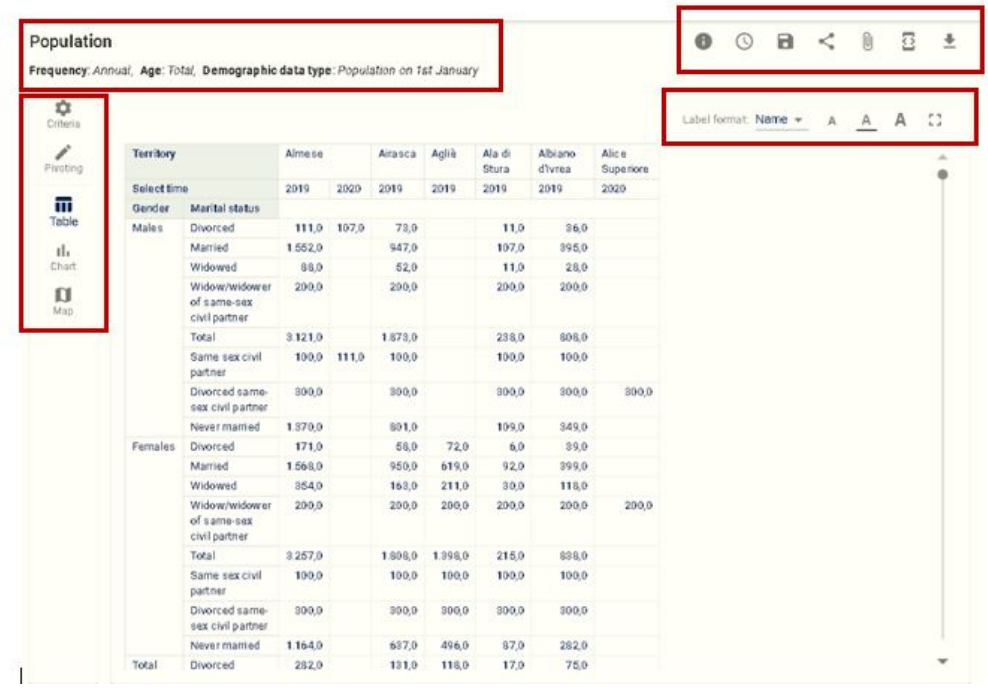
In case there is more than one category scheme involved in the research results, two levels of category (category scheme at first level and at second level) will be shown. If there is only one category scheme, results will be filtered only for the first level categories of the only category scheme.
Shown categories, will be the ones for which there is at least one research result between its children.



5.3 How to visualize data

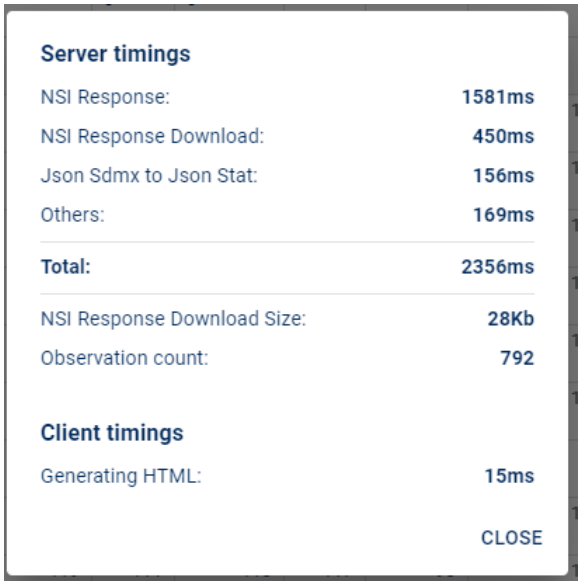
In this section we'll explain how to configure data visualization, how to change layout, how to visualize data in different ways and download information.
On the top of the table, the name of the dataset is shown and right under it, dimension containing one single item are displayed.
In the central part of the visualization window it is possible to directly access the information contained in the data. On the left side, a side bar containing several buttons allow the user to change configuration and layout, to view metadata and to change the type of data visualization among table, chart and map. On the top right page, two small panels make the user able to change font size, choose a full-screen view and to save the visualization, to share it, to download attachments and data.
Most importantly, the "Label format" menu allows to change the labels of the dimensions in the table. Possible options are:

- Name: shows the names of the dimensions
- ID: shows the ids of the dimensions
- Both: shows ids and names of the dimensions

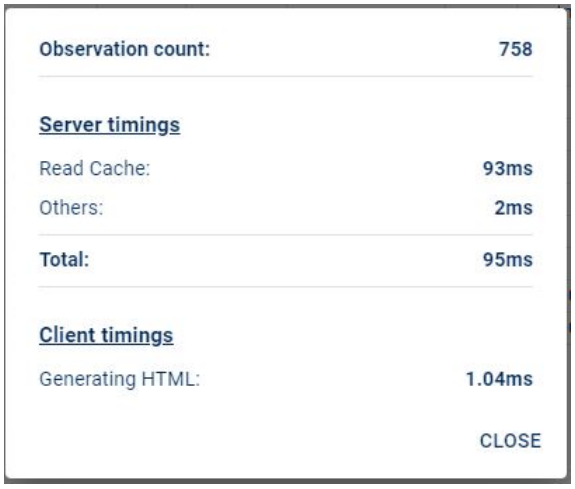


Once the data is visualized, it is possible to check interaction and responses timing between node and servers by clicking on the clock button on the top left menu of the page:

This is a possibile output once the button is clicked:



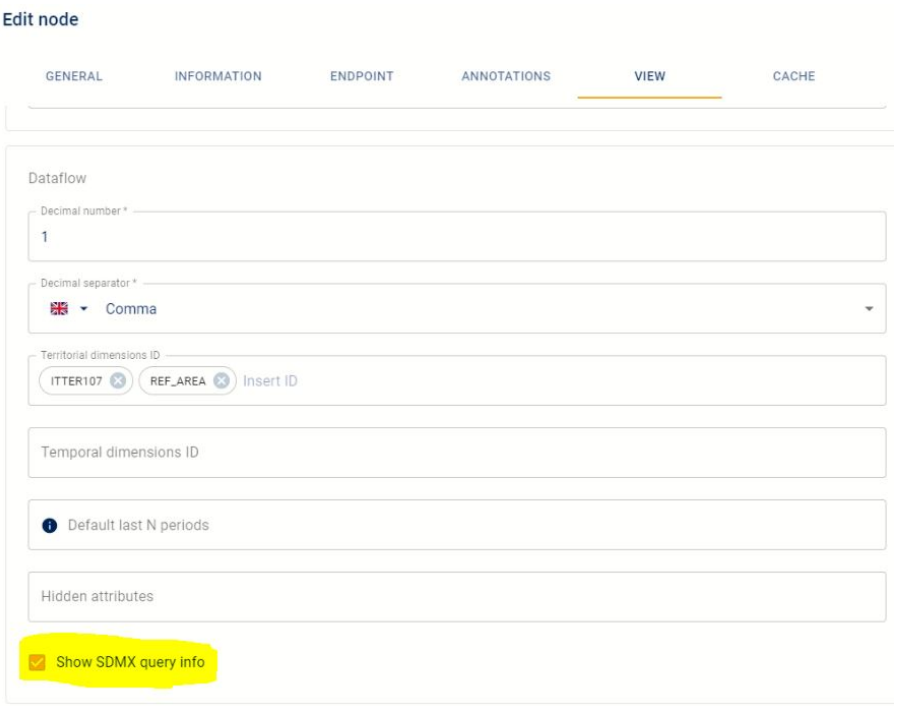
If the file was already available in cache, response calls will be different compared to the first time and also times will show new entries. This is an example of time logs returned when a cached data is requested:



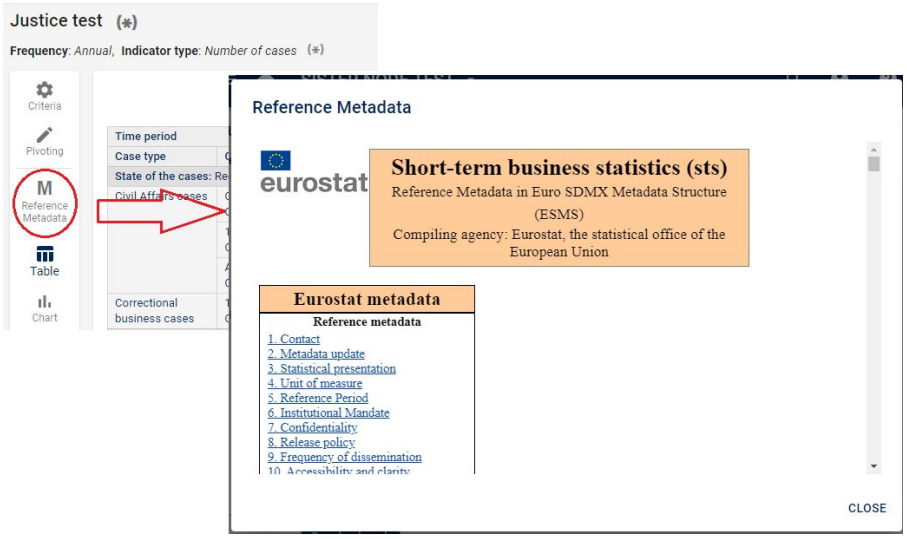
Once the data is displayed, you can do the query download to get data and artifacts:

This is a possibile output once the button is clicked:

This possibility is configurable in the administrative section of the node by the administrator user.



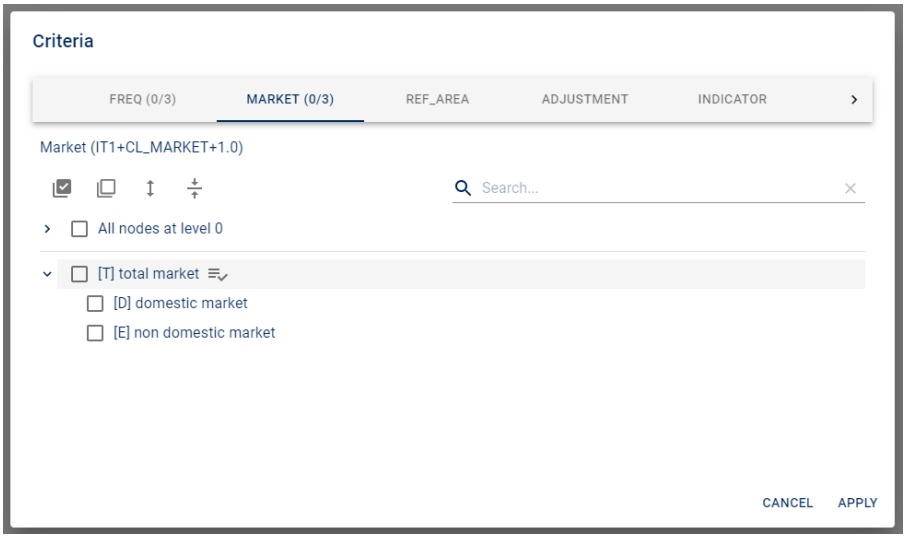
If data contains referential metadata, there will be a button on the left side menu that, once clicked, opens a pop-up window showing the information:



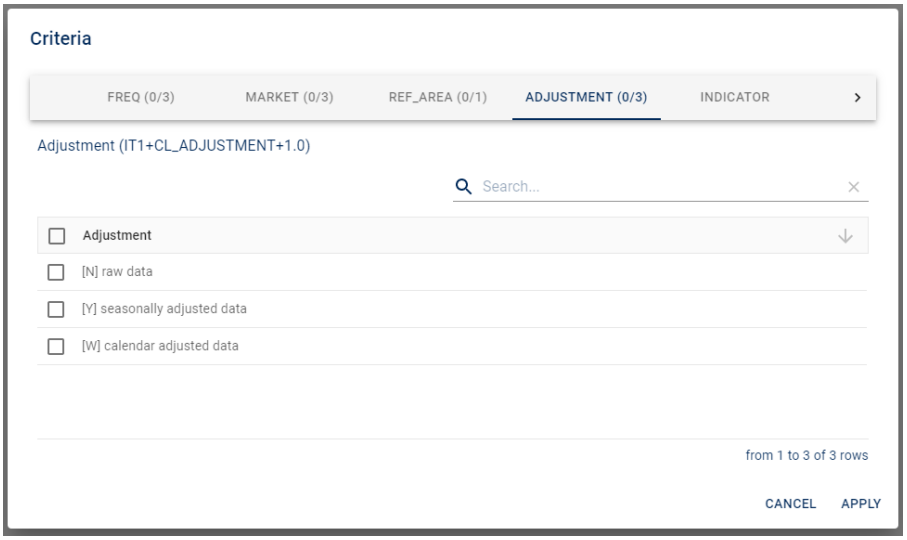
5.3.1 How to manage data criteria

By clicking on the **Criteria** button in the visualization window, a page opens where users can manage what information will be shown in data visualization. For each dimension the system will show available values in a hierarchical way (if a hierarchy is defined) and the user can select the values to filter. By clicking on **Apply** the system will retrieve data according to the filters set, and just the result set will be shown to the end users. The set query must return at most the maximum number of observations (configurable at single node level), beyond which the user will be asked to set more restrictive conditions.

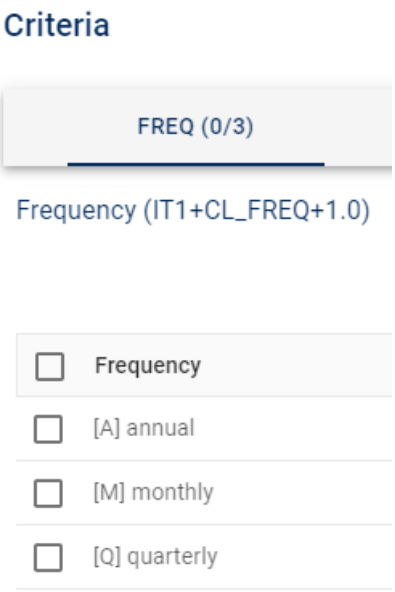
Example #1:



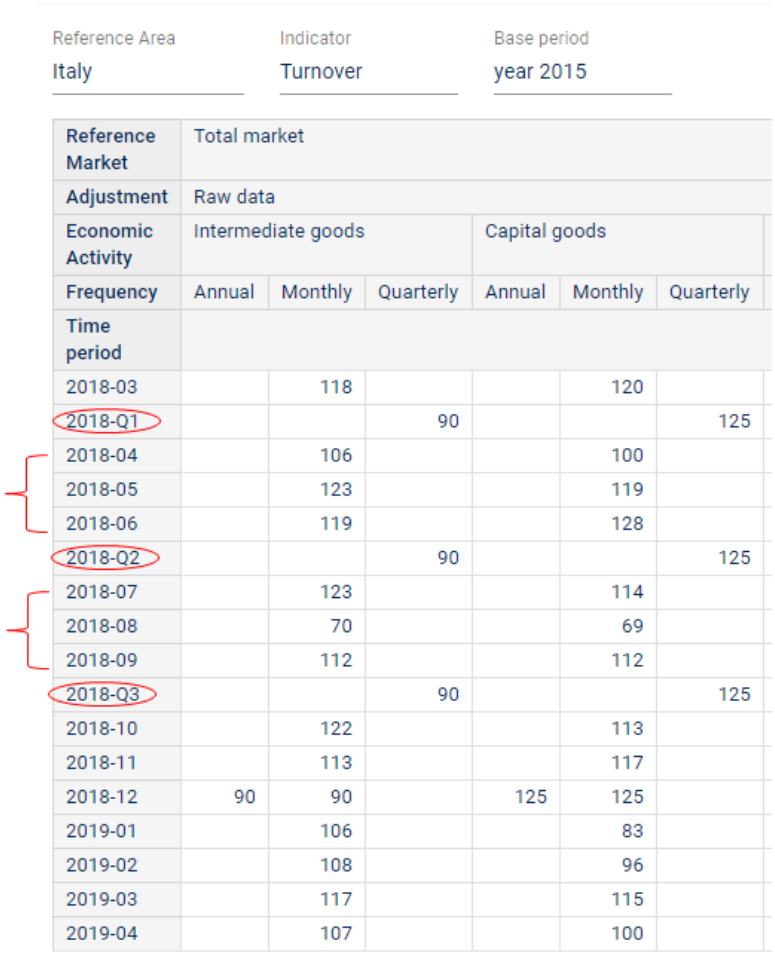
Example #2:



A particular focus can be set on the visualization of **time_period**, especially on its ordering and how it is managed. If data has multiple frequencies such as:



the output table as default, will return the time_period ordered in a "hierarchical" way. Considering the previous example, starting from the the first month available, in order, quarters, semesters and year will follow, as shown in this image:



Reference Market	Total market					
Adjustment	Raw data					
Economic Activity	Intermediate goods			Capital goods		
Frequency	Annual	Monthly	Quarterly	Annual	Monthly	Quarterly
Time period						
2018-03		118			120	
2018-Q1			90			125
2018-04		106			100	
2018-05		123			119	
2018-06		119			128	
2018-Q2			90			125
2018-07		123			114	
2018-08		70			69	
2018-09		112			112	
2018-Q3			90			125
2018-10		122			113	
2018-11		113			117	
2018-12	90	90		125	125	
2019-01		106			83	
2019-02		108			96	
2019-03		117			115	
2019-04		107			100	

It is also possible to select the range for time_period choosing between the minimun and the maximum date present in the data table.

Or in some cases, if time can be separated in periods, user can choose the period to visualize

If a default is defined in node configuration or in dataflow annotation the text box will be set with this value.

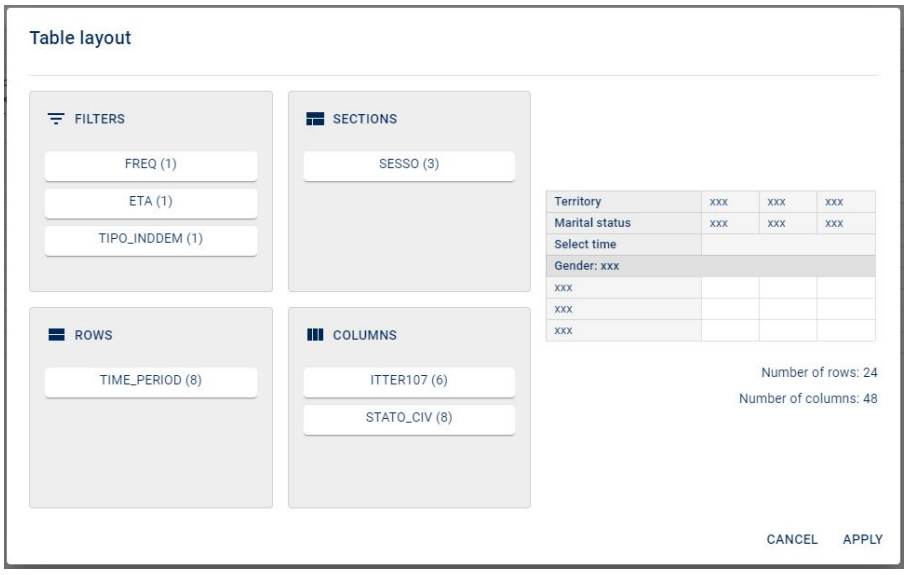
The authenticated user can select “Full range” to not apply filters on the time dimension. This option is useful if a view is present and data is update with new periods. If “Full range” has been chosen to build the view this will be updated with new periods otherwise the view will have just periods selected as custom range or last periods. The “Full range” option is not available for anonymous users as they cannot build views.

5.3.2 How to customize a table

By clicking on the **Pivoting** button opens a page where users can change the layout choosing which dimensions will be shown in rows, in columns, in sections or used to filter data.

The layout page interface changes according to the type of active visualization selected among Table, Chart and Map. If the active visualization is Table, information will be shown in a multidimensional table with the dimensions arranged on the axes according to layout settings, in descending order of priority. When the user chooses how to arrange dimensions among axes, sections and filters, a preview of the resulting table will be shown on the right of the layout page. Filters are shown on the top of the multidimensional table: if a dimension has only one value (fixed data) it will be automatically included in filters. Dimensions in rows and columns will be shown according to the order set in the layout pane. Choosing to show a dimension as row_section means "breaking" the table into many subtables for all the different values of that dimension. Some layout configuration examples.

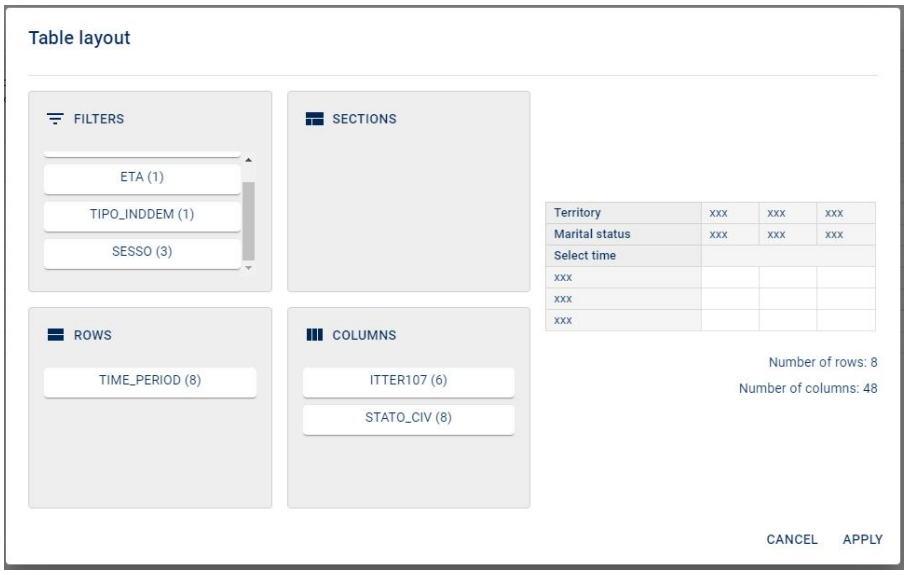
Example #1:

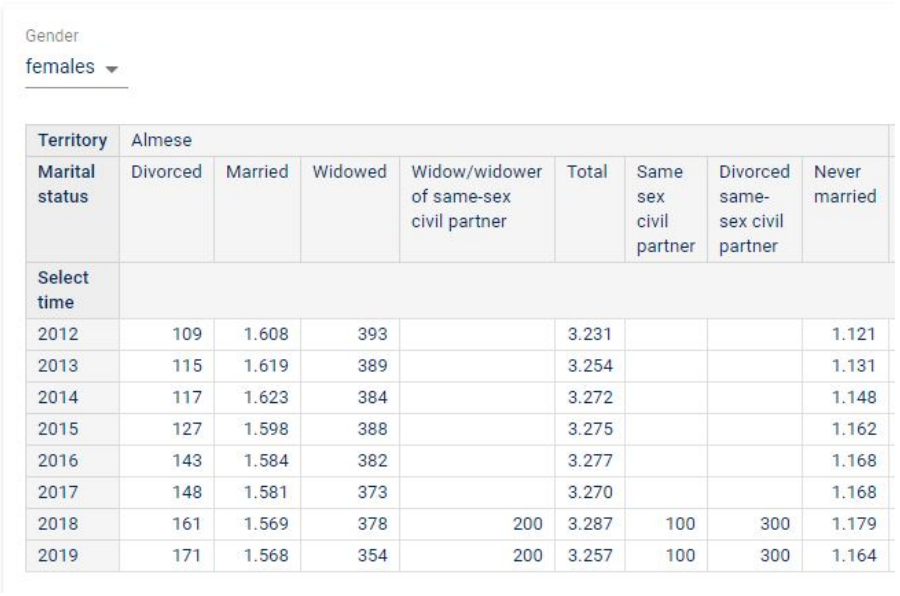


Territory	Almese							
Marital status	Divorced	Married	Widowed	Widow/widower of same-sex civil partner	Total	Same sex civil partner	Divorced same-sex civil partner	Never married
Select time								
Gender: Females								
2012	109	1.608	393		3.231			1.121
2013	115	1.619	389		3.254			1.131
2014	117	1.623	384		3.272			1.148
2015	127	1.598	388		3.275			1.162
2016	143	1.584	382		3.277			1.168
2017	148	1.581	373		3.270			1.168
2018	161	1.569	378	200	3.287	100	300	1.179
2019	171	1.568	354	200	3.257	100	300	1.164
Gender: Males								
2012	81	1.587	73		3.056			1.315
2013	88	1.626	73		3.123			1.336
2014	84	1.626	73		3.134			1.351
2015	92	1.612	77		3.133			1.352
2016	97	1.582	80		3.124			1.365
2017	110	1.563	89		3.153			1.391
2018	107	1.559	90	200	3.122	100	300	1.366
2019	111	1.552	88	200	3.121	100	300	1.370
Gender: Total								
2012	190	3.195	466		6.287			2.436
2013	203	3.245	462		6.377			2.467
2014	201	3.249	457		6.406			2.499
2015	219	3.210	465		6.408			2.514

=>

Example #2:




Gender
females

Territory	Almese							
Marital status	Divorced	Married	Widowed	Widow/widower of same-sex civil partner	Total	Same sex civil partner	Divorced same-sex civil partner	Never married
Select time								
2012	109	1.608	393		3.231			1.121
2013	115	1.619	389		3.254			1.131
2014	117	1.623	384		3.272			1.148
2015	127	1.598	388		3.275			1.162
2016	143	1.584	382		3.277			1.168
2017	148	1.581	373		3.270			1.168
2018	161	1.569	378	200	3.287	100	300	1.179
2019	171	1.568	354	200	3.257	100	300	1.164

=>

Data can have an information icon next to the title to show any notes associated with data. Notes can be also associated to dimensions and even to a single cell.

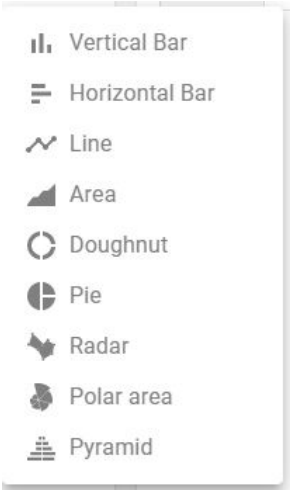
For more information see section *Attributes*

It will also be possible to access the reference metadata of the dataflow through a special button located in the bar on the left.

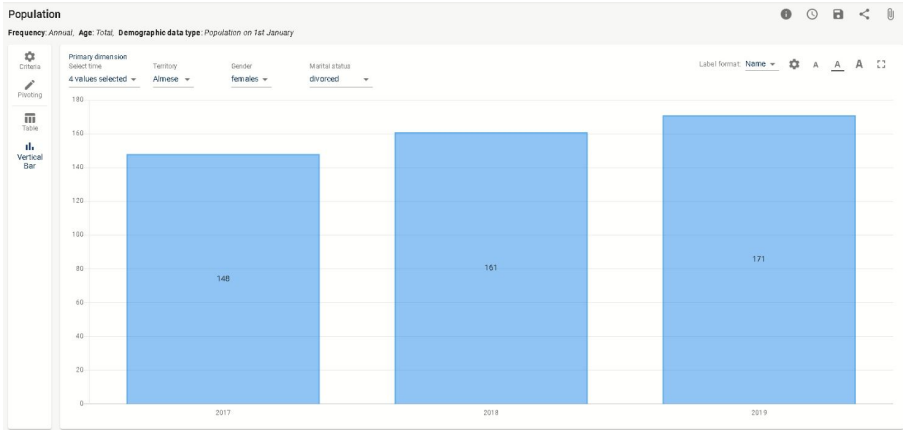
5.3.3 How to customize a chart

Once data is displayed in tabular format, it is possibile to create customizable charts by clicking on the chart symbol on the left pane of the table.

User can choose the chart type between the ones listed once the chart symbol is clicked



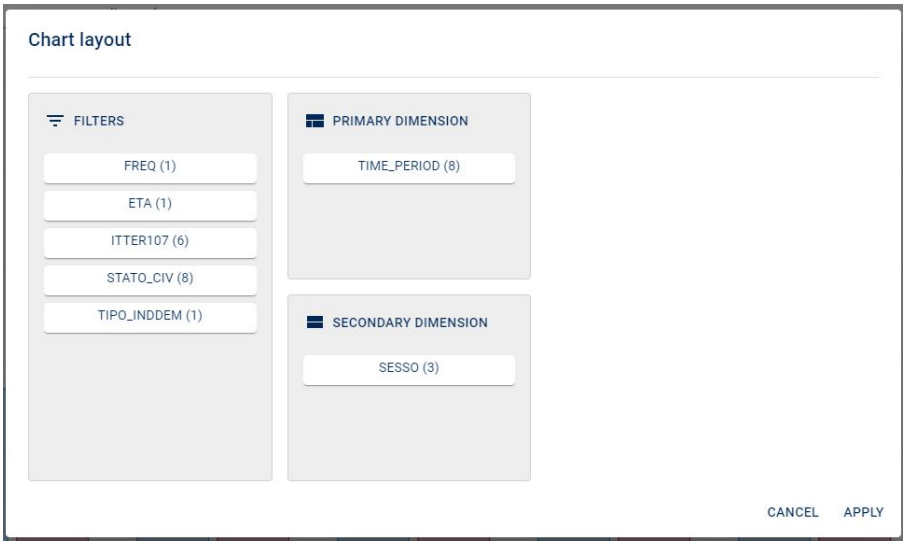
By default one primary dimension is set (usually the time_period) and all other dimension are contained in the filter section



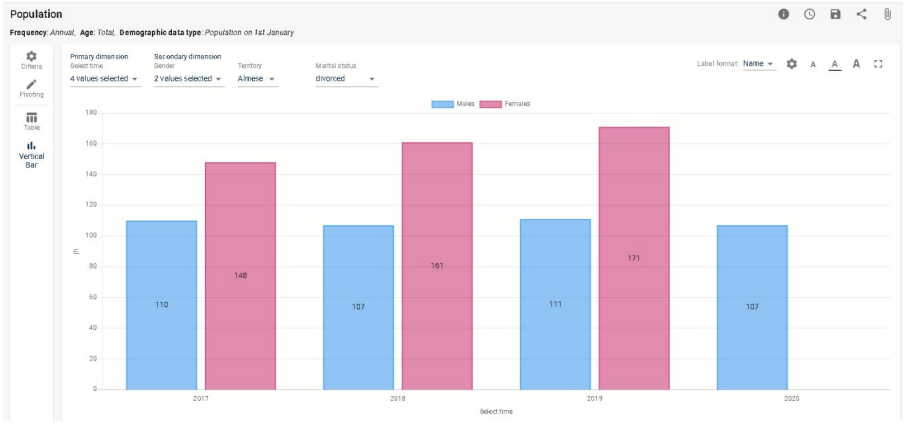
Obviously, these settings can be modified by clicking the layout button. This operation allows the user to select a secondary dimension to consider in the chart or move dimensions as filter

Example:

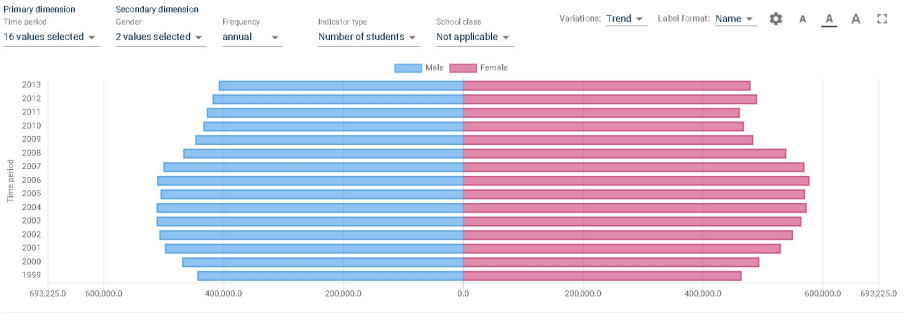
Setting the following layout



this is the returned chart



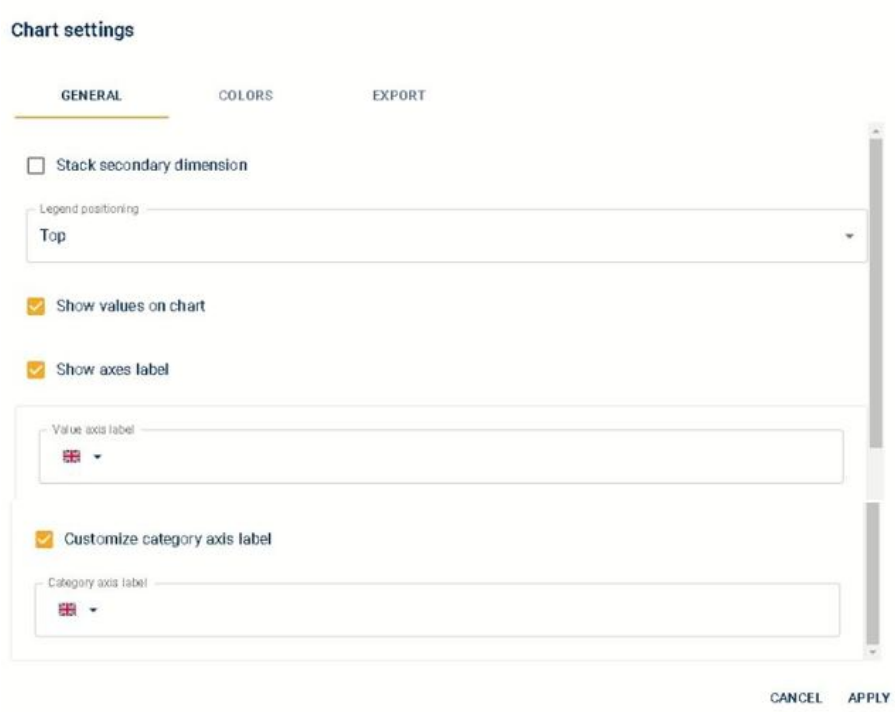
Depending on the type of chart some settings on layout are mandatory: e.g. for pyramid chart the user must choose both primary and secondary dimension, moreover secondary dimension must have just two values selected.



The main two functionalities of the chart section are that:

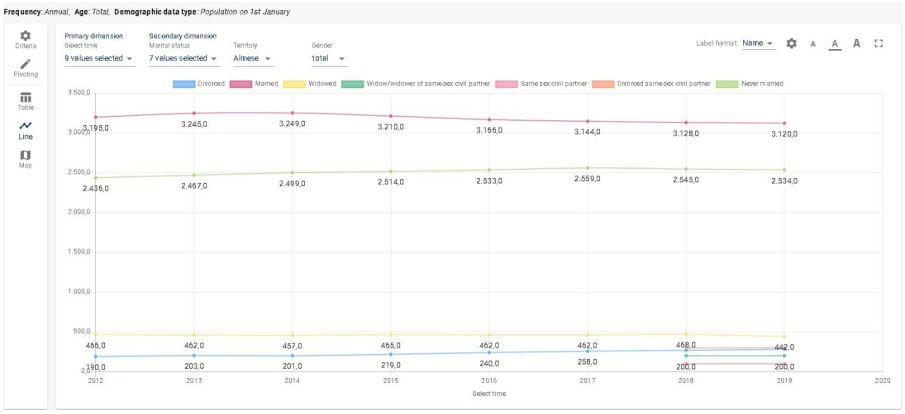
- by selecting the export icon a JPEG version of the chart is downloaded
- by selecting the dimension values in the chart caption, these can be excluded from the graph

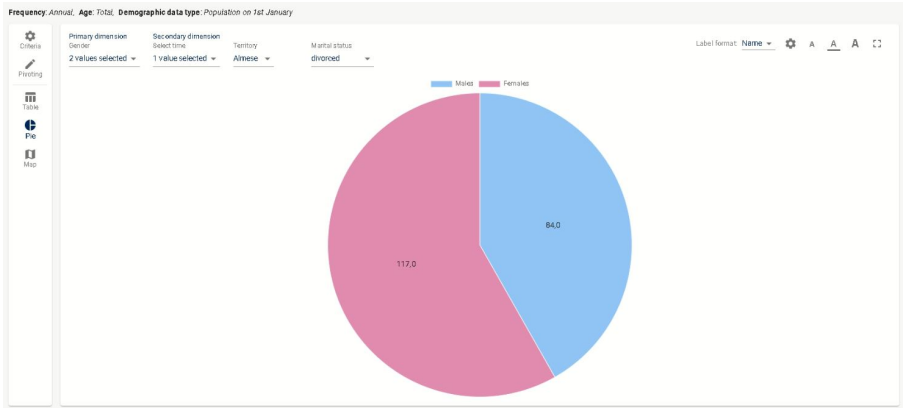
Another functionality added to the graphical visualization, consists in the possibility to change dimensions' colors, change caption's position and stack secondary dimension, if the graph allows it.
This configuration is reachable by clicking on the configuration icon on the top right of the graph. Once the icon is clicked the following window appears:



This window contains three tabs:

in the first tab, “General”, it is possible to set caption’s position (top, left, right or bottom) and the possibility to stack secondary dimension (in checked and the graph does not allow this configuration, nothing will change in the final visualization).

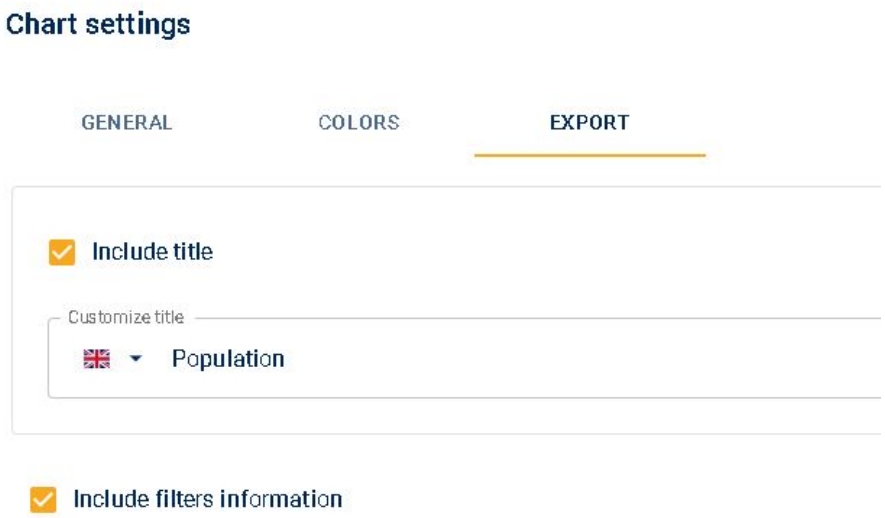




Moreover, by selecting “Show axes label” the user can type a label for the value axis in multiple languages, as well as customize category axis label which has category label by default.

In the second tab, “Colors”, it is possible to set colors to a dimension’s items just as shown in this example

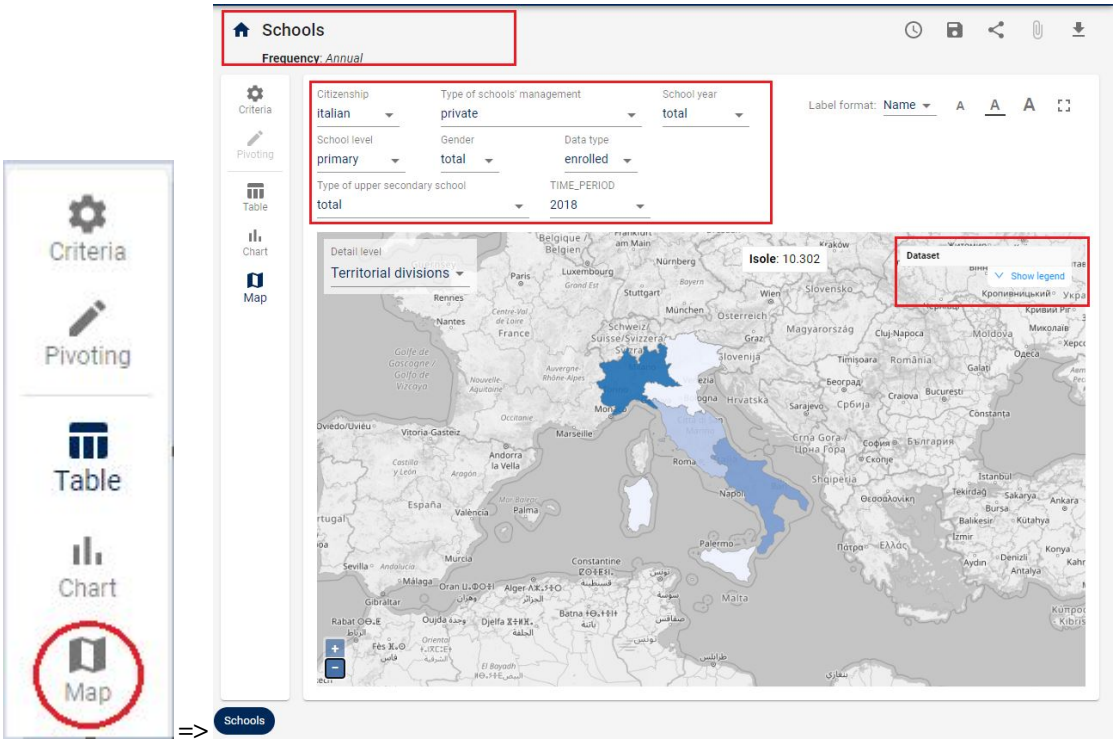
On the third tab, “Export”, the user can include title and filters for export, this is useful when exporting images to have information about the filters applied. These settings are inherited when selecting export in dashboards.



5.3.4 How to customize a map

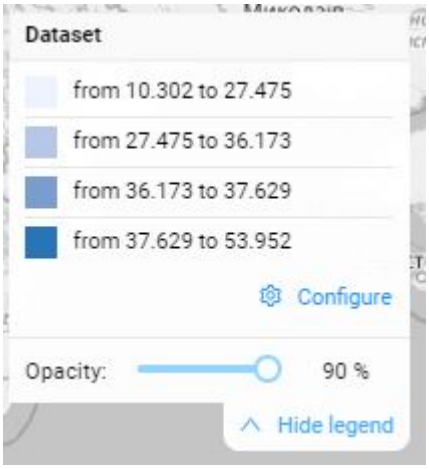
From the tabular visualization, it is also possible to move to the map section where a map is shown. Although the map button is not always present, in fact if in the node configuration the user does not specify territorial dimension's IDs, the application doesn't recognize any dimension for the map automatically. So first of all, in the node configuration window, under the "View" tab, the "Territorial dimensions Ids" must be filled with the territorial ids that it is possible to find in the datasets (for example: ITTER107, REF_AREA, COM and so on). If no territorial id is set in the node configuration, another way to visualize the map for a specific data is to set a geographic annotation when data is uploaded.

Once the user detects that the map visualization is available, he can click on the related button. If no criteria is selected initially, the dimensions present in the data are inserted as filters in the map visualization otherwise they are printed, with the chosen value, under the title of the data.

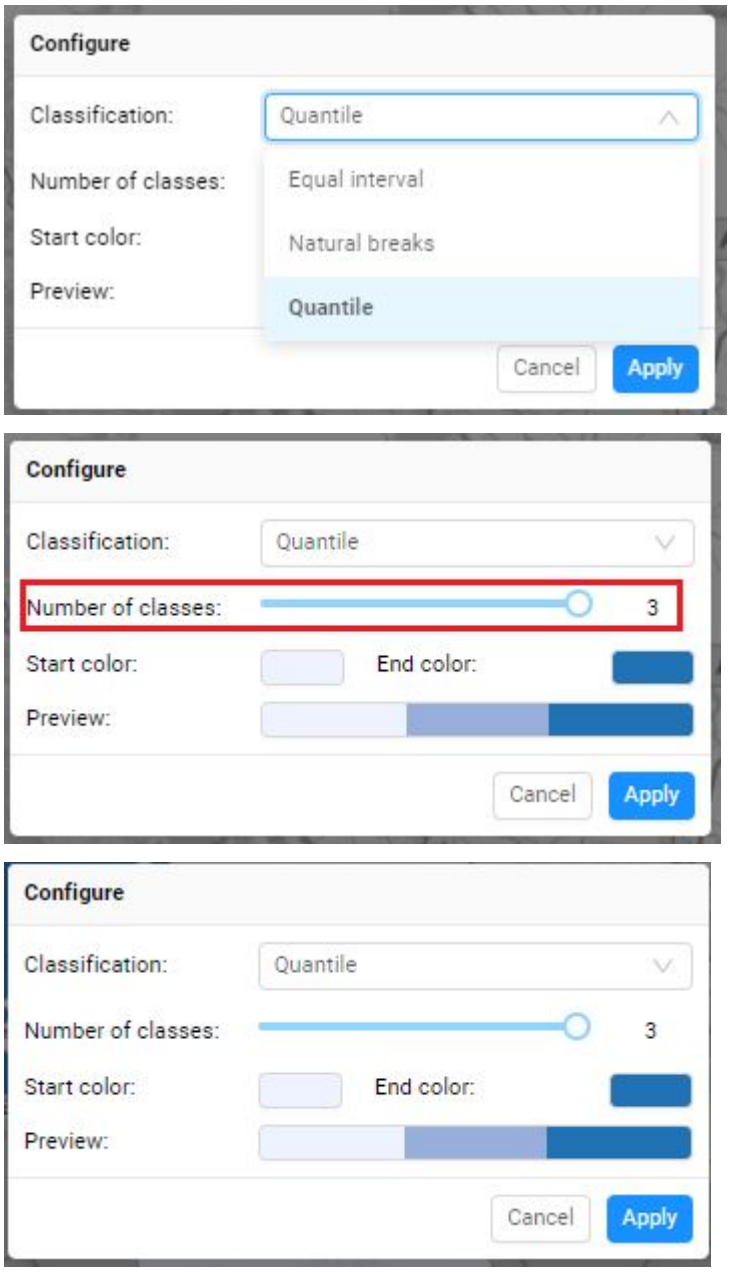


The dimension based on the territory, goes in a specific filter, labelled as "Detail level", which is categorized considering the geographic section (area, region, province, municipality) and user can drill down or drill up in the map.

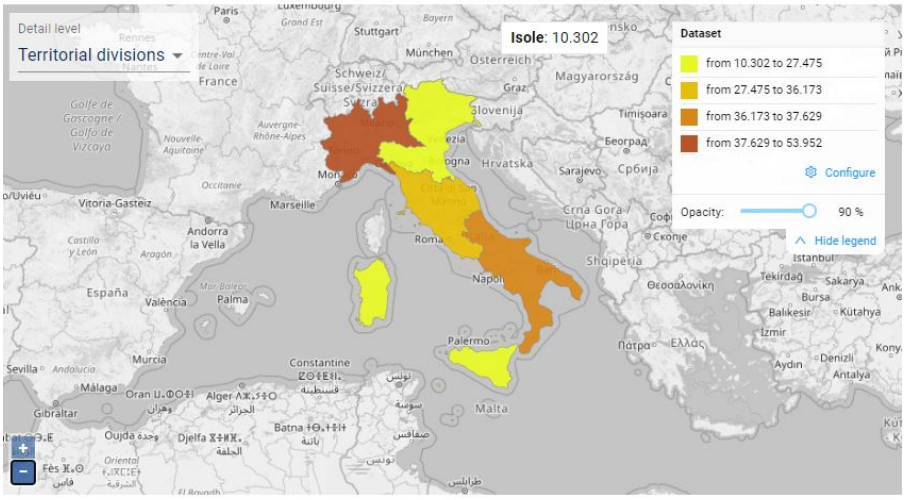
Map configuration can be personalized expanding the "Show legend" indication on the top right of the map which opens the following window



Opacity of the map can be selected by moving the scroll bar that shows the percentage of opaqueness. If, instead, the user decides to change the colors in the map, classification or number of classes in order to choose the way intervals are divided, this is made possibile by clicking on "configure".



Selecting “Start color” or “End color”, enables the user to change colors in the map and, right away, there is a preview of the new colors. Modifications take place right after the user clicks on “Apply”. The map will now have the new colors.



As in graphs also in maps is possible to configure "Export" deciding whether or not to include title and filters for export, the user can set "Export" configuration by clicking on "Map settings" icon.

5.3.5 Attributes

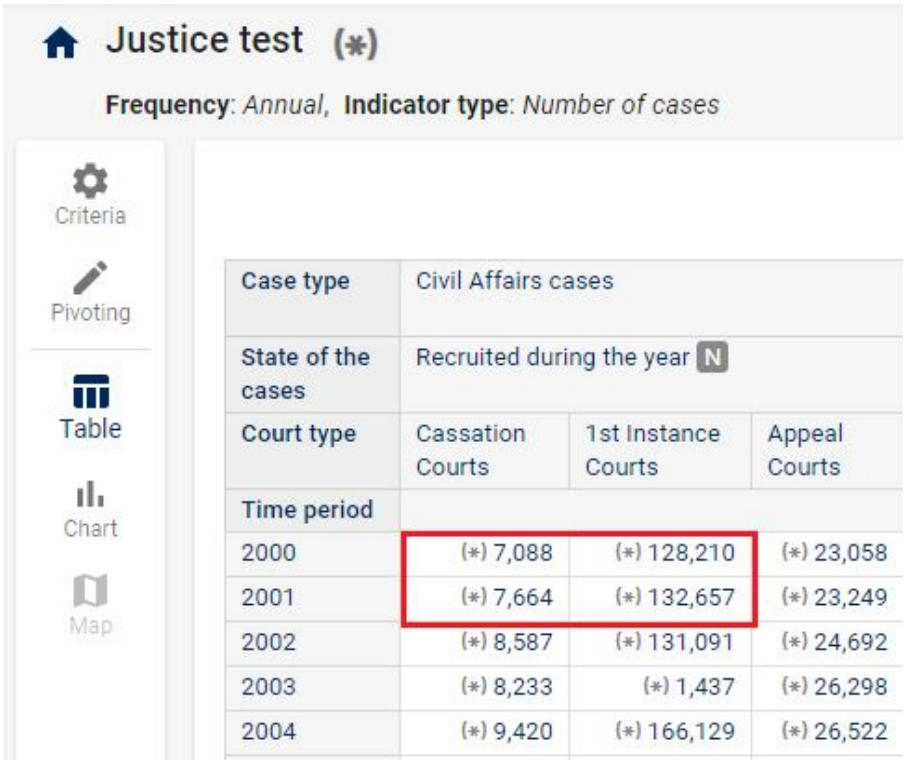
Data can have four attribute types:

- Dataset attributes
- Group attributes
- DimensionGroup attributes
- Observation attributes

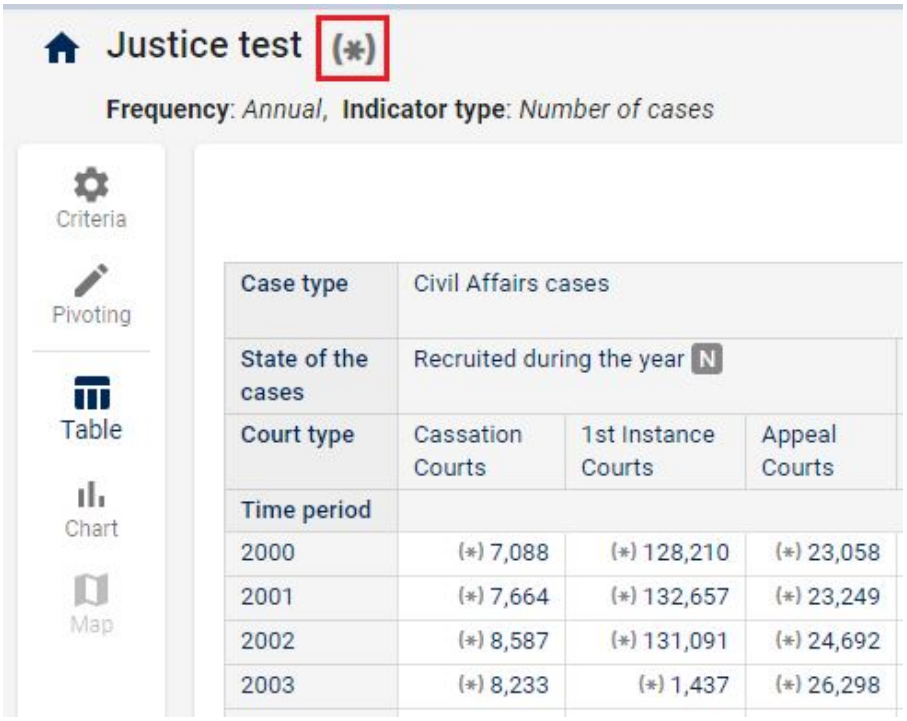
The user can see attributes in the same window where data is visualized.

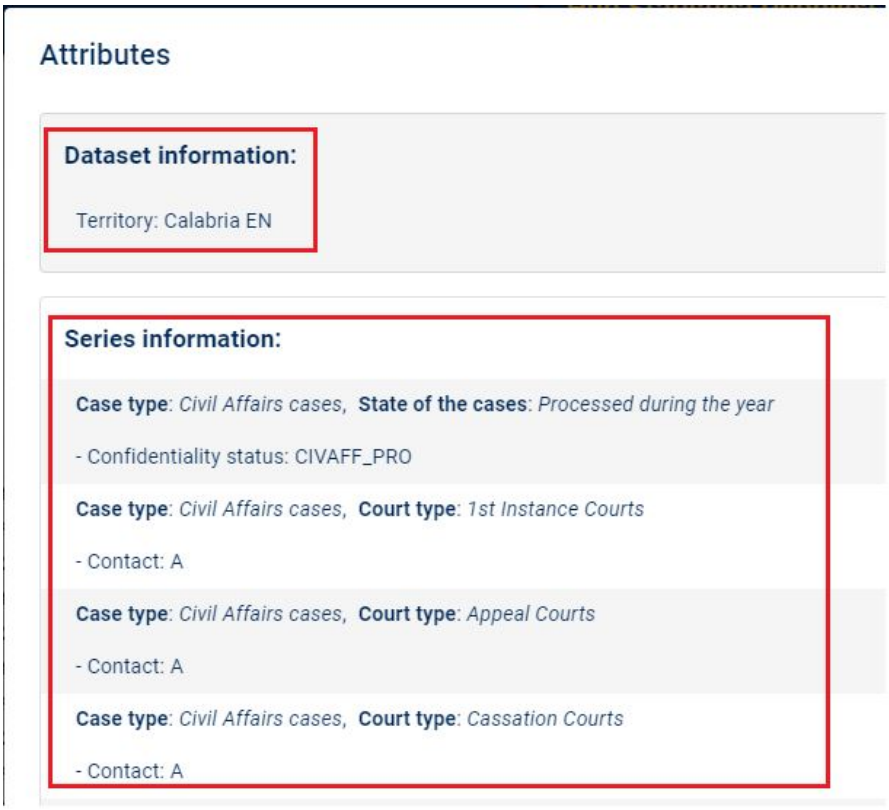
When an attribute is present, the application generally shows an asterisk. On mouse over the cell containing the attribute, a tooltip shows the attribute(s) and its value(s) and the user can copy it; a tooltip can contain more than one attribute.

For encoded attributes, if the code id is short it will be shown together with the asterisk.

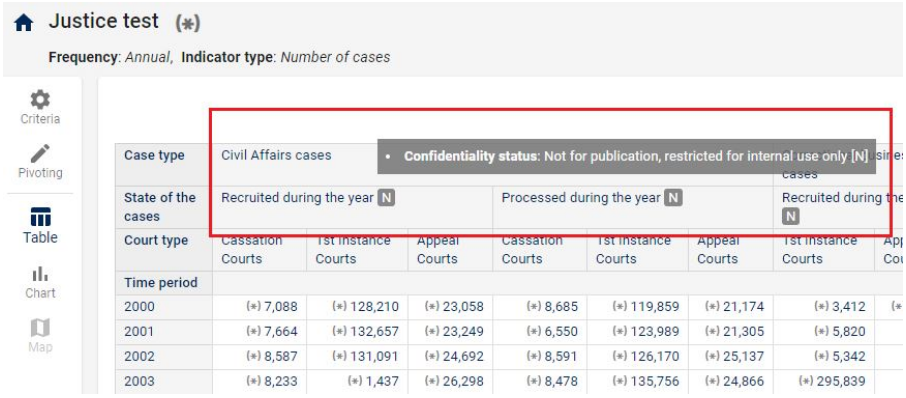


The user can see attributes at dataset level by clicking the asterisk between brackets - **(*)** - next to the title (Dataset level). If the dataset has attributes for more than one dimension (Group/DimensionGroup attributes) they will also be shown by clicking the same icon (Series level).

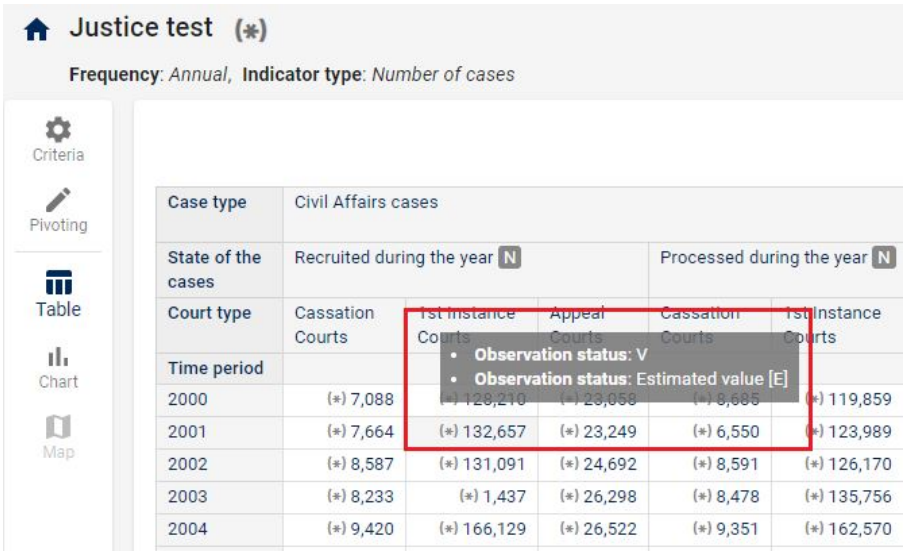




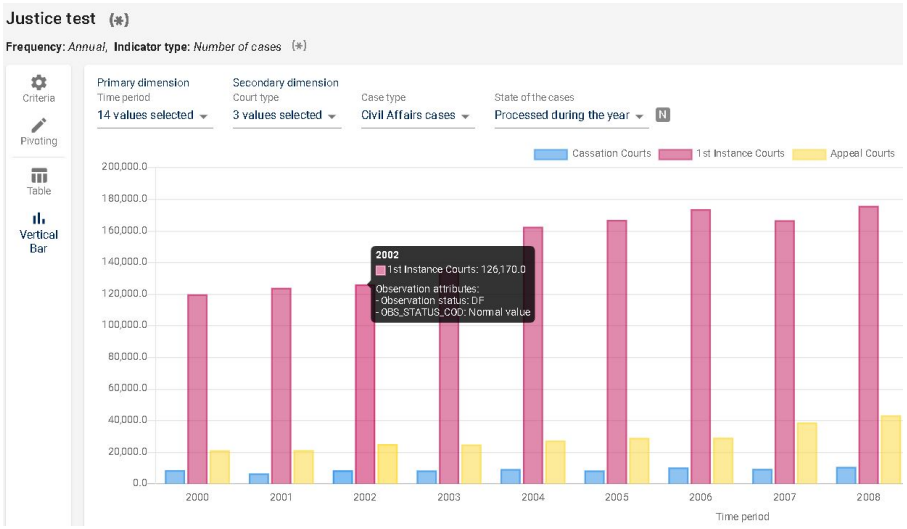
If the dataset has dimension level attributes an asterisk will be present for each value of the dimension, if the the dimension has just one value the attribute is visible in the subtitle.



If data has observation level attributes an asterisk will be present in the corresponding cell.



If the user chooses Chart as display mode, the attributes are also displayed. The attributes at observation level are visible as tooltips by clicking on chart elements, the attributes at dimension or table level follow the same logic described for the tabular view.



5.3.6 Annotations

When talking about annotation, we consider the possibility of setting configurations, at meta and data manager level (this means when creating the dataflow), regarding

- the visualization of single dimension or dimension's items
- items' order
- inclusion of keywords to the dataflow
- row, column and section layout
- criteria selection mode
- territorial dimension ids

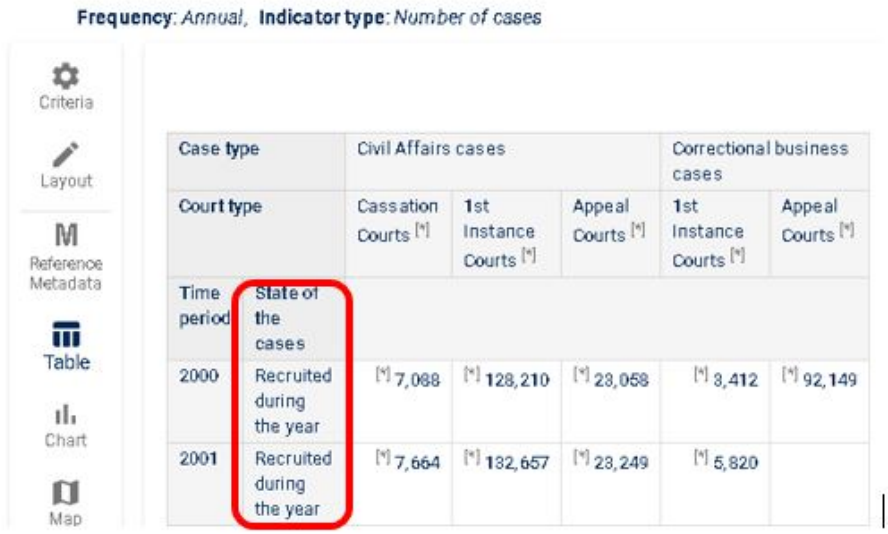
- dataflow update and more.

In general, for the application Data Browser to recognize the annotations, annotations' IDs must be inserted in the node configuration under the ANNOTATION tab. Foreach annotation type, the correspondent ID (which must be exactly the same that appears in the metadata manager application), has to be written in the textbox.
Let's get a closer look to the most used annotations.

Annotation Not Displayed

The Administrator user can decide to not show some elements in the dataset by setting annotation "Not Displayed" in Dataflow's metadata. The user can choose to not display the whole dimension or just some items.

During data visualization, if the annotation "Not Displayed" is at dimension level, the dimension is not added in the results, but just if it has only one element otherwise the annotation is ignored.
If the annotation "Not Displayed" is at item level all the lines with items having this kind of annotation are not displayed.



Annotation Order

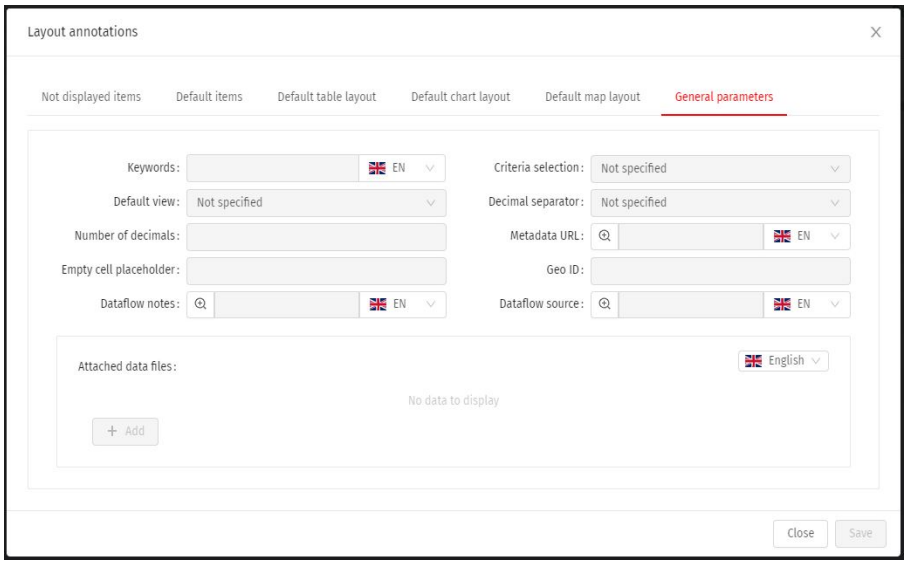
This annotation specifies the order for codelists, dimensions and categories present. Such annotation can show its effects when visualizing the table of the dataflow or the category tree.
Items' sorting is defined in the metadata manager.

Other annotations

- **Default items** : used to initialize criteria with the filters set in this field.
- **Default table layout** : sets how dimensions must appear in table for rows, columns, filters and sections' configuration.
- **Default chart layout** : sets how dimensions must appear in chart (primary dimension, secondary dimension and filter dimensions).
- **Criteria selection** : sets the criteria selection mode for a dataflows (independently from the node configuration).
- **Attached Data File** : sets list of files and formats in which the user can download the dataflow.
- **Decimal separator** : sets the decimal separator (dot or comma)
- **Number of decimals** : sets the number of decimal after the separator.
- **Empty cell placeholder** : sets the value to visualize in case of empty cell.
- **Dataflow notes**: sets notes associated to dataflow
- **Dataflow source**: sets source of dataflow (e.g. Eurostat)
- **Metadata URL**: sets url where metadata are defined.
- **Keywords** : used for dataflow reseach.
- **Default views** : sets default display mode of the dataflow (table, chart or map)
- **GEO ID**: allows to indicate the territorial dimension used in the dataflow in order to show the map visualization.
- **Last update** : if set, shows the information regarding last update of the dataflow.
- **Linked dataflow node** : indicates that a dataflow type is linked and specify the node to which the dataflow is linked
- **Dataflow catalog type** : identify dataflow type (e.g. linked, virtual, normal)

By clicking on the icon **'i'** the user can have further information about the annotation.

All these annotation can be set at Data Structure Definition or Dataflow level in the metadatamanager platform.
The following image shows the configuration of some annotation at dataflow level.

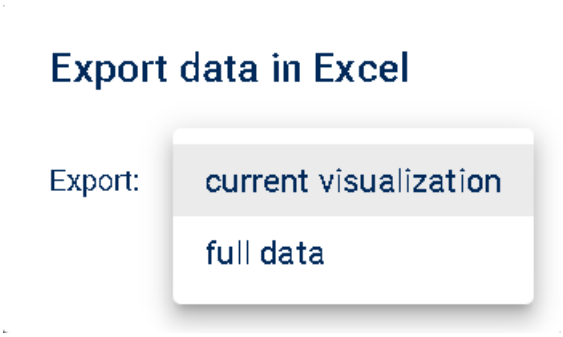


5.3.7 How to download data

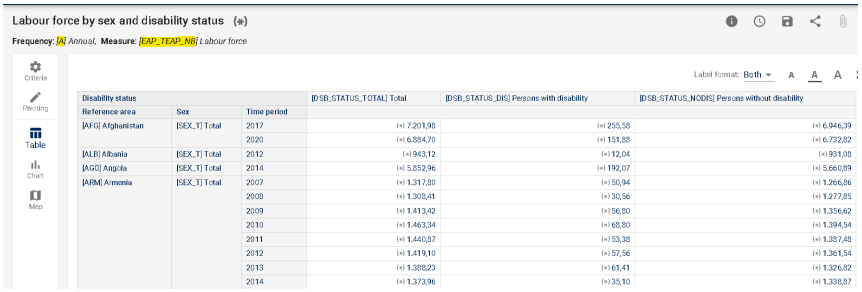
The download of a dataflow in the different formats can be activated through a down arrow icon on the top right corner of the page.

The formats available for each node are defined in the node configuration and will be a subset of the overall formats supported by the application, which are:

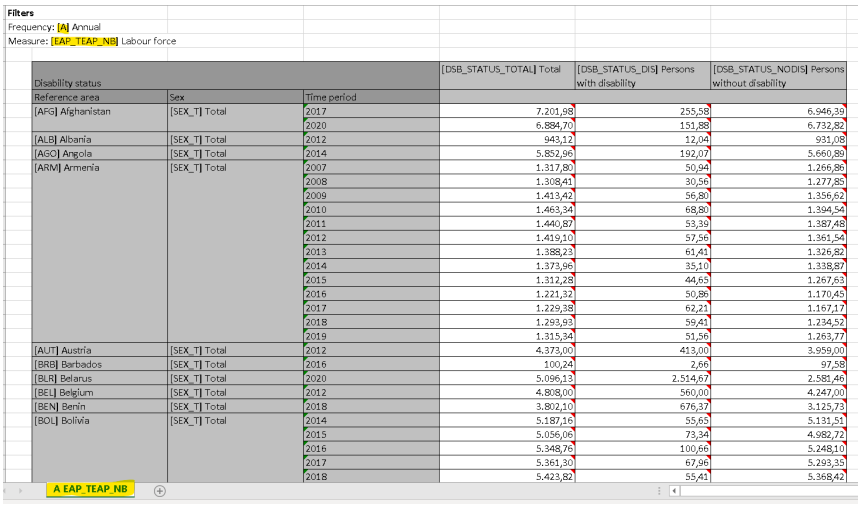
- SDMX Standard
- SDMX - Generic 2.1*
- SDMX - Generic 2.0*
- SDMX - Compact 2.0*
- SDMX - Structure Specific 2.1*
- Custom CSV
- SDMX-CSV*
- SDMX-JSON*
- RDF
- JSON-STAT
- PC-AXIS
- HTML (table only)
- JPEG (only for graph and map)**
- Excel (table only): a popup will allow the user to select the desired export:



if the user chooses “current visualization” he will get in return an excel sheet containing exactly what the table is showing at the moment. On the other hand, by choosing “full data”, the system will create a sheet for each combination of filters.
For example, if we consider Dataflow:



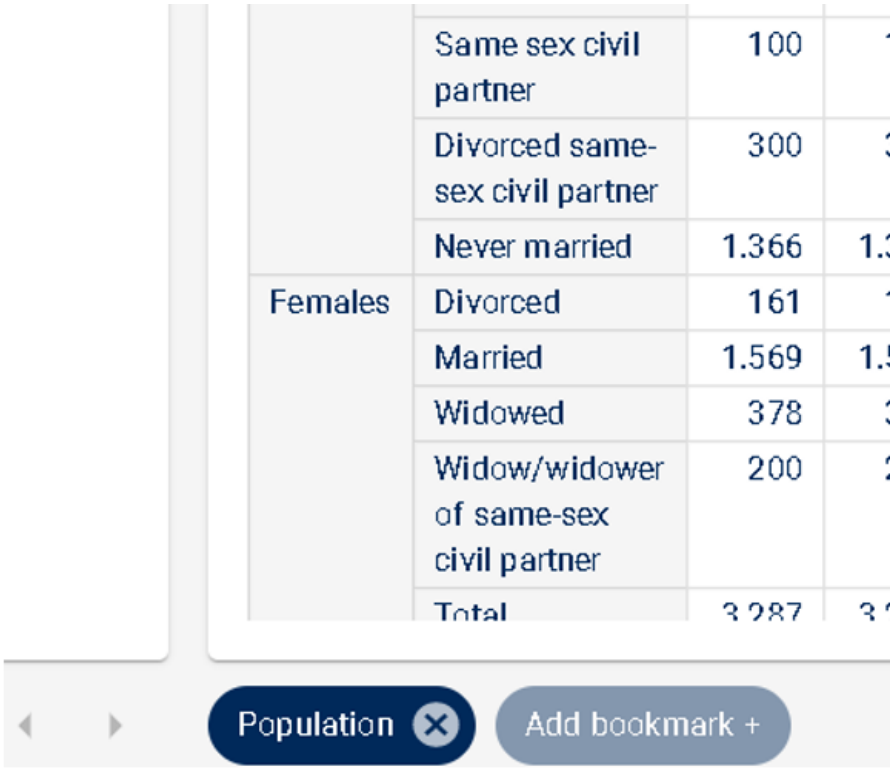
the user can download the following current visualisation:



*only available for REST nodes
**to enable image downloading, the user just needs to insert “image” in the *Download file formats* field in the node configuration’s view tab.

5.3.8 How to add a bookmark

When we have finished visualizing the data for a given dataflow, you can keep track of this navigation by adding a bookmark using the button at the bottom of the visualization itself. If, for example, you are viewing data from the "Population" dataflow, adding a bookmark will take that name.



The bookmark will always be present even when other data are opened, so clicking on it will always take you back to the last view you made on "Population". Once the browser is closed the saved bookmarks are all deleted.

5.4 Views

In this section, we will explain how to manage, save and share views once data is visualized.

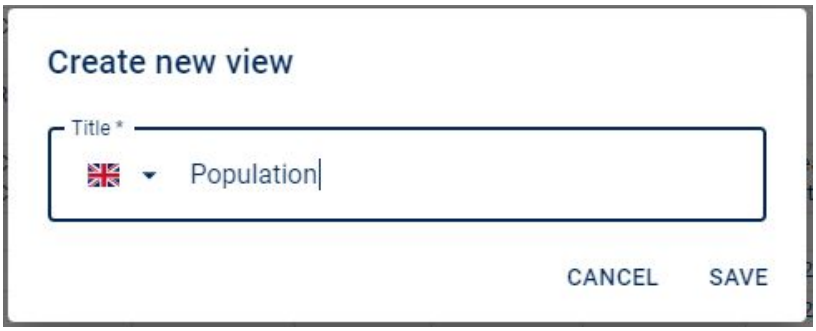
5.4.1 What is a view

Views can be described as visualization at user level that, once saved, they can be accessed again in other moments as long as the user doesn't decide to delete them. The user modifies criteria and/or layout and saves his choices, so that when he reopens the saved view, the output will show the settings he previously configured for the table.

5.4.2 How to save a view

Once the user visualizes his data, as shown in previous paragraphs, it is possible to change the criteria (this enables filters on the output) or the table layout (by changing position of the dimensions in the table). The new table presents a different output from the default. This new visualization can be saved as a View. Multiple views can be saved for the same table. To save a view, the user must click on the save button and select "Save View" from the list that appears:

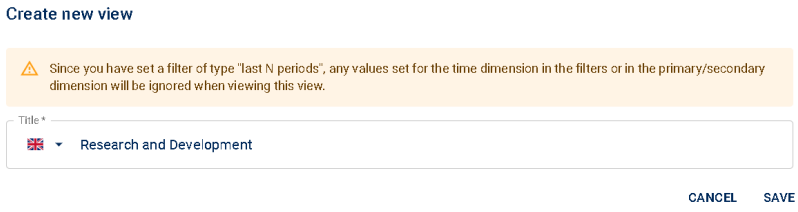
this will bring to a new window that enables the user to set the name to the view. The multilanguage functionality allows users to set different titles depending on the selected language. This is made possible by selecting the flag related to a language and by defining a title for that language.



If a filter of type "last periods" is set in the criteria on the temporal dimension, then the view will always show:

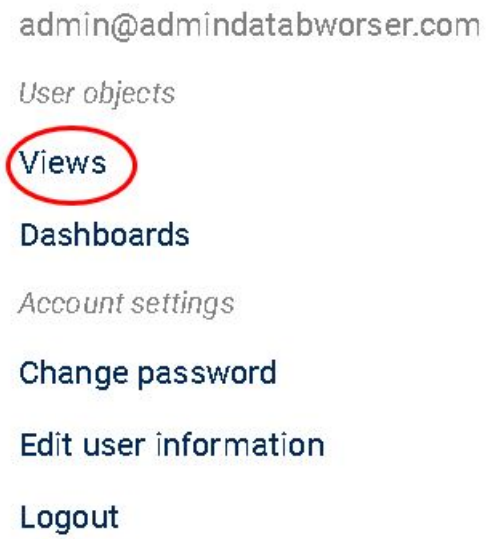
- the last available period in the data, if the temporal dimension is inserted in the filters of the multidimensional table
- all the values present for the data, considering the last N periods, if the time dimension is set as primary or secondary dimension of the graph

even if the current view that the user is saving is different (e.g. a period other than the last one is filtered). The user who creates this view will be warned at save time that the values of the filter or the primary/secondary dimension he has set will be ignored at display time and will be asked to set a "custom range" policy if he wants to preserve those values.



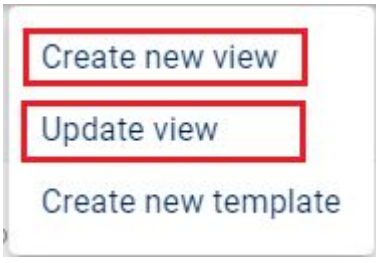
5.4.3 How to manage views

The user can search for the views he saved and delete them, by clicking the user icon, that appears in the main menu bar on the top-right of the pane, and selecting "Views".



The new window shows the information about the saved view (which node is part of, the data ID, name) and also actions the user can perform (visualize the view, delete it).

Once the user visualizes a view, it is also possible to modify it. In this case changes can be used to overwrite the existing view or create a new one. This selection can be made by choosing the preferred option from the menu shown by clicking the save button.



5.5 Linked dataflows and only file dataflows

The application provides the possibility to include within the nodes also linked dataflows.
A linked dataflow is a dataflow defined in a node but linked to another dataflow which is contained in a totally different node.
The advantage of having a linked dataflow consists in the fact that it is not necessary to have all the base structures mandatory for a dataflow to be published on the node we are using but all we need is the link to the orginal node that contains it.
Nevertheless, the most important thing for all the mechanism to work is that the orginal node that contains the dataflow we want to link to, must be included in our hub named with its original ID.
Another feature that we can find in our platform, consists in the opportunity of having only file linked dataflows which are dataflows without data but just with attachments. In this case the annotation *DataflowCatalogType* must be set in the configuration node (as well as in the meta and data manager side) and also the checkbox *Show only file dataflow* must be checked in the view tab. This is an example of how an only file data appears in the Data Browser:

In this case only attachments are downloadable but not the data itself.