EUBIROD

EUropean Best Information through Regional Outcomes in Diabetes

Executive Agency for Health and Consumers

The BIROBox User Guide

November 2010

The “BIROBox User Guide” is published as an Annex to Deliverable “D5.3: Database Engine”, a product of “WP5: Data Collection”, the European project “European Best Information through Regional Outcomes in Diabetes” (EUBIROD), co-funded by DG-SANCO, European Commission, 2008 (G.A. 2007115)

WP Leader: University of Perugia, Italy

Scientific Coordinator: Prof.Massimo Massi Benedetti
Technical Supervisor: Prof.Massimo Massi Benedetti

Compiled by the EUBIROD Consortium Core Development Team:

Fabrizio Carinci, Senior Statistician, Serectrix snc, Italy
Valentina Baglioni, Software Engineer, University of Perugia,Italy
Luca Rossi, Statistician, University of Perugia, Italy
Peter Beck, Joanneum Research, Austria
Phillip Perner, Joanneum Research, Austria
Scott Cunningham, University of Dundee, Scotland
Massimo Brillante, University of Dundee, Scotland
Ritchie Mc Alpine, University of Dundee, Scotland

Secretariat

Anna Rita Ragni, Secretary, EUBIROD Coordinating Centre, University of Perugia, Italy

Citation

The EUBIROD Core Development Team, The BIROBox User Guide**,** EUBIROD Consortium, 2010

Address for correspondence

EUBIROD Coordination Centre
Via E. dal Pozzo 06126 Perugia – ITALY
Ph/Fax. +39 075 5727627
Email: **eubirod@unipg.it**

Project Website http://www.eubirod.eu

Table of Contents

1. Objectives

Why the BIROBox?

This section explains why a BIRO user should use the BIROBox.

1.1 Aim of the BIROBox

The BIROBox is a tool allowing the user:

- to extract clinical data on diabetes from a local data source
- to export local data towards a common European format defined by the BIRO project
- to perform a standardized statistical analysis on imported data, delivering reports based on the BIRO Report Template adopted in EUBIROD
- to access reports produced in .pdf and .html format

1.2 Output indicators

The BIROBox uses a set of statistical routines ("statistical engine") to process data directly provided and managed by the user at each data source. The result is a report including a total of N=72 indicators agreed by the EUBIROD Consortium as a common standard, including the following:

- demographic characteristics
- clinical characteristics
- health system
- population
- intermediate and terminal risk adjusted outcomes

The complete list of indicators is included in Appendix A.

2. Data Requirements

Which data is needed by the BIROBox?

This section describes the datasets required to get results from the BIROBox.

The fundamental prerequisite for using the BIROBox is that the user maintains an updated **diabetes register** of any form.

The diabetes register may be organized in different ways. The possible cases, in increasing order of complexity include: periodic data collections (annual survey), electronic registers of linked administrative data from multiple sources (hospital data, pharmaceutical prescriptions, diabetes clinics), routine collection of clinical data in computerized format at one or multiple diabetes clinics (diabetes management programs).

Under the most advanced conditions, the register is **population-based**, meaning that records from different datasets are linked to a master index including all subjects IDs (either real IDs or pseudonyms) and geographical references that allow assigning each subject to a specific catchment area or region.

Population-based registers usually allow performing the same referencing procedures not only on diabetes patients, but also to the general population. Under such conditions, it is possible to define accurate denominators for all diabetes indicators. This way, the epidemiological analysis can be considered more robust, as it is possible to capture most cases and to control bias by comparing each sample of diabetic patients to the overall population. Moreover, the highly relational shape of population-based databases easily allows to link structural profiles of each data source to the data collected, allowing to explore the organizational characteristics that may impact on diabetes outcomes.

Optimal conditions for the realization of population-based registers are not easily found. For this reason, the system realized in the BIRO project targeted less restrictive situations in which diabetes data are carefully collected, but may not necessarily be able to assign an individual subject to a specific population group or denominator.

Nevertheless, the system embedded in the BIROBox aims to apply the population-based logic to all possible cases.

The design of the system **splits the data requested** to the BIRO user across different tables that can be dynamically used to compute accurate diabetes indicators from an epidemiological perspective. Some registers may be able to submit only specific tables, others – particularly population-based registers – will be able to feed the system with all required information.

The nature of the input data requested by the BIRO system is different from most reporting systems created in diabetes

The approach allows generating quality improvement loops for the production of better diabetes information. By repeatedly applying this logic for the production of local reports, users conform to a common standard and progressively improve their internal to collect complete diabetes information.

The BIROBox requires five different datasets (tables): “Merge”, “Activity”, “Population”, “Diabetic Population”, “Data Source Profile”.

Currently only the **“Merge Table”, the “Population Table” and the “Data Source Profile” are mandatory** for a correct execution of the BIROBox.

In the following sections we describe the contents of each table required.

2.1 Merge Table

The “Merge Table” is a file or database supplied by the user that includes all information collected in a series of clinical encounters (episodes) occurred during a set time frame to a group of individuals.

Currently the population table **is mandatory** to correctly complete a BIROBox import procedure.

The structure of the merge table is the following:

```
{patient_id,episode_date,[data field 1],[data field 2],...,[data field n]}
```

where:

- **patient_id** is a field for the subject ID
- **episode_date** is a field for the date in which the individual data were recorded
- **data field n** is a generic expression indicating any other BIRO parameter included in the “BIRO Common Dataset” (risk factors, measurements, tests, treatments, outcomes) which can be extracted from the local data source to compute target BIRO indicators. All possible [data fields] along with target formats are listed in the “BIRO Common Dataset quick reference” (see Appendix B).

The couple (patient_ID, episode_date) is used as a **primary key** of the BIRO table that will result from the import of the Merge file to the BIRO database.

Currently the BIRO database can only be successfully created by the user if he/she supplies the following **mandatory fields: patient_id, episode_date, date of birth, date of diagnosis, sex, type of diabetes**. Please note that subjects with missing or non valid values (e.g. date of diagnosis after an episode date) are allowed, but the corresponding records will not be imported in the BIRO database. Therefore, files with high frequencies of missing values for the above variables will lead to unreliable results.

The user is encouraged to **complete the mandatory items of the merge table**, if possible. In fact, in some cases approximated values can be assigned to any field when precise data are not available. However, such operation should be performed with great caution. For instance, if the date of diagnosis is not available in a dataset where the user knows that only new diabetes cases in the current year have been included, the date of diagnosis field can be set to an arbitrary value (e.g. 1st Jan or 30th Jun). The same applies to the Type of Diabetes if the data collection was only performed for specific types (Type 1, 2). The above operations ensure that the BIRO analysis can be carried out without substantially altering the quality of the final results.

The user shall be aware that the BIRO system will use local data that in most cases can be regarded as “non anonymous”, even if the patient ID is only a pseudonym. However, the BIROBox presents no additional risk and can be compared to the most secure software installed on an average machine. In fact, **individual data will never leave the local computer or be accessed by any external user. The security and data protection of the BIRO system corresponds to the level of security and data protection adopted by the user managing the system on top of which BIRO is running. Furthermore, the BIROBox is installed on a Linux virtual machine that can be regarded among the most secure environments available.**

The Merge Table includes a mix of demographical, clinical and outcome characteristics that are usually stored in diabetes registers across many tables. The user may need to build the table performing data transformations that require an advanced technical capacity. This operation can be performed separately using **Pentaho** and imported as a Kettle transformation through the BIROBox “Customized Toolbox”.

The BIROBox is able to import the Merge Table from a database (e.g. MS SQL, MySQL, PostgreSQL), .csv file, BIRO XML export file, or Pentaho Kettle transformation.

2.2. Activity Table

The “Activity table” is a file including information about the transition of individual subjects from a state to another in the local data source. In particular, it includes the dates of entry and exit from the population covered by the data source (e.g. inclusion/exclusion in the list of patients in charge of the diabetes clinic), along with the specific reasons for the entry/exit (e.g. birth, death).

Currently the activity table **is not mandatory** but it is highly recommended to control for epidemiological bias in the denominators. In fact, its contents can influence the calculation of indicators, as the statistical engine would consider patients in the denominator only for the time interval during which they were active at the data source.

The activity table has the following structure:

```
{patient_ID, start_date, start_reason, end_date, end_reason}
```

where:

- **patient_id** is a field for the subject ID which must match the one included in the merge table
- **start_date** is a field for the date in which the individual becomes **active**
- **end_date** is a field for the date in which the individual becomes **inactive**
- **start_reason** is a field stating the reason for which a subject has become **active**. Possible values include: birth, diabetes diagnosis, transfer **from** another centre.
- **end_reason** is a field for the reason for which a subject has become **inactive**. Possible values include: death, transfer **to** another centre, lost to follow-up.

The couple (patient ID, start_date) is used as a primary key of the table imported in the BIRO database. Two different records with the same starting date related to the same patient are not allowed. The same patient may appear in more than one record because it is possible for a patient to have one continuous or several disjointed intervals of activity – for reasons e.g. a residency change.

Details about the requested fields can be found in the Common Dataset Quick Reference Guide (see Appendix B).

The BIROBox is able to import the Activity Table table from a database (e.g. MS SQL, MySQL, PostgreSQL), .csv file, BIRO XML export file, or Pentaho Kettle transformation.

2.3 Population Table

The Population Table includes information concerning **the general population in the catchment area of the local data source**. In particular, it includes data on the total number of subjects dead or alive in the target year, stratified by gender and age bands according to the BIRO data definitions. It allows to correctly compute denominators for population-based indicators.

Currently the Population Table **is mandatory** to correctly complete a BIROBox import procedure.

The Population Table has the following structure:

```
{year, ageband, popM, popF, morM, morF}
```

where:

- **year** is a field for the reference year of values included in the table
- **age_band** is a field for the code of the age band used as a stratification factor
- **popM** is a field for the total number of males in the related age band for the target year
- **popF** is a field for the total number of females in the related age band for the target year
- **morM** is a field for the total number of deaths among males in the related age band for the target year
- **morF** is a field for the total number of deaths among males in the related age band for the target year

Details on the requested fields can be found in the Common Dataset Quick Reference Guide (see Appendix B).

For the Population Table, the user can use official governmental statistics or internal sources, but he/she must ensure that the figures are **as close as possible to the general population in the catchment area, i.e. the population from which individual records included in the Merge Table are generated**.

The BIROBox is able to import the Population Table from a database (e.g. MS SQL, MySQL, PostgreSQL), .csv file, BIRO XML export file, or Pentaho Kettle transformation.

2.4 Diabetic Population Table

The Diabetic Population Table provides essential information concerning the overall diabetic population in the catchment area of the local data source. It includes data on the total number of subjects by type of diabetes for the target year, stratified by gender and age bands according to the BIRO data definitions. The table allows to correctly compute denominators for population-based indicators related to diabetes.

Currently the Diabetic Population Table **is not considered mandatory** to correctly complete a BIROBox import procedure.

If not present, the statistical engine will estimate it from the population included in merge table. The resulting indicators can be grossly biased, as they assume that the local source includes all the diabetic population in the region.

The Diabetic Population Table has the following structure:

```
{year, age_band, typedm, diabM, diabF}
```

where:

- **year** is a field for the reference year of values included in the table
- **age_band** is a field for the code of the age band used as a stratification factor
- **typeDM** is a field for the type of diabetes of subjects included in the table
- **diabM** is a field for the total number of males with diabetes in the related age band for the target year
- **diabF** is a field for the total number of females with diabetes in the related age band for the target year

For the Diabetic Population Table, the user can use official governmental statistics or internal sources, but he/she must ensure that the figures are **as close as possible to the diabetic population in the catchment area, i.e. the diabetic population from which individual records included in the Merge Table are generated**.

The BIROBox is able to import the Diabetic Population Table from a database (e.g. MS SQL, MySQL, PostgreSQL), .csv file, BIRO XML export file, or Pentaho Kettle transformation.

2.5 Data Source Profile

The Data Source Profile provides essential information on the organizational structure of the local data source (diabetes clinic, regional registry, etc). The table allows to interpret the BIRO results taking into account the characteristics of the sources involved in the calculation of diabetes indicators.

Currently the Data Source Profile **is mandatory** to correctly complete a BIROBox import procedure.

The must provide information on the following characteristics:

- **Data Source Type**
- **Contact details**
- **Geographical information**
    - Total population in the catchment area
    - Geographical Area
- **Health services organization**
    - Number of hospital beds
    - Number of physicians
    - Number of diabetes specialist consultants
    - Number of doctors
    - Number of specialist diabetes nurses
    - Number of Disease Management Programmes
    - Number of physicians offering DMP's for diabetes

The Data Source Profile can either be supplied manually by the user, typing values in the fields included in a specific form of the BIROBox, or can be automatically loaded as an XML file extracted from the Online Data Source Questionnaire, made available by the EUBIROD Consortium as a web platform (see section 2.5).

3. BIROBox Software

Which software is needed to run the BIROBox?

This section provides an overview of the software needed to run the BIROBox.

3.1 Software components

All the software components required to run the BIROBox are included in the distribution provided by the EUBIROD Consortium, broadly defined as "The BIRO System", a suite of tools conceived and realized for the first time by the BIRO Consortium (www.biro-project.eu). The BIROBox is a comprehensive software "container" integrating all components included in the BIRO system.

Briefly, the BIRO system includes the following components:

- the BIROBox, a Graphical User Interface written in the Java programming language
- the Database Engine, written in Java and using PostgreSQL as a database management system to store local data in the BIRO format
- the Customized Toolbox, using Pentaho - a suite of open source applications written in Java - to perform BIRO data transformations
- the Statistical Engine, using the R language to perform statistical and epidemiological analysis, and producing BIRO reports through Latex, a document markup language and document preparation system for the TeX typesetting program that delivers high level typographical outputs.

All BIRO software is bundled in BIROX, a "virtual operating system" realized using Linux Ubuntu. This feature allows running all software on different operating systems (including Windows, Linux, MacOS) using the same identical interface and avoiding any interference with the local operating system. In other words, BIRO can be run regardless of any particular configuration of the user machine.

The BIROX distribution is packaged into a single file that must be run using the "Sun Virtual Box", a specific software produced to enable virtualization.

3.2 Software license

All BIRO components are available in the public domain at no charge. However, different licenses can be associated to each individual software used by the BIRO system and developed by third parties. For details, please refer to the main websites of all non-BIRO software components specified above.

All software developed by the BIRO/EUBIROD Consortia is released under the GPL.

3.3 Disclaimer

The BIRO software has been extensively and successfully tested by many users on different operating systems. However, improper usage of the software or unreported bugs are always possible and cannot be controlled by the development team. Please note that usage of the software is at your own risk. The BIRO/EUBIROD Consortia will not be liable for any direct, indirect or consequential loss arising or damages resulting from usage of the software.

4. Software Download

How do I get the BIROBox?

This section provides instructions on how to download a full version of the BIROBox.

4.1 Downloading the "Sun VirtualBox"

The BIROBox has been packed into a virtual machine software distribution named "BIROX" which can only be run within the application "Oracle VirtualBox" (**Figure 1**). To operate the BIROBox, users must have the VirtualBox properly installed, up and running.

Figure 1: Oracle VirtualBox

1. To download the most updated version of the "Sun VirtualBox" direct your web browser to the following link: http://www.virtualbox.org/wiki/Downloads
2. Select the option "Personal use and evaluation license (PUEL)" or choose the "Open Source Version" available at the bottom of the page (equivalent to the former but not providing USB support).
3. Choose the most recent version available for your operating system (Windows, Linux, …)
4. Save the install file on your computer

4.2 Downloading the BIROX distribution

1. Direct your web browser to the EUBIROD website, at the following page providing links to the software components: http://www.eubirod.eu/academy/software/software.html. The webpage will request username/password to access the restricted area. The credentials can be obtained by email from the EUBIROD Coordination Centre (eubirod@unipg.it).
2. Download the zip file of the VirtualBox image.
3. Create a directory on a location of your choice (e.g.: d:\VirtualBox\BIROX) ensuring the availability of an adequate amount of free disk space (usually >2 GB)
4. Extract the downloaded zip folder to the chosen directory.

5. Setup

How do I install the BIROBox?

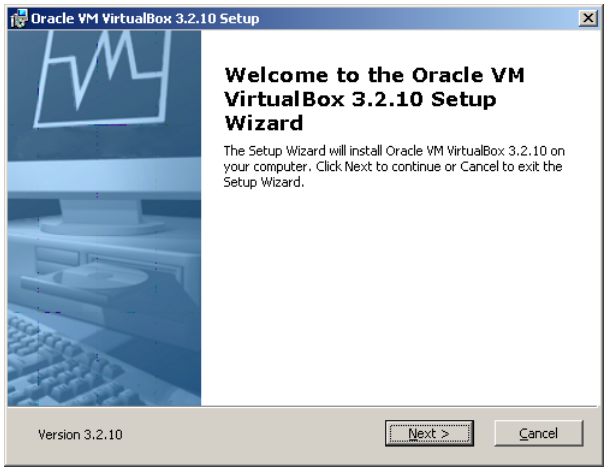
This section provides detailed instructions on how to install the BIROBox software.

5.1 Installation of the VirtualBox

Execute the installation file with Administrator privileges.

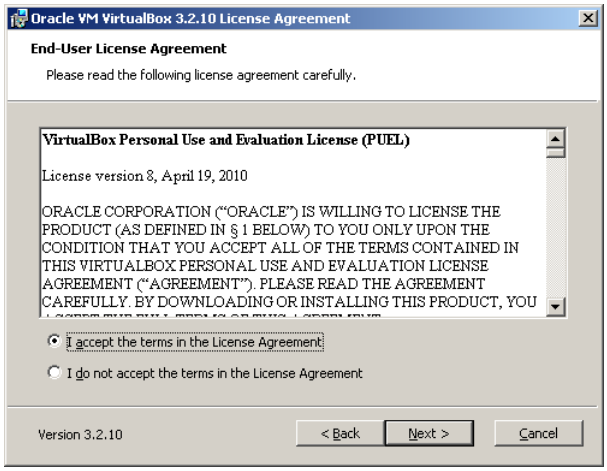
The following steps illustrate all steps on a machine equipped with Microsoft Windows Operating Systems. Other systems include very similar steps for which the user can proceed using the same guidelines.

1) Welcome



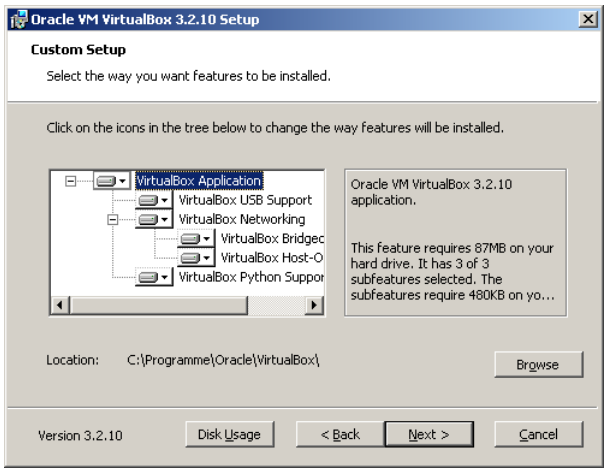
Click **"Next"**

2) License Agreement



Click **"Next"**

3) Custom Setup

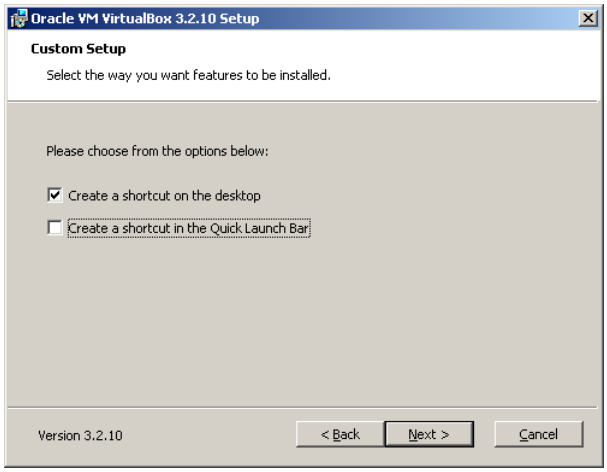


Install all packages (preset)

Accept the location without any changes

Click **“Next”**

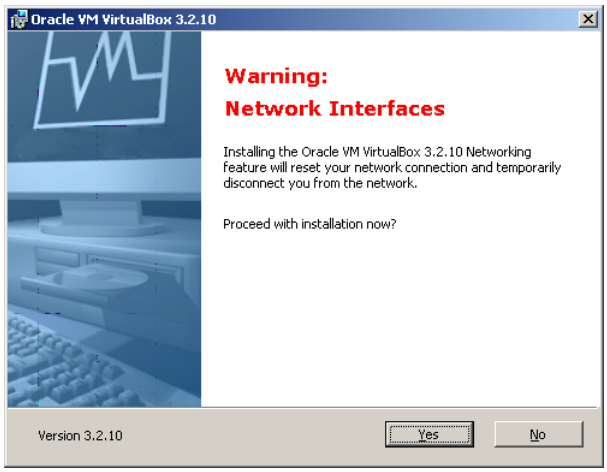
4) Shortcuts



The user can disable the Shortcut for the Quick Launch Bar.

Click **“Next”**

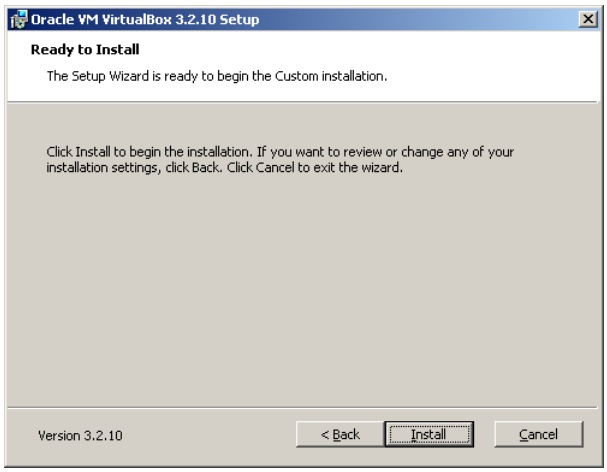
5) Warning (no changes required)



As a precaution, the user should shut down all applications using the network (Skype, downloads, etc) before going on.

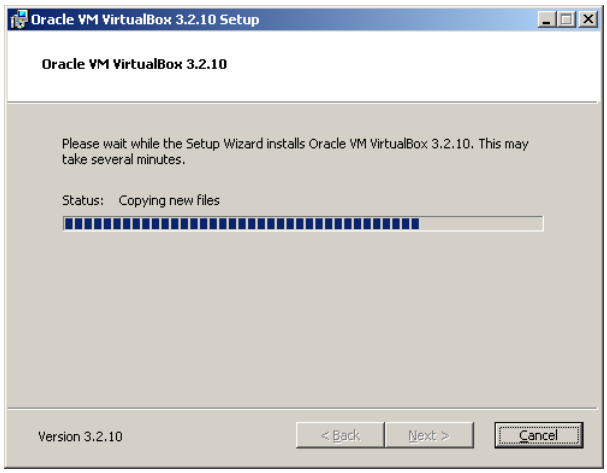
Click **"Next"**

6) Ready to install



Start installation by clicking on **"Install"**

7) Installation



(Wait)

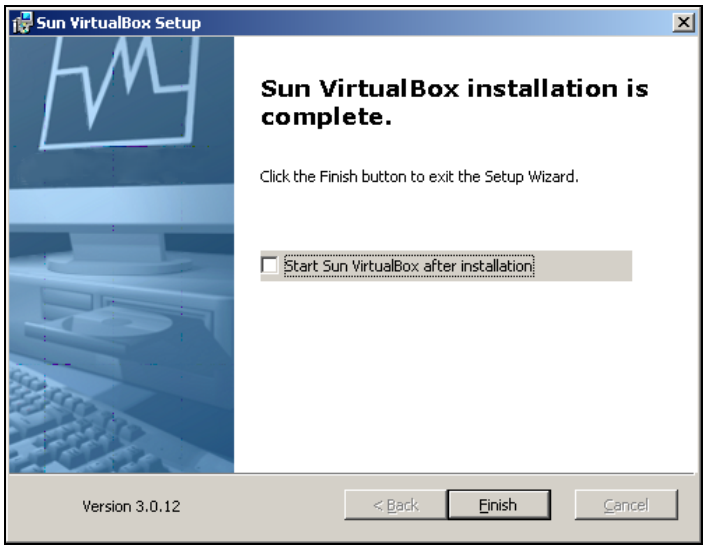
8) Windows Logo Test

At this stage the user can be informed that the driver did not pass the Windows Logo Test, and asked if he/she wants to continue.

Choose: **"Continue installation".**

This window can pop up more than 10 times.

9) Installation complete



Click **Finish**

If you started the application with "Run as…" as Administrator, **Quit and Uncheck "Start Sun VirtualBox after installation"**.

If you are asked if you want to reboot your computer answer with "**Yes"**.

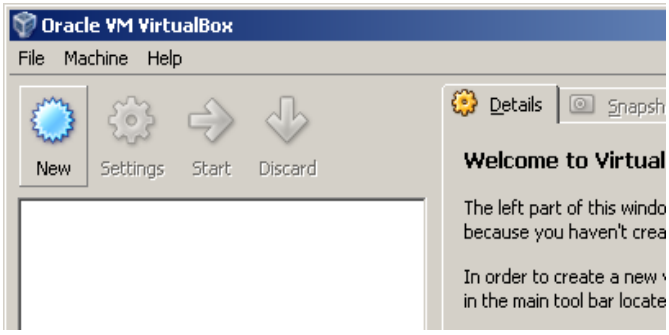
5.2 Creation of a new Virtual Machine

This step is complete to successfully create and configure a new Virtual Machine hosting the BIROBox.

1) Start VirtualBox

Start the Sun VirtualBox by **Clicking** on the Entry in the Start Menu or on the Link Icon on your desktop

2) Create a New Virtual Machine

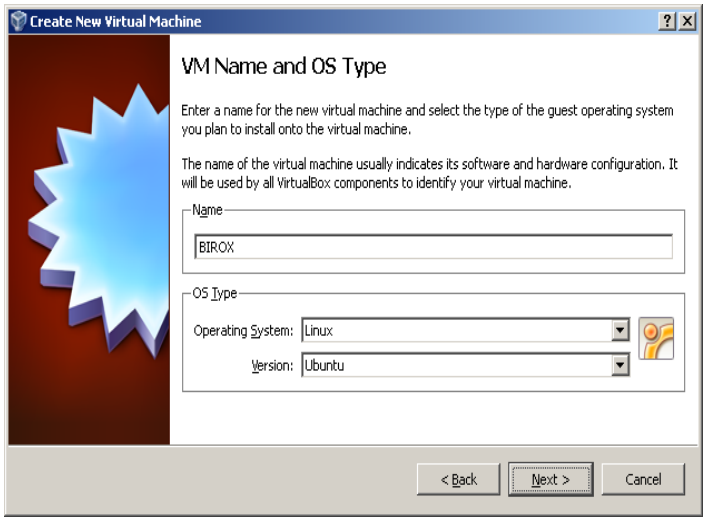


Click **"New"**

3) New Virtual Machine Wizard

Click **"Next"**

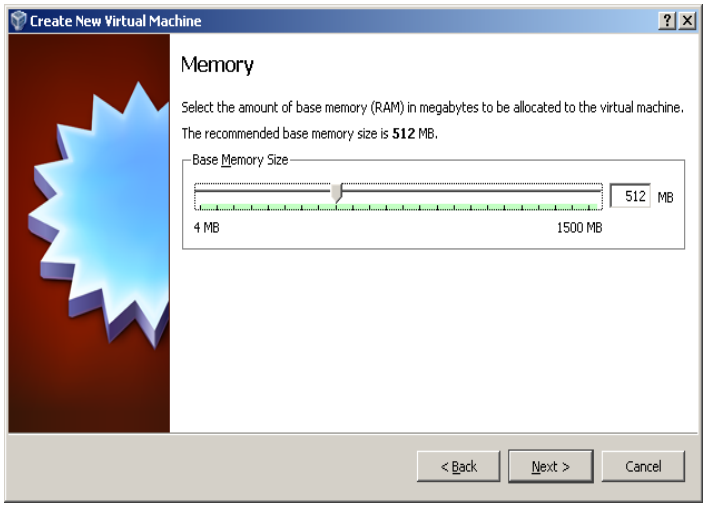
4) Choose name and OS



Enter Name: "BIROX"

Select "Linux" and "Ubuntu"

5) Set Base Memory Size



If your computer has enough memory, you can give more memory to the virtual machine.

A value of **1024Mb** would be optimal under most conditions

You should give as much memory to your virtual machine as possible so that your host operation system is still able to run.

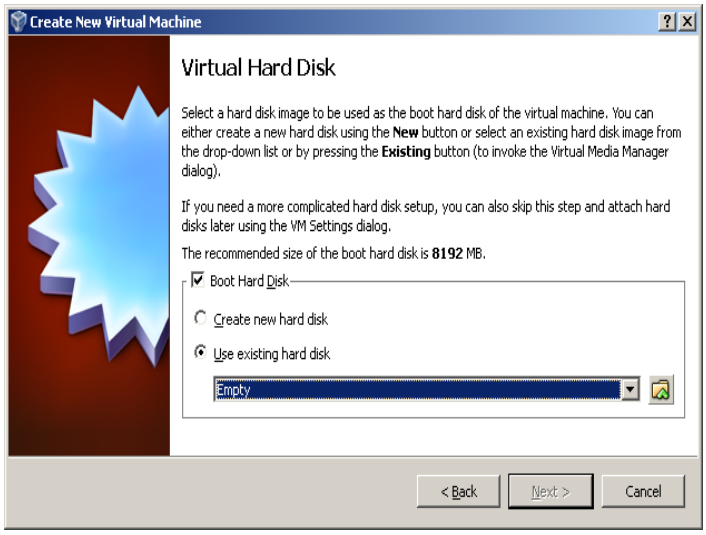
For Windows-operated machines:

- Windows Vista needs 1GB
- Windows XP needs 756 MB

Examples:

- If your computer has 2 GB memory and you are running Windows Vista you can give 1024Mb memory to the virtual machine.
- If your computer has 1 GB memory and you are running Windows XP you can give 512 MB to the virtual machine. (That's a minimum requirement for both)
- If your computer has 1 GB memory and you are running Windows Vista, try 512 MB to the virtual machine....and see what happens!

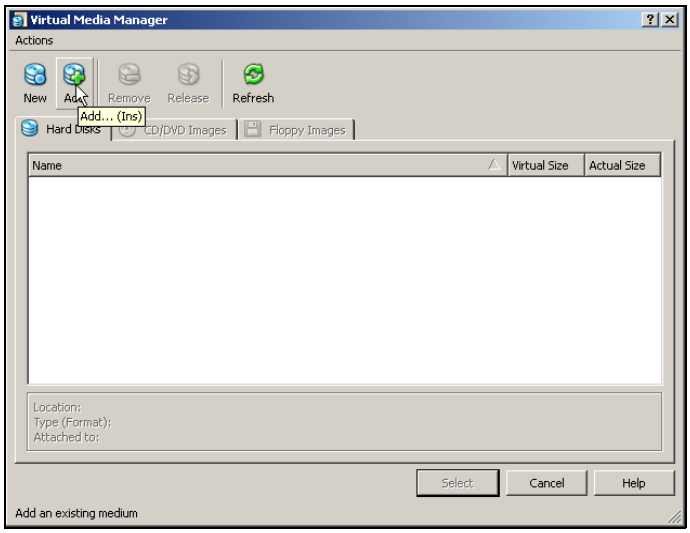
6) Hard Disk



Select **"Use existing hard disk"**

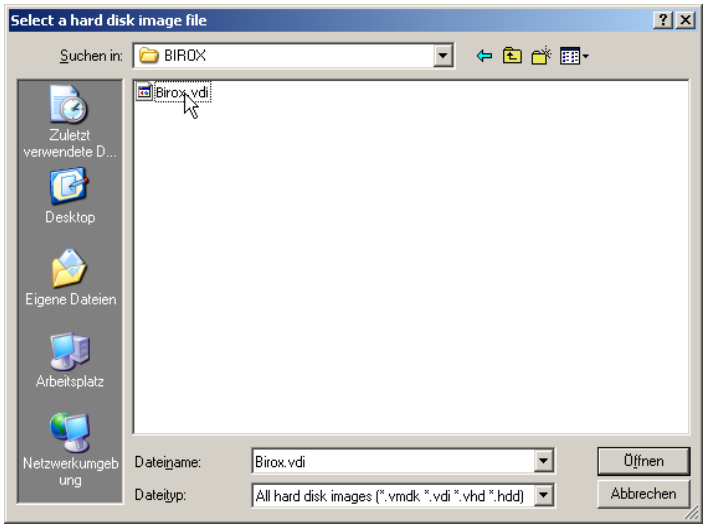
Click on the folder icon

7) Virtual Media Manager



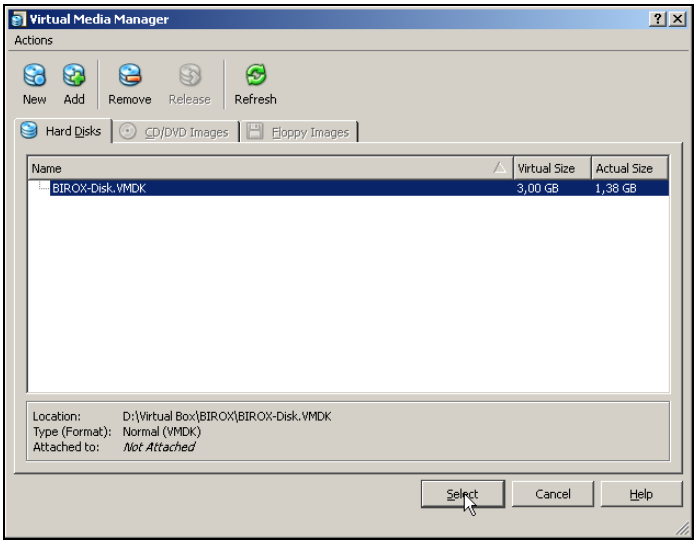
Click **"Add"**

8) Select BIRO Disk file



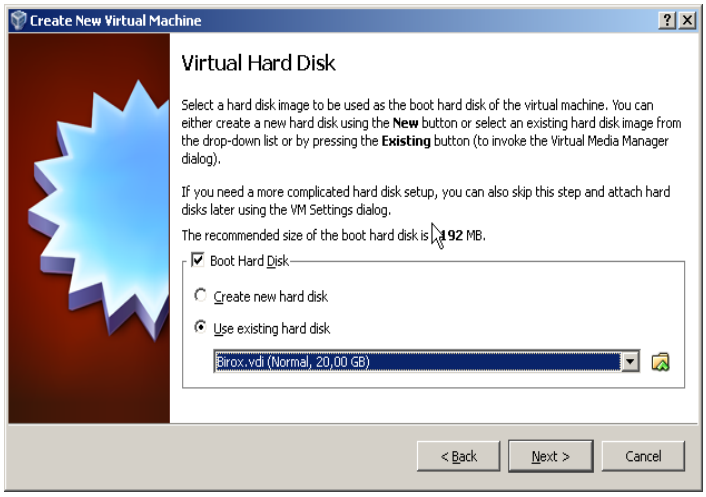
Select the **BIROX*.vdi file** extracted before

9) Virtual Media Selection



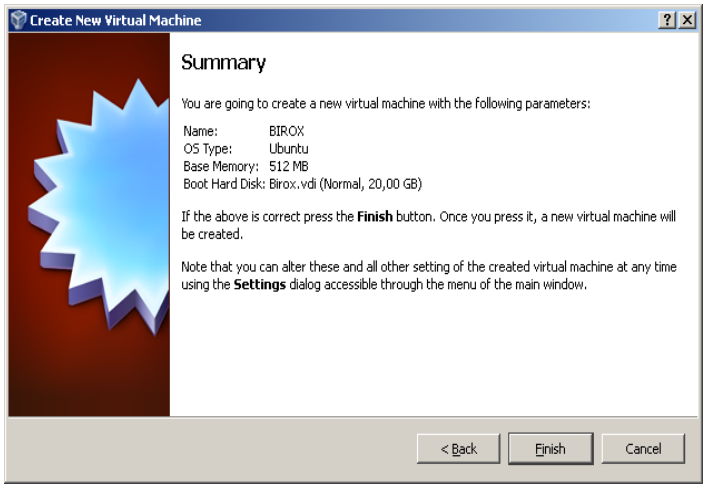
Click **"Select"**

10) Hard Disk selected



Click **"Next"**

11) Summary



Click **"Finish"**

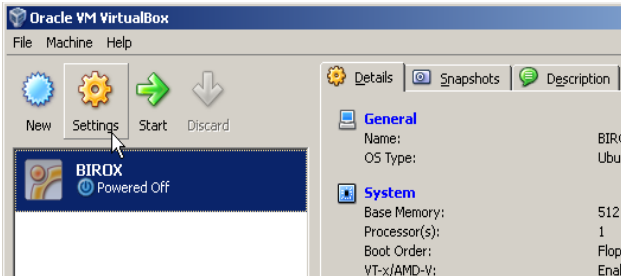
5.3 Configuring the shared folder

The shared folder is a special directory used to exchange data between the Virtual Machine and the host pc. The user definitely needs it to read/copy/move files – particularly local data – from/into the Virtual BIROX.

Saving all BIROBox data inside the shared folder allows the user to reinstall/update the Virtual Machine without loosing the work previously done.

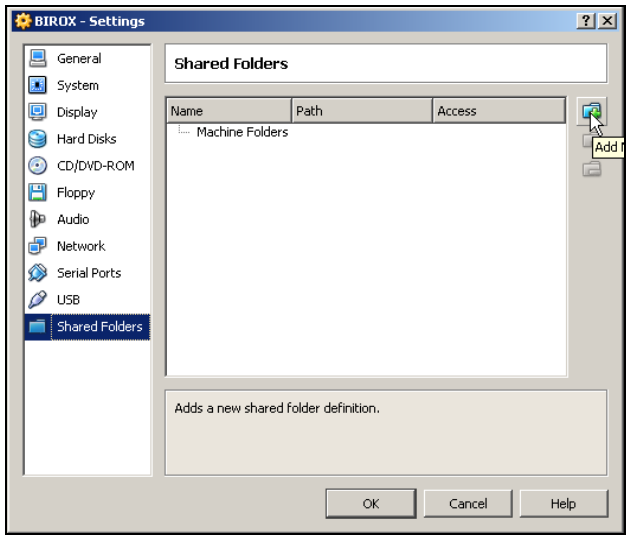
The following steps explain how to configure and explore the shared folder.

1) Edit settings



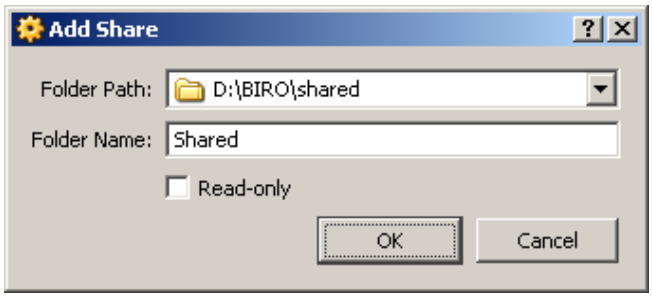
Click the **"Settings"** button

2) Shared Folders



Select "**Shared Folders**" and click **"Add"** on the right

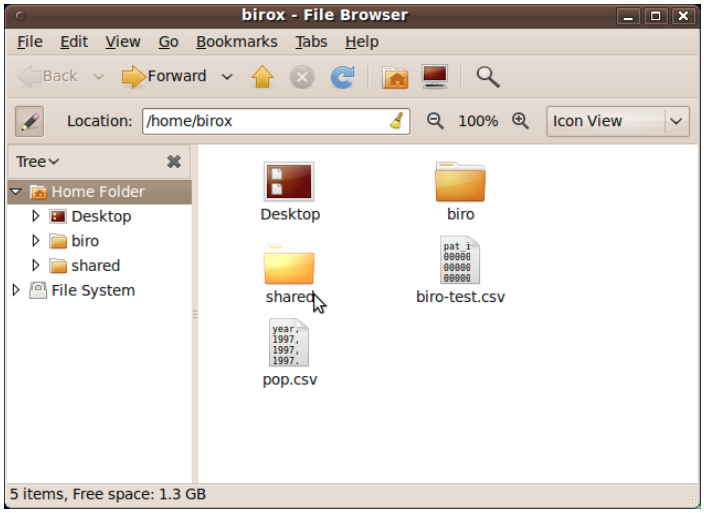
3) Edit Share



Folder Path: Set this to a directory where BIRO settings and data can reside, e.g. D:\BIRO\Shared\
Folder Name: **Must be set to "Shared".**

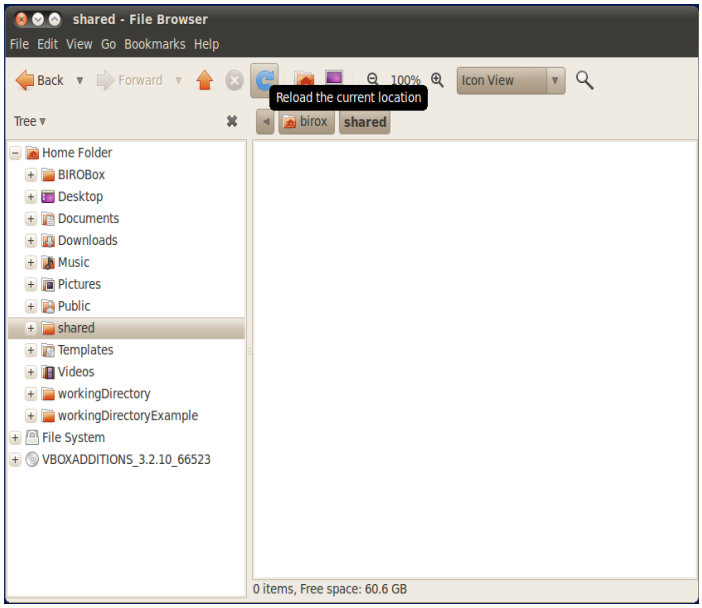
Click **"OK"**

4) Explore the shared folder



Opening “Places”=>“Home Folder”=> “shared” gives you the shared folder BIROX.

All files copied into the shared folder will be visible in BIROX. So, if you have data files stored on your PC, copy or move them to the exchange directory, and you can process them in BIROX (open them with the BIRO Box etc.).

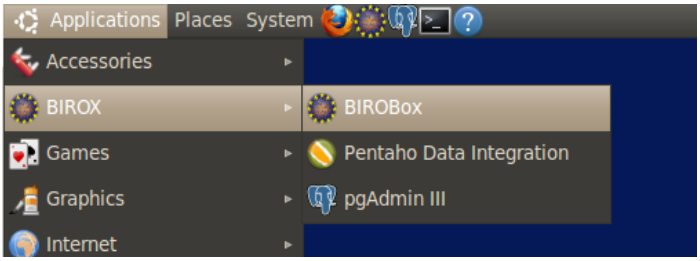


5.4 Running the Virtual Machine and the BIROBox

Once the Virtual Machine has been configured, the following steps are required to run the BIROBox.

1) Start the Virtual Machine	Click **"Start"** Now a "Computer inside your Computer" is booting. If everything works OK you will be prompted with a login.
2) Login	First **select your preferred keyboard layout** in the bottom bar. Then **login** using: **User: birox** **Password: birox**
3) Start the BIROBox (1)	**Start** the Biro Box using the link on the desktop

Start the BIROBox (2)



You can **alternatively** use the Applications Pull-Down Menu to start the BIRO Box or other applications.

6. Using the BIROBox

How does the BIROBox work?

This section shows the use of the BIROBox in its various components

6.1 Startup

At startup the BIROBox requests to specify a working directory (**Figure 2**). The working directory is the default folder where the BIROBox:

- stores all outputs
- searches for all input files
- saves all configuration files

The working directory is organized into a hierarchy of sub folders: a configuration folder plus a folder for each specific work package (database engine, statistical engine, communication software, etc.).

The definition of a working directory allows you to arrange all local files and settings, keeping them well separate from the source code of the software, so that the whole BIROX distribution can be safely updated - if needed - without loosing the work done.

Having a separate working directory allows you to easily backup, freeze or share a system configuration whenever convenient.

6.2 Working directory configuration panel

Figure 2: Dialog to configure the BIROBOx Working Directory

The start-up panel allows you to select an existing Working Directory or a folder that will be used as a Working Directory.

The panel simply displays a text field containing the last working directory you've chosen.

You may use the “Browse” button to search for a suitable folder in your file system to be used as working directory.

It is strongly recommended that you choose a directory within the Virtual Machine shared folder (home/birox/shared) as a Working Directory. Since the shared folder can also be accessed from your host pc, this choice would allow you to:

- easily copy input datasets and files from the host pc into the working directory
- easily copy the outputs from the working directory into your host pc
- protect your work from any updates made to the BIROBox software engine

You can create **different working directories** and decide to use the most convenient one for the current run, using this panel to switch between them.

Please consider that when you switch from a working directory to another one the BIROBox assumes that all the correct settings have been saved into the one indicated in the panel.

If no configuration files can be found, then the BIROBox creates a new fresh, empty configuration.

6.3 Main Window

The main window of the BIROBox displays a button panel on the **left side**, allowing you to access all the main functions of the system:

- Setup: this panel shows the current working directory; allows the user to set the credentials required to access the underlying PostgreSQL BIRO Database; allows the user to specify the credentials of the external input database to be used as source of data.
- Database Engine: this section includes all the functions linked to the input data management: import, quality check, export and inspection.
- Statistical Engine: in this section you may configure and trigger the BIRO statistical analysis and to browse all results obtained.

The **right side** always displays the screen currently active.

6.4 Setup Panel

Figure 3: BIROBox Setup Panel

The **Setup Panel** allows the user to configure the connection to the underlying PostgreSQL Database and the source database where the input data is stored.

The panel includes three sections (**Figure 3**):

1. Current Working Directory: it contains a text field highlighting the working directory previously chosen
2. Output BIRO Database: includes a form to be filled with the info about the PostgreSQL database used to store imported data (host and port, username, password)
3. Input Source Database: if the input data comes from a database, the user must use this form to indicate all connection details (DBMS Driver, database host and port, username and password)

The BIRO **Output Database** section is filled by default with the PostgreSQL configuration packaged with the BIROX distribution. Unless you use a custom BIROBox install, you don't need to change anything in this section.

When clicking the “Check Connection” button, the BIROBox tries to establish a connection to the database using the provided settings in order to verify their correctness.

The **Input Source Database** section needs to be filled only if the data used as input by the

BIROBox are stored into a database

In this case, the user shall select the input source database from a sliding menu, indicating the preferred DBMS Driver from a predefined list of embedded drivers, including:

- PostgreSQL
- MySQL
- MS SQL 2000-2005
- MS SQL 2008
- Oracle 10g
- SyBase

If your local DBMS Driver is not included among those listed, you may add a new custom Driver by clicking the **"+"** button. A dialog pops up asking the user to specify:

- the DBMS name (e.g. "PostgreSQL")
- the JDBC driver class name (e.g. "org.postgresql.Driver")
- the URL pattern with flags for host and port, database name, database username (optional), database password (optional) instead of real values (e.g. "jdbc:postgresql://<hostAndPort>/<databaseName>")
- the absolute path of the jar file containing the JDBC driver for the local DBMS (e.g. "C:\myFolder\postgresql-8.2-504.jdbc3.jar")

Once created, the custom driver will be added to the list of available drivers.

The creation of a new driver requires some IT proficiency, so it is highly advised to perform this step either if you are an expert or with the assistance of an IT expert.

By clicking the **"-"** button you may delete the DBMS Driver currently included.

The **screen configuration can be saved** by clicking on the **"Save"** button.

6.5 Database Engine Panels

The Database Engine Panels allow the user:

- to define one or more data import configurations, specifying the location and format (database, .csv, .xml) of the local data and the differences with the standard BIRO format
- to import local data into the BIRO database
- to inspect the differences between the resulting BIRO database and the original data, through a detailed data quality report
- to export data from the BIRO database to the standard BIRO XML format.

6.5.1. Configuration Manager Panel

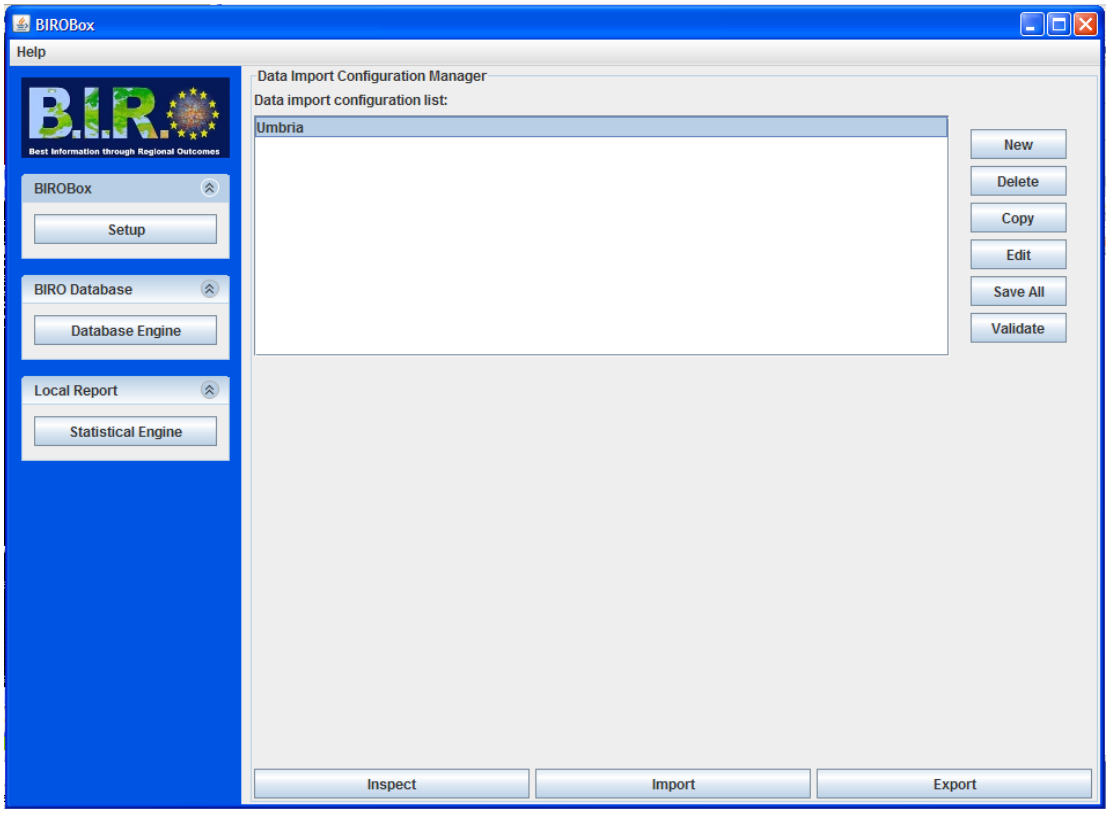


Figure 4: Configuration Manager Panel

The Configuration Manager Panel lists all **data import configurations** saved by the user under the current Working Directory chosen at startup.

The panel allows performing the following actions (**Figure 4**):

- creating new configurations
- managing those already created
- launching the three main BIROBox data import procedures: data inspector, data importer and data exporter.

The white panel lists all the configurations already created. The configuration currently selected is always highlighted in blue.

At the right of this list panel, **small buttons** allow carrying out the following actions:

- **new:** creates a new empty configuration and adds it to the list
- **delete:** deletes the configuration currently highlighted
- **copy:** makes an exact copy of the configuration currently highlighted, adding it to the list
- **edit:** allows the user to edit the configuration currently highlighted through a wizard
- **save all:** saves the complete configuration list into a file
- **validate:** checks the correctness of the currently selected configuration.

Buttons at the bottom of the panel allow the user to open the data inspector panel and to import/export data to different formats, based on the configuration selected in the setup.

The configuration list panel is empty the first time that the database engine section is opened. The only active configuration function is “New”. The inspect, import and export buttons are disabled.

To start configuring the BIROBox, it is necessary to create a new configuration by **selecting the “new”** button. Each new configuration is sequentially labelled as “Configuration <n>”, unless the user selects a custom name for each configuration.

The content of each new configuration is null by default. The user must edit all contents through the most appropriate “**edit**” button. Please refer to the “edit section” for all details of this operation.

To test the BIROBox with different data sources or alternative options, it is possible to create an identical copy of a specific configuration through the “copy” button. This way, it is possible to build a new configuration from an existing one, modifying its contents at best convenience.

The user can simultaneously save all configurations by clicking on the “save all”. The contents of all configurations will be freezed into a special file named “BIROAdaptorConfigurationList.dat”, located in the “conf” folder under the current Working Directory. This advanced feature explains why it is fundamental to backup a Working Directory – better into the shared directory residing on the real machine – to restore its contents at any time.

The **validate** button checks for the correctness of the currently selected configuration. The validator verifies that the BIROBox can connect successfully to the local data sources and that at least all the mandatory fields can be extracted from them.

Data indicated in a specific configuration can be imported only if such configuration is successfully validated. The step must be performed each time the configuration screen is accessed. The import button is enabled only if this operation is successfully concluded within the current session.

The output of the validation process is logged into a pop-up window helping the user to locate and correct mistakes. Although the validator can identify most of the common configuration mistakes, it cannot be exhaustive.

The **validate button** does not relate to the accuracy of local data but is only related to the correct **configuration** of the options required to import the local data into the BIRO database. Furthermore, it does not perform any test on the correct mapping between the local data and the BIRO standard. These operations are left to the user and can only be ascertained through a careful evaluation of the results of the import process, which can be facilitated by the inspection routines and the direct examination of the data quality log file.

After the validation of a specific configuration, the user may import data in the BIRO database by clicking on the **“import”** button. A log window will display all actions performed, including any warnings arising from the import process. A progress bar indicates the state of advancement of the procedure. The import and quality check are carried out in sequence.

The import process may require a long time to be completed, depending on the local data source size and quality.

The data quality check includes the following steps:

1. *Data format check*: the system detects missing values, values with wrong format and values out of range
2. *Coherence check (merge table)*: the system compares values in the same record. It detects those groups of values which are well formatted and within ranges but are not reciprocally consistent (e.g. date of birth following the date of diagnosis)

3. *Duplicate check (merge table)*: in the merge table two records can be defined as "duplicates" when they have the same patient ID and episode date. Duplicates are merged together in order to retain as much information as possible. If the system finds two "overlapping values" (non null values in the same BIRO field) it flags an error on the record and discards one of the values.
4. *Cleaning*: the system discards all records with non admissible values in at least one of the mandatory fields.

After the data quality check, the BIROBOx creates the Data Quality Log File including all statistics and errors resulting from the import of input datasets. At the end of the process, if anything failed during the import, the BIROBox may warn the user about the need to revise the specific configuration.

Once the import is complete, the user may click on the **"inspect"** button to explore the imported data and the data quality report.

The data **export** function will create a zip archive containing a set of XML files extracted from the import dataset (after the data quality check).

The exporter creates the following files:

- DataSource.xml: the XML representation of the data source header, profile and field export profile
- PopulationTable.xml: the XML representation of the population table
- DiabeticPopulation.xml: the XML representation of the diabetic population table
- a set of XML files for the activity data, profile and episodes relative to N patients

The above archives may be used as direct input for the BIROBox.

6.5.2. Configuration Editor Panel

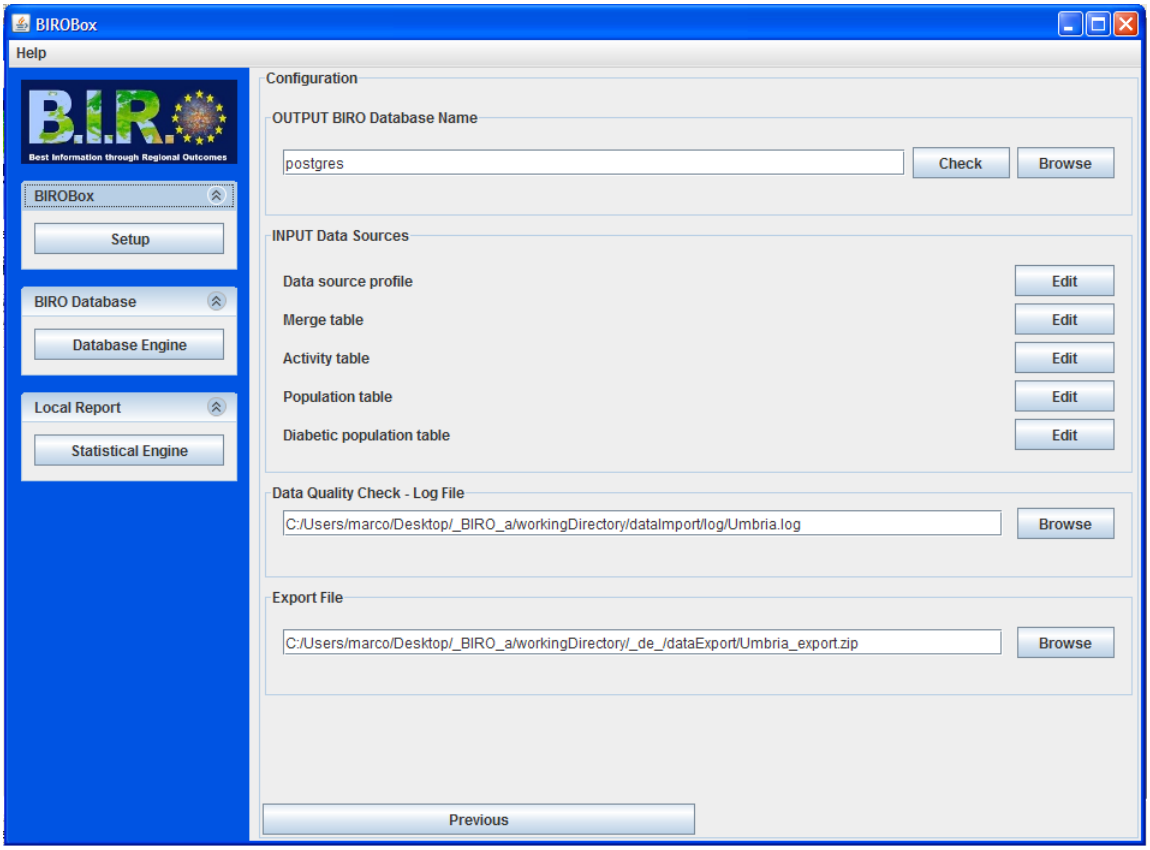


Figure 5: Configuration Editor Panel

The **Configuration Editor Panel** allows to specify the data supplied by the user at the data source level.

The configuration editor includes four major sections (**Figure 5**):

- Name of the Output BIRO Database: this section allows choosing a name for the BIRO Database in which the local data will be imported
- Input Data Sources: this section allows to open other screens allowing the user to configure the Data Source Profile and the Merge Table, Activity Table, Population Table and Diabetic Population Table.
- Data Quality Check Log File: this section allows defining the path and name of the data quality check log file
- Export File: this section allows you to define the path and name of the ZIP archive where the data will be exported

On the bottom of the panel a navigation button allows the user to leave the data configuration screen and open the configuration list panel.

The Output BIRO Database Name section allows the user to choose a name of the database where the data will be stored. The browse button helps choosing the name from a list of databases already present. The check button verifies if the desired database exists, otherwise it creates it.

If the name of the database corresponds to an already existing Postgres database, all data previously contained in the database will be lost during the import process.

Clicking one of the **"edit"** buttons will display a specific wizard for the Data Source Profile, Merge Table, Activity Table, Population Table and Diabetic Population Table.

The section on the **location of the quality log file and export archive** include default values - "<WorkingDirectory>/_de_/dataImport/log" and "<WorkingDirectory>/_de_/dataExport" respectively.

The user may change them either by typing them directly or using the **"browse"** button to browse the file system for more suitable location/names.

6.5.3. Data Source Configuration panel

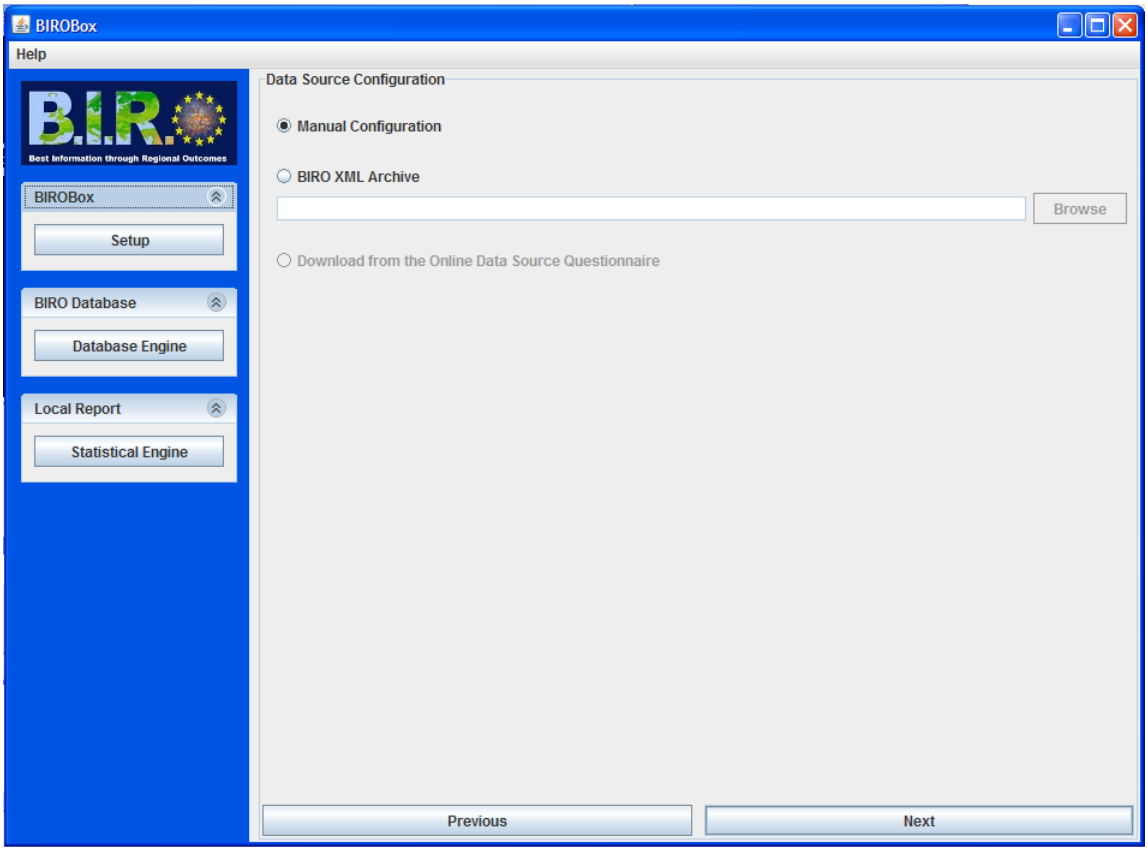


Figure 6: Data Source Configuration Panel

The **Data Source Configuration Panel** allows the user to choose if the data source profile will be supplied to the BIROBox through an external BIRO XML Archive or directly by typing the values of all parameters in a specific form (**Figure 6**).

The panel includes a simple set of two radio buttons providing the user with two alternative options. He/she may decide to import the site information directly from a previously exported BIRO XML archive (if available), by selecting the appropriate option and browsing for the correct file path.

To this end, the XML summary file extracted from the online Data Source Questionnaire (available at http://questionnaire.eubirod.eu) can be profitably used as a site profile source within the BIROBox. The questionnaire has been specifically designed to capture the basic 'metadata' required to accurately describe local differences in data collection, data standards and data quality across all partners. Such information allows documenting the content of all EUBIROD data sources and/or any significant deviations from the BIRO standard definitions.

Otherwise, the user may leave the default option and proceed with the manual configuration, by clicking on the **"Next"** button in the navigation bar.

6.5.4. Data Source Profile Panel

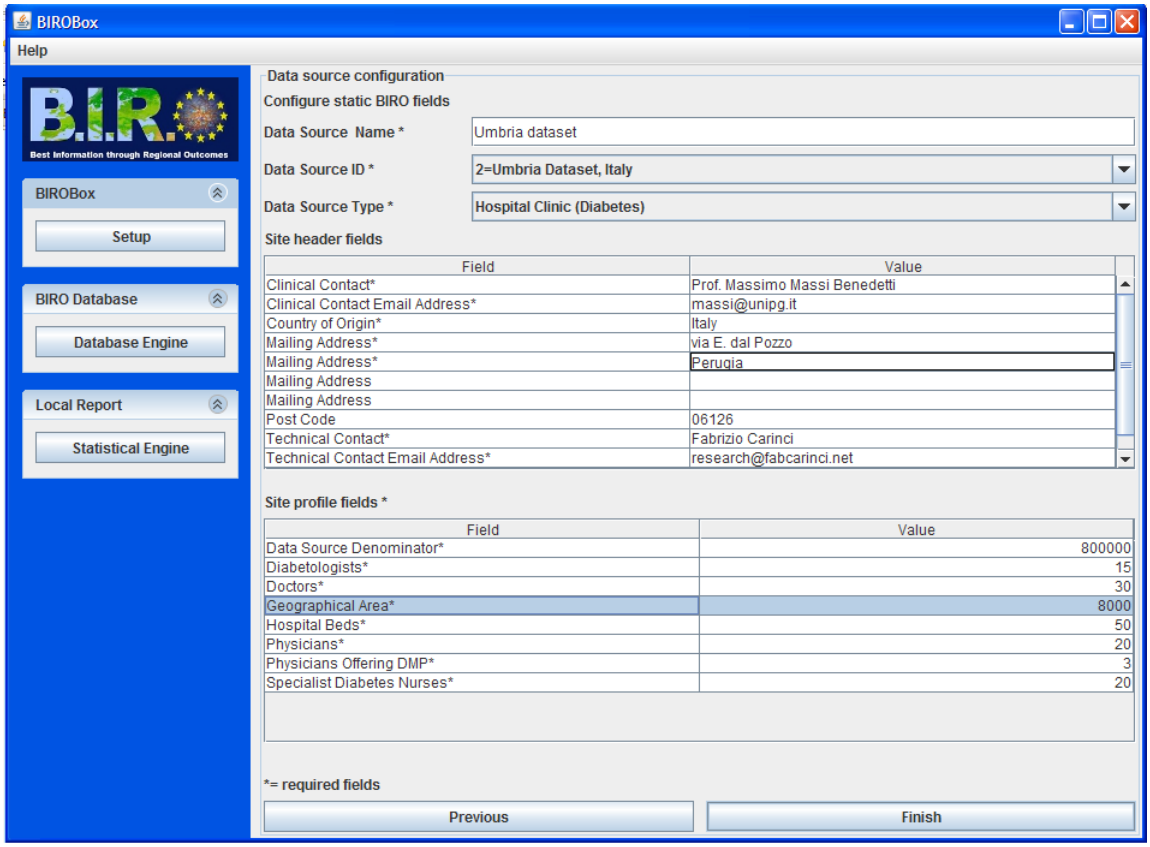


Figure 7: Site profile panel

The **Data Source Profile Panel** allows the user to specify extra information that cannot be included in the input datasets, as it involves a description of the organizational structure of the local data source.

The panel includes three main sections (**Figure 7**):

- **General References**
- **Site Header**: a form to be filled with information about the data source contacts and relevant addresses
- **Site Profile**: a form to be filled with information concerning the catchment area, workforce and capacitty of the local data source

In the **General References**, the user must provide the name of the data source, the ID from the list of known EUBIROD centres, and the type of data source among a list of known categories.

The **Site Header and Site Profile** sections must be filled using the fields included in the right column of the site header and site profile table.

All fields marked with an asterisk are **mandatory**. If the user forgets to fill any required field, the configuration validator will pop up a warning message and will disable the import function.

6.5.5. Datasets Editor Panel

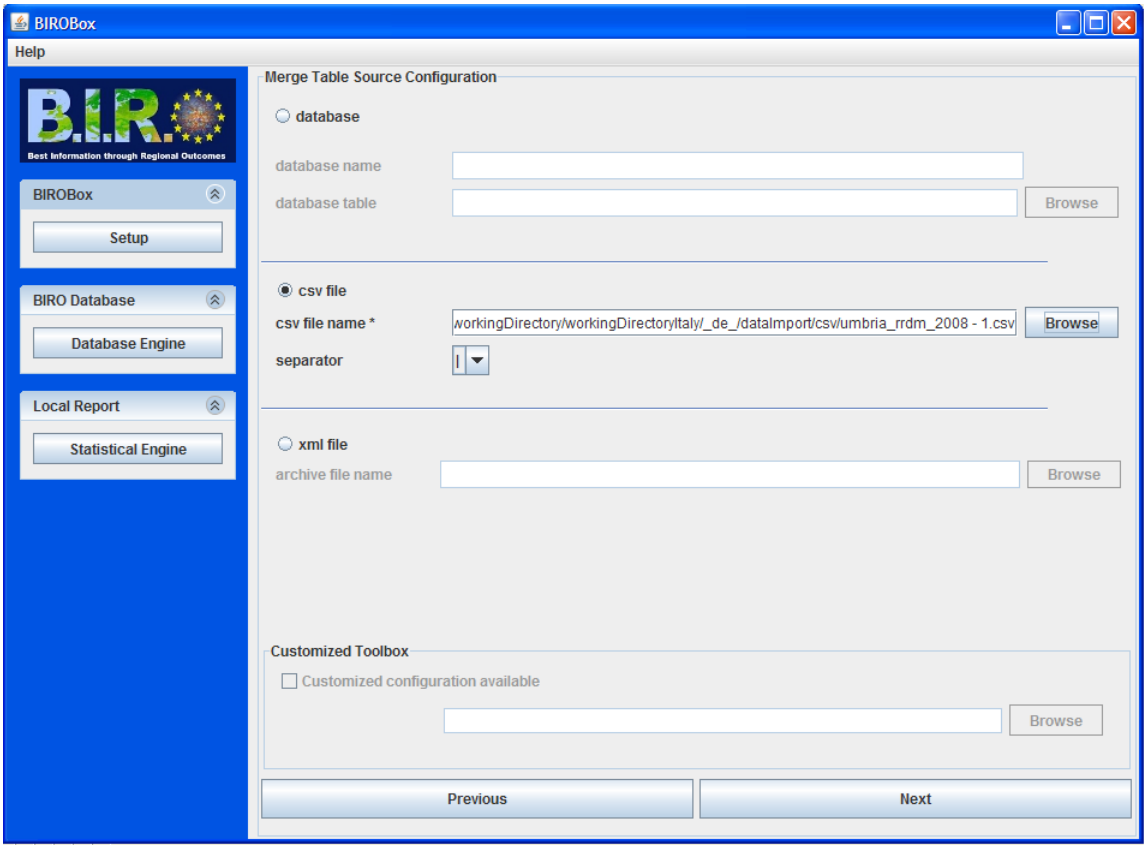


Figure 8: Dataset Editor Panel

The **Dataset Editor Panel** allows the user to select the dataset type and to specify its location, or to use a Customized Toolbox to create the target dataset from multiple datasets through an external script ("Customized Toolbox").

The panel has the same layout for all input datasets (Merge Table, Activity Table, Population Table, Diabetic Population Table). **Figure 8** shows the panel in the case of the Merge Table.

The panel includes three sections:

- **dataset type section**: a list of three alternative dataset types with the corresponding specific settings (table and database name for database, file path for CSV files and XML archives)
- **customized toolbox section**: it may be optionally used to specify the customized transformation file, when available. This section is enabled only when the database or the .csv file option is selected
- **navigation bar**: the "Previous" button leaves the panel and returns to the Configuration Editor Panel. The "next" button is enabled only when mapping is required, i.e. when a database name or the .csv is supplied (see next section). If an XML archive or a customized transformation is invoked, the BIROBox does not need further information to import local data.

Depending on the format of the local dataset, the user must provide specific information:

- the database name and the table name for database sources
- the file path and the column delimiter character for CSV files
- the file path for XML archives

The user may click the “Browse” button to search for the desired source. Once finished, the user may go to the next step to proceed for mapping to the BIRO format, or go back to the Configuration Editor Panel and choose another input dataset to be edited.

If the user cannot provide specific field mappings or the local data cannot be matched easily to the BIRO dataset, it could be possible to use the Customized Toolbox option.

Currently the Customized Toolbox option is only implemented for .csv files.

The **Customized Toolbox** is a BIROBox add-on allowing the user:

- to overcome the difficulties of mapping the local dataset to the BIRO format
- to create the tables required by the BIROBox through complex transformations of multiple datasets stored at the level of the local source

The customized Toolbox wraps up an open source ETL tool, Pentaho, implementing advanced integration techniques known as Business Intelligence. The Customized Toolbox acts as a black box invoked from within the BIROBox. The file included in the specific field (kettle transformation) is a an XML format describing the transformations processed by Pantaho.

In other words, such file describes all the steps required to create any required BIRO dataset from one or more local datasets. Pentaho transformations can be saved by the users using the Pentaho graphical user interface, named Spoon. Although Spoon is not embedded within the BIROBox, it can be freely downloaded from http://*kettle.pentaho.com/*. Otherwise the user may ask support from the EUBIROD Core Development Team to help defining the transformation.

6.5.6 Field Mapping Panel

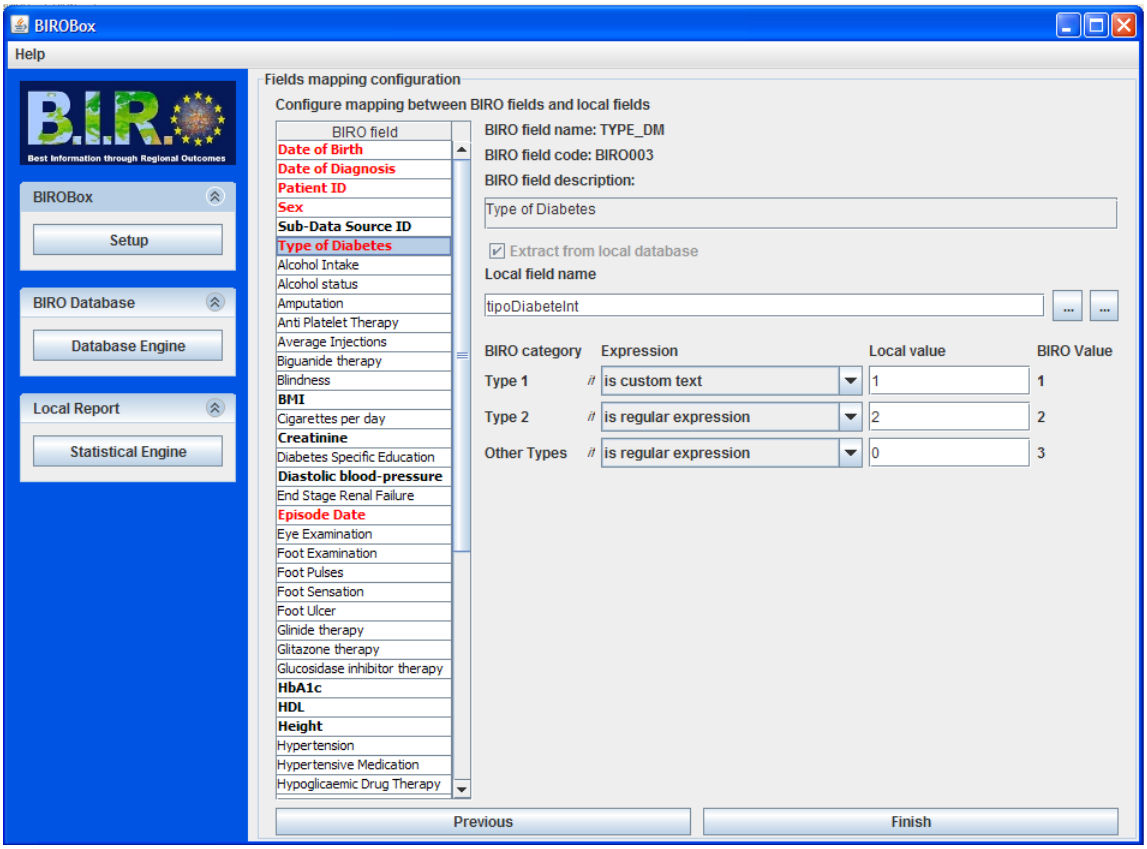


Figure 9: Field Mapping Panel

The **Field Mapping Panel** allows the user to:

- choose the fields to be extracted from the local dataset
- map the columns names to the BIRO field names
- specify the local data format for each field so that the BIROBox could perform the necessary transformations

The Field Mapping Panel includes the following sections (**Figure 9**):

- **BIRO field list:** contains the list of all BIRO fields in alphabetical order. When a field is selected, the corresponding form is displayed on the right side of the panel. Fields marked in red represents mandatory items. Field marked for extraction are highlighted in bold.
- **Data dictionary section**: displays information included in the BIRO data dictionary e.g. the BIRO field name, code and description for each field selected
- **Mapping section**: it is the interactive part of the panel. It allows the user to mark a field for extraction, to specify the local field name and the local format. The extraction check box is disabled for mandatory items. The layout changes according to the BIRO field type (date, numeric, enumerated)

In order to speed up the process of mapping BIRO fields, you can inspect the local data source by selecting the special **"..."** squared buttons at the right side of the panel. These buttons open two pop-up windows containing the **list of the local column names** and an **extract of the local dataset** (first 50 rows).

For **date fields** the user must specify the date format adopted by the local data source.

The sliding menu provides the user with the most common date patterns.

If no default date pattern matches the desired date format, the user may simply edit one of the format and create a customized pattern according to common rules.

The following table describes symbols that can be used to create a customized date pattern (e.g. "dd MMMM yyyy" for "12 December 2010"):

Symbol	Meaning	Type	Example
G	Era	Text	"GG" -> "AD"
y	Year	Number	"yy" -> "03″ "yyyy" -> "2003″
M	Month	Text or Number	"M" -> "7″ "M" -> "12″ "MM" -> "07″ "MMM" -> "Jul" "MMMM" -> "December"
d	Day in month	Number	"d" -> "3″ "dd" -> "03″
h	Hour (1-12, AM/PM)	Number	"h" -> "3″ "hh" -> "03″
H	Hour (0-23)	Number	"H" -> "15″ "HH" -> "15″
k	Hour (1-24)	Number	"k" -> "3″ "kk" -> "03″
K	Hour (0-11 AM/PM)	Number	"K" -> "15″ "KK" -> "15″
m	Minute	Number	"m" -> "7″ "m" -> "15″ "mm" -> "15″
s	Second	Number	"s" -> "15″ "ss" -> "15″
S	Millisecond (0-999)	Number	"SSS" -> "007″
E	Day in week	Text	"EEE" -> "Tue" "EEEE" -> "Tuesday"
D	Day in year (1-365 or 1-364)	Number	"D" -> "65″ "DDD" -> "065″
F	Day of week in month (1-5)	Number	"F" -> "1″
w	Week in year (1-53)	Number	"w" -> "7″
W	Week in month (1-5)	Number	"W" -> "3″
a	AM/PM	Text	"a" -> "AM" "aa" -> "AM"
z	Time zone	Text	"z" -> "EST" "zzz" -> "EST" "zzzz" -> "Eastern Standard Time"
'	Excape for text	Delimiter	"'hour' h" -> "hour 9″
"	Single quote	Literal	"ss"SSS" -> "45′876″

For **numeric fields** the user must choose the measurement unit adopted by the local data source among those listed in the sliding menu. Currently it is not possible to add any other unit of measurement at runtime. If the local measurement unit it is listed among those available, the user must change the data source manually and run the mapping again using the standard schemes, or skip the field extraction.

Simple numeric fields, e.g. patient ID and BMI do not require any mapping.

For parameters that can present only a discrete number of distinct categories (**enumerated fields**) the user must configure each correspondence between a specific local category and the relative enumerated value expected by the BIRO standard. Several choices are possible: null value, any string, null or any string, regular expression, custom text. While the custom text option is used to map straight one-to-one relations, regular expressions can be used to map one-to-many relations.

Example: the mapping of the Type of Diabetes field represents a common example of one-to-one and one-to-many mapping. Consider the following situation: the local data records envisage four different values for the parameter “Type of Diabetes”:

- 1 = Type 1 diabetes
- 2 = Type 2 diabetes
- G = Gestational diabetes
- M = Monogenic diabetes

The BIRO standard includes only three enumerated values:

- 1 = type 1 diabetes
- 2 = type 2 diabetes
- 3 = other types of diabetes

In order to map type 1 the user has to select the “custom text” option and write the text “1” in the corresponding text field.

The same applies for the mapping of type 2 values.

Gestational Diabetes and Monogenic Diabetes should both be mapped to other types of diabetes. This can be done by selecting the “regular expression” option and writing as argument an expression representing the logical OR of local values for gestational and monogenic diabetes, e.g. “G|M”.

6.5.7. Inspector Panel

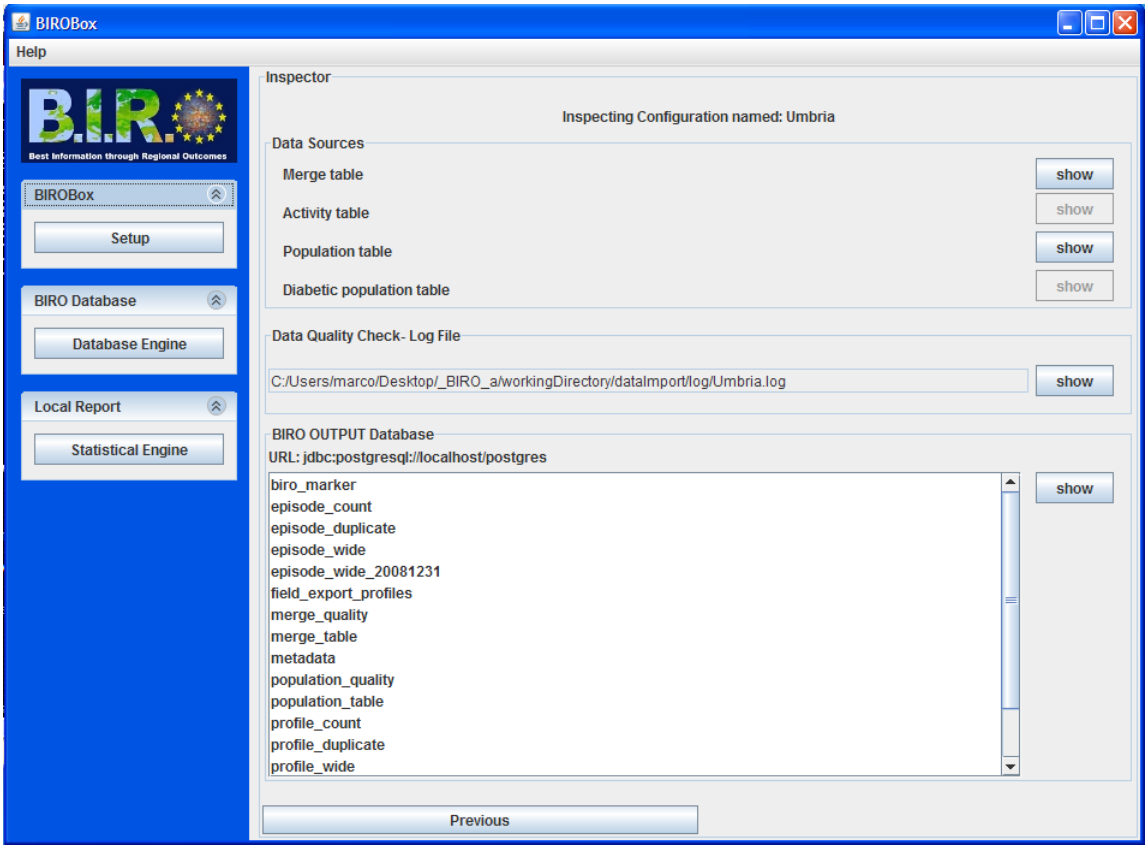


Figure 10: Inspector panel

The aim of the Inspector Panel is to allow comparing the content of the imported BIRO Output database with the corresponding input datasets.

The inspector panel includes the following sections (**Figure 10**):

- input dataset
- data quality check log file
- BIRO output database

On the top of the panel the user finds a reminder of the configuration under inspection

The user may open one of the four expected input dataset (Merge Table, Activity Table, Population Table, Diabetic Population Table) by clicking on the corresponding “show” button.

The “show” button may be disabled if:

- the button corresponds to one of the optional datasets (Activity Table, Diabetic Population Table) configured as not available
- the XML archive has been configured as data source for the Database

The user may find all tables in the BIRO output database, if any, listed in the white panel.

Clicking the **“show”** button on the right will open a pop up window showing the selected table.

The database URL is displayed as a reminder on top of the white panel.

The user may open the **data quality log file** by clicking on the corresponding "show" button. The quality check report provides the user with an overview of the local data quality. The report includes the following information for the four fundamental datasets (Merge Table, Activity Table, Population Table, Diabetic Population Table):

- Total number of :
    - missing values (absolute value and percentage)
    - values with wrong format (absolute value and percentage)
    - values out of range (absolute value and percentage)
    - inconsistent values (absolute value and percentage)
    - duplicate records
    - records with non admissible values in any required field
- Distribution of:
    - missing values in each field (absolute value and percentage)
    - values with wrong format in each field (absolute value and percentage)
    - values out of range in each field (absolute value and percentage)
- Detailed list of errors:
    - values with wrong format
    - values out of range
    - inconsistent values
    - duplicate episode records
    - overlapping cells in duplicate episode record
    - records with non admissible values in any required field

Only for the Merge Table, the data quality log file provides also the following information about duplicate records:

- Total number of duplicate records
- Detailed list of errors:
    - duplicate episode records
    - overlapping cells in duplicate episode record

The data quality log file can be extremely useful to understand the nature of any problems and inconsistencies found at the local data source.

Since all errors are listed along with the related record id, it is possible to correct the local dataset and run the import again.

The check does not prevent the user from **improperly import the inconsistencies** found in the local datasets to the BIRO database. In certain situations, this can undermine the validity of the statistical reports produced by the BIROBox.

7. Statistical Engine

The statistical engine section allows the user to configure and launch the statistical analysis on the imported data and to explore the results delivered in the form of .pdf and .html reports.

7.1. Statistical Engine Panel

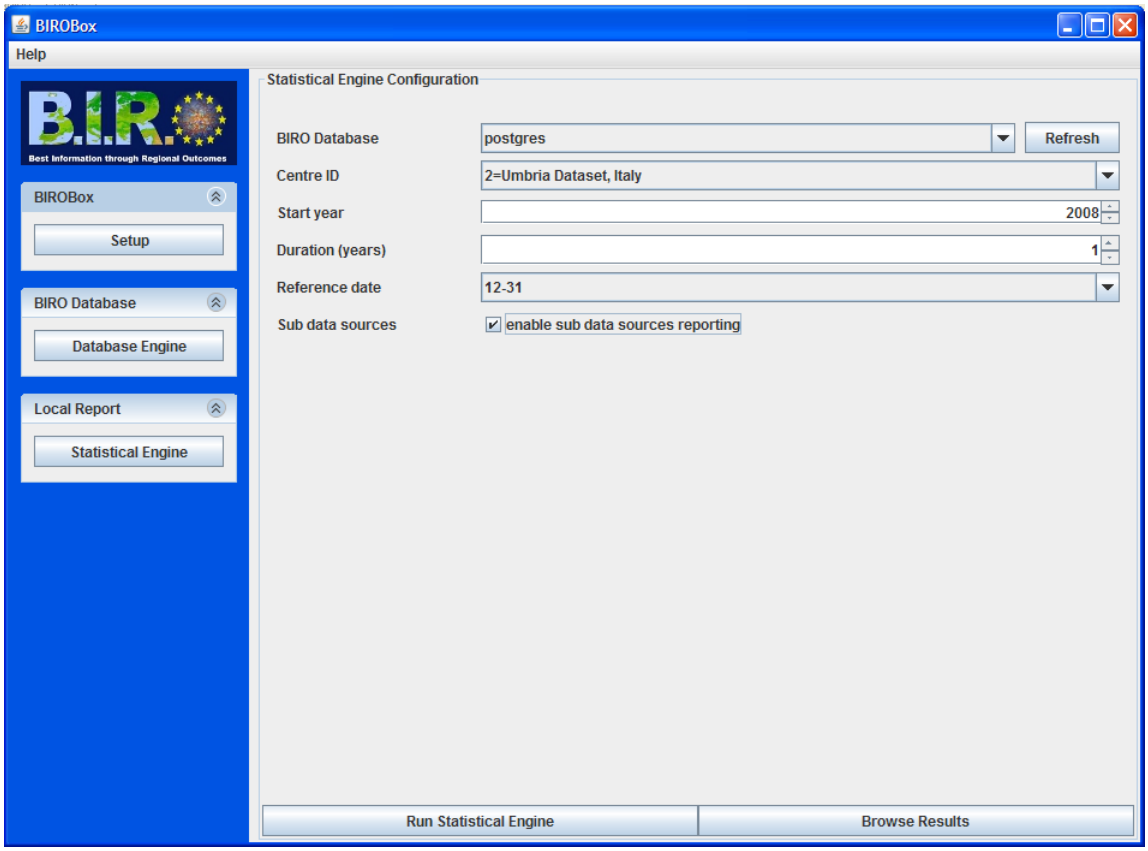


Figure 11: Statistical Engine Panel

The user can select the “statistical engine” through the specific button located at the left side of the main function panel. Through this panel (**Figure 11)** the user may define the few settings required by the statistical engine to run properly. The navigation section included at the bottom allows for the user to directly run the statistical engine to access the results panel once the process has finished.

In order to run the statistical engine the user needs just to configure a few parameters:

- **BIRO Database**: the user can choose the database to be processed by the statistical engine from the sliding menu. It is possible that some of the existing databases are not listed in the menu. The user will find only databases that are suitable for the statistical engine. If the database import process has failed for any reason, such database will not be included in the list. Clicking the “refresh” button triggers a scan of the databases to help identifying the preferred one.
- **Centre ID**: the user must select the target identifier of the local data source, among those listed in the sliding. This way the statistical engine can mark any result with the correct user

identifier

- **Start year**: a list allows the user to select the year to be examined by the statistical engine. Based on the episode dates included in the BIRO target database, the statistical engine will extract only a subset of records starting from the selected year.
- **Duration**: although the BIRO report refers to a default period of one year, the user may decide to run the analysis on an extended period using the duration list. Please note that setting the duration at N will not allow to deliver a report on N years, but will result into N yearly statistical reports.
- **Reference Date**: the user has the possibility to choose between two reference dates (December 31st or June 30th)
- **Sub Data Sources**: this check box allows the user to perform stratified statistical calculation of diabetes indicators over sources identified by a specific sub data source ID.

Once the user is satisfied with a specific configuration, the statistical engine may be directly run simply by clicking the “Run Statistical Engine” button.

An **execution window will pop up**, showing the user the progress of the calculation and providing details for all actions performed by the statistical engine (**Figure 12**). Please note that the progress bar is based on the fraction of indicators computed over the total expected, and not on an estimation of the execution times. Therefore, the progress is only indicative and cannot be related to the actual time to wait for the end of the execution.

The user may find the same information included in the execution in a .log file named as the input database.

The file can be found at the following path:

- <Working Directory>/_se_/output/reports/<timestamp>/<year>/<sitecode>.

If the statistical engine is interrupted by any error, this file is not created and the corresponding information can be found at the file “statisticalEngineSinkFile.txt” located in the statistical engine output folder under the user Working Directory.

The correct execution of the Statistical Engine will be notified by the progress bar reaching 100%.

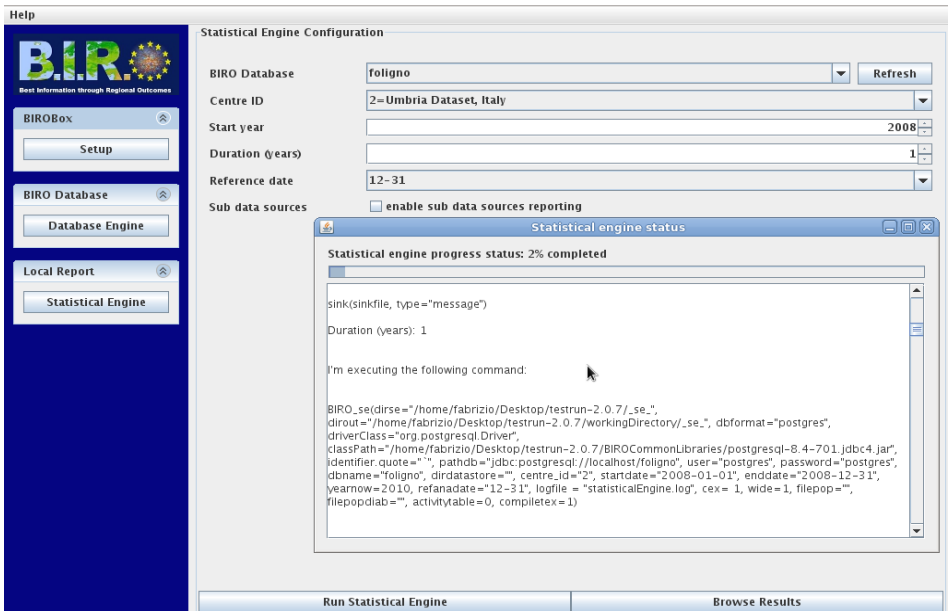


Figure 12: Statistical Engine Progress Window

The **execution time** of the statistical engine strongly depends on the amount of data to be processed. Be prepared to wait for long execution times for very large datasets.

The following table presents the average execution times observed on a standard notebook for the statistical engine running under BIROX with 1024 Mb memory assigned to the VirtualBox:

Centre	N Patients	N episodes	Elapsed Time	
	1	2,842	9.097	10' 46″
	2	3,202	8,316	9' 23″
	3	1,115	1,948	8' 26″
	4	1,268	1,456	8' 17″
	5	994	1,329	8' 02″
	6	318	438	8' 19″
	Overall	9,739	22,584	24' 52″

The output produced by the statistical engine can be directly accessed from within the BIROBox, simply by clicking on the “Browse results” button (see next section).

7.2. Statistical Engine Results Panel

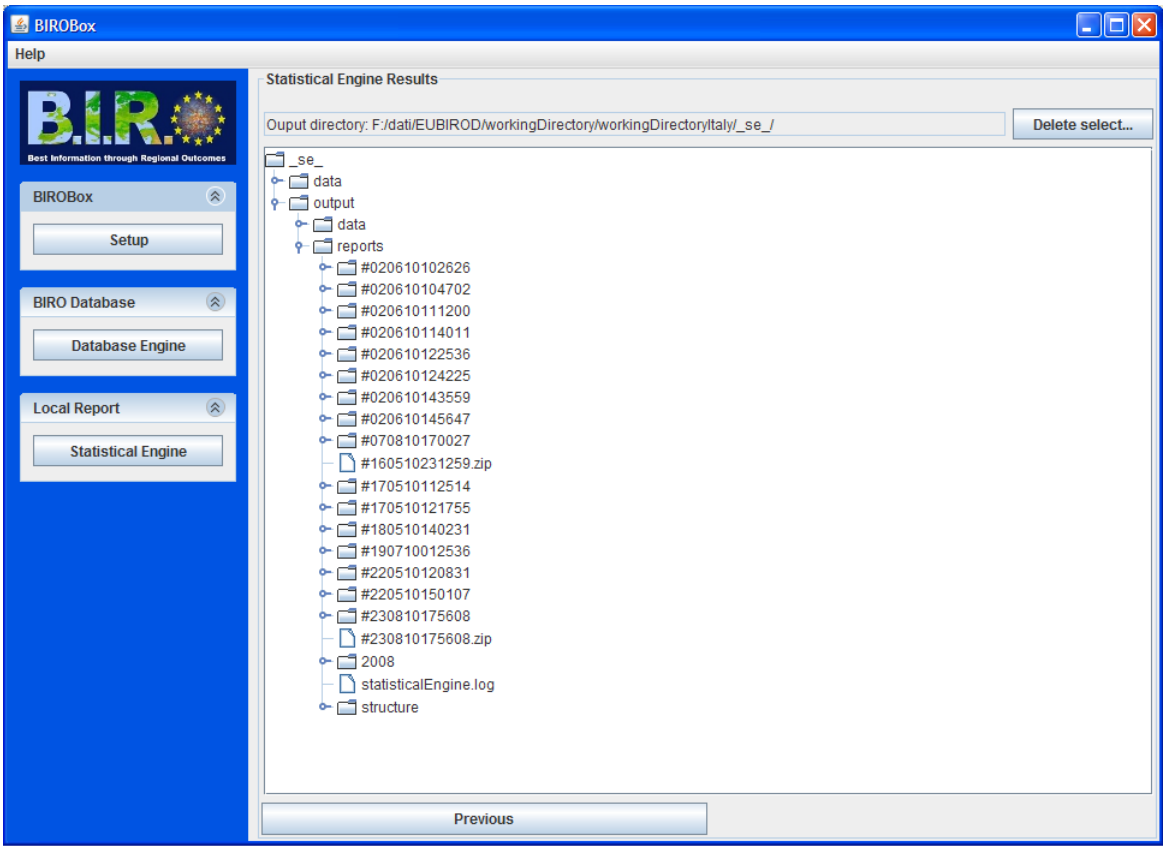


Figure 13: Statistical Engine Results Panel

The statistical engine results browser allows the user to browse the output directory of the statistical engine and examine results of all statistical analyses (**Figure 13**).

The statistical engine results browser allows the user to inspect not only the results produced by the last statistical engine process, but also those resulting from all processes previously launched using the same Working Directory.

At the top of the panel, the user can find a pointer to the output directory, allowing to easily locate statistical results under the specific file system.

The panel presents a file system tree map with the statistical engine output directory at his root.

The user may navigate the statistical engine tree map as in any traditional file system browser. A double click on a folder icon will display its content. A double click on a leaf element will open the corresponding file using the most appropriate application.

The statistical engine tree will display the following items:

- ***data***: folder containing .csv export files of patient profiles and episodes representing the statistical engine cohort.
- ***output***
  - ***data***: folder containing the .csv files of statistical objects (aggregated data to be sent to

the central engine)

- ***report***: folder containing the statistical reports and all its components
  - ***<#timestamp>*** *(Every time an analysis is run, the statistical engine creates a new folder named with the corresponding timestamp)*
    - ***<year>***
      - ***<data source id>***
        - *report.html:* the index page of the HTML report in HTML format
        - *report.pdf:* the report in PDF format
        - *graphs*: folder containing all graphs included in the report in SVG, PDF, JPG, PNG format
        - *html*: folder containing the HTML pages of the report
        - *pdf:* folder containing all the latex component of the PDF report
        - *tables:* folder containing all tables in the report, made separately available in HTML format
        - *wp:* folder containing all the HTML pages of the report, specifically formatted for the EUBIROD Web Portal

To **remove obsolete outputs** from the output directory of the statistical engine, the user can select an element of the tree and then click the **"delete"** button at the top of the screen. Please note that this will irreversibly remove the selected element from your computer.

7.3. Direct Use of the Statistical Engine Output Directory

The main reports produced at each run of the statistical and central engine are directly accessible in a sub-folder of the working directory located under _se_/output/reports, where a directory named with the date and time of the start of execution is created (timestamp). The pdf and html reports are saved each time under the relative timestamp directory and subdirectories with the reference year and centre code for the statistical engine, region code for the central engine.

Figure 14 presents a practical case in which the report for Regione Umbria is included under the specified chain of subfolders.

The html report can be directly accessed by double clicking on the main .html file, named with the database name. The browser will display the table of contents listing all indicators. For those indicators that can be computed by the statistical/central engines (based on the availability of the basic variables originally included in the mapping), links will be active and can be opened by clicking on the specific code/description.

An example of an indicator subpage directly accessible through the main html report is displayed in **Figure 15**. This includes a long list of html tables that are also saved for use in the web portal under the “tables” subdirectory.

The above html files can also be particularly useful to find a particular image that can be included in slides/presentations or high quality typographical outputs.

A quick tip to identify a graph of interest for a specific indicator is shown in **Figure 16** through the use of the Firefox browser: by right clicking on a page, and selecting “Page Info”, the user can access a form that includes a “Media” tab. Selecting it would display a list of all graphical files (png, svg) included in the page. When the user clicks on a specific files, a preview will be available. The location will be printed in the form, from which it can be cut/paste in the browser window, or simply used to access the file. A pdf version of the same file (not visualizable in Firefox) will be also available in the same directory.

Figure 17 displays the contents of the graphs directory, which can be quickly navigated using default image viewers as an additional resource to select the most convenient outputs.

An example of a pdf report is displayed in **Figure 18**. This is directly accessible by clicking on the main page.

Statistical objects are all saved in the data directory, also located under the above timestamp sub directory, in the branch _se_/output/data. The aggregate data saved in CSV format can be directly displayed using an ordinary text editor, as in the case shown in **Figure 19**. The CSV files include a transparent definition of all the variables in the first row, using the normal conventions adopted for this type of files.

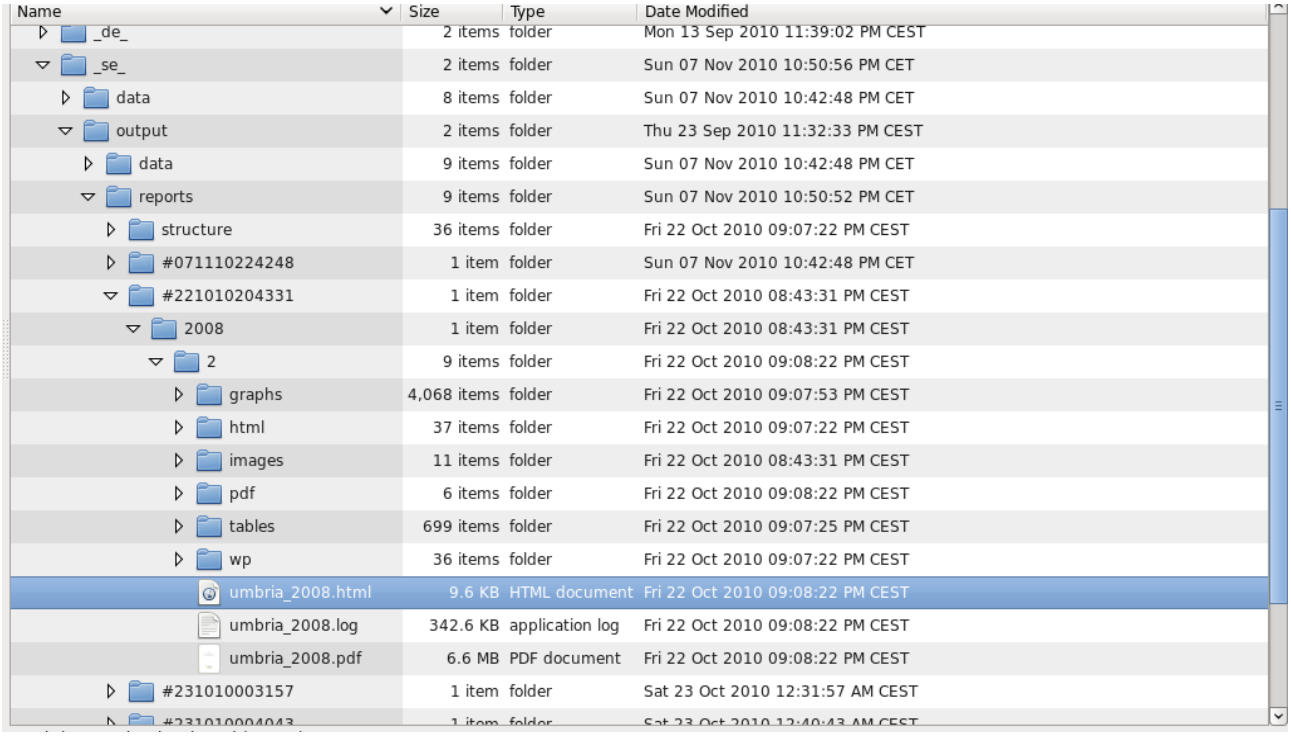
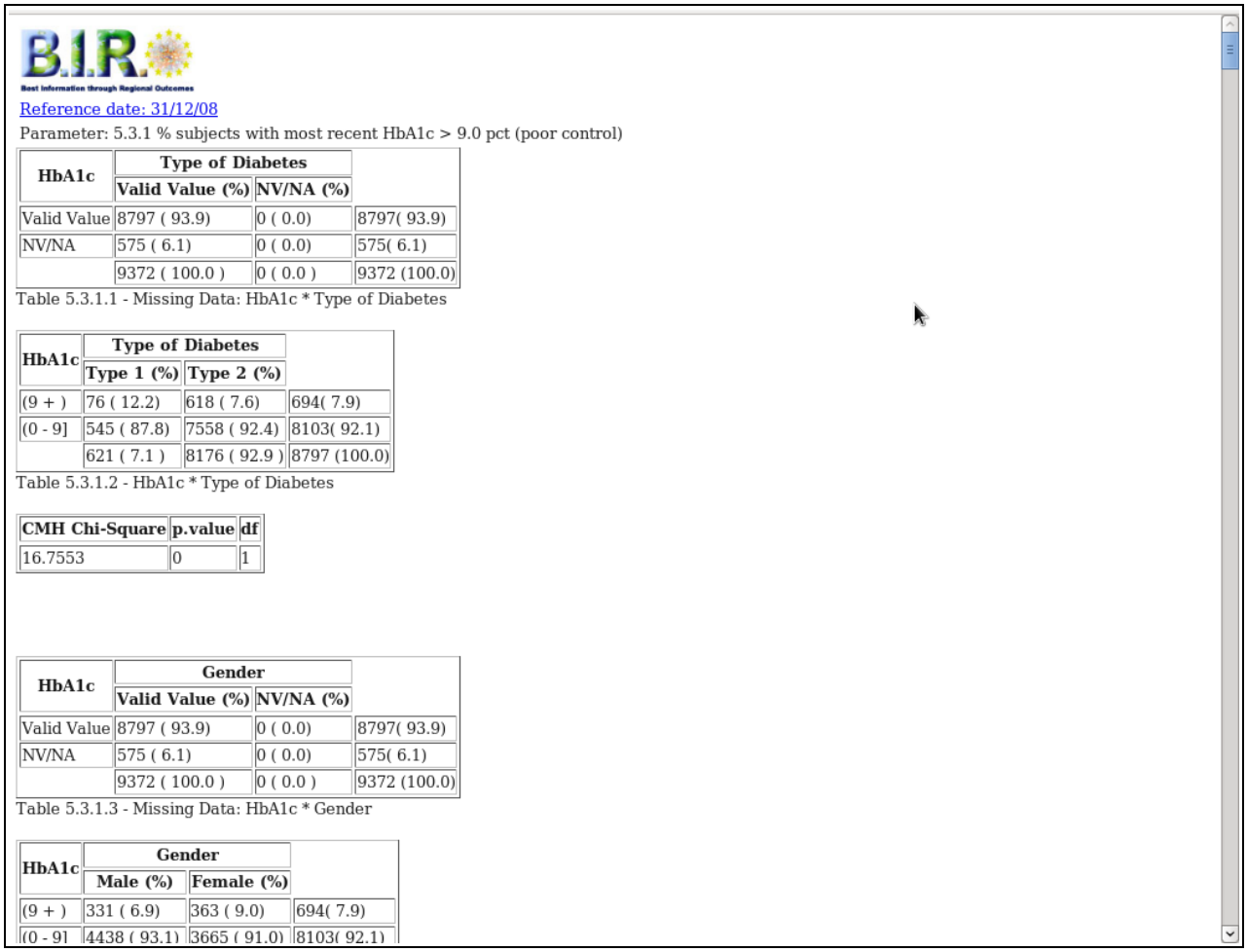


Figure 14: Selecting Outputs from the EUBIROD Output Directory



Best Information through Regional Outcomes

Reference date: 31/12/08

Parameter: 5.3.1 % subjects with most recent HbA1c > 9.0 pct (poor control)

HbA1c	Type of Diabetes		
	Valid Value (%)	NV/NA (%)	
Valid Value	8797 (93.9)	0 (0.0)	8797(93.9)
NV/NA	575 (6.1)	0 (0.0)	575(6.1)
	9372 (100.0)	0 (0.0)	9372 (100.0)

Table 5.3.1.1 - Missing Data: HbA1c * Type of Diabetes

HbA1c	Type of Diabetes		
	Type 1 (%)	Type 2 (%)	
(9 +)	76 (12.2)	618 (7.6)	694(7.9)
(0 - 9]	545 (87.8)	7558 (92.4)	8103(92.1)
	621 (7.1)	8176 (92.9)	8797 (100.0)

Table 5.3.1.2 - HbA1c * Type of Diabetes

CMH Chi-Square	p.value	df
16.7553	0	1

HbA1c	Gender		
	Valid Value (%)	NV/NA (%)	
Valid Value	8797 (93.9)	0 (0.0)	8797(93.9)
NV/NA	575 (6.1)	0 (0.0)	575(6.1)
	9372 (100.0)	0 (0.0)	9372 (100.0)

Table 5.3.1.3 - Missing Data: HbA1c * Gender

HbA1c	Gender		
	Male (%)	Female (%)	
(9 +)	331 (6.9)	363 (9.0)	694(7.9)
(0 - 9]	4438 (93.1)	3665 (91.0)	8103(92.1)

Figure 15: Opening the EUBIROD HTML Report

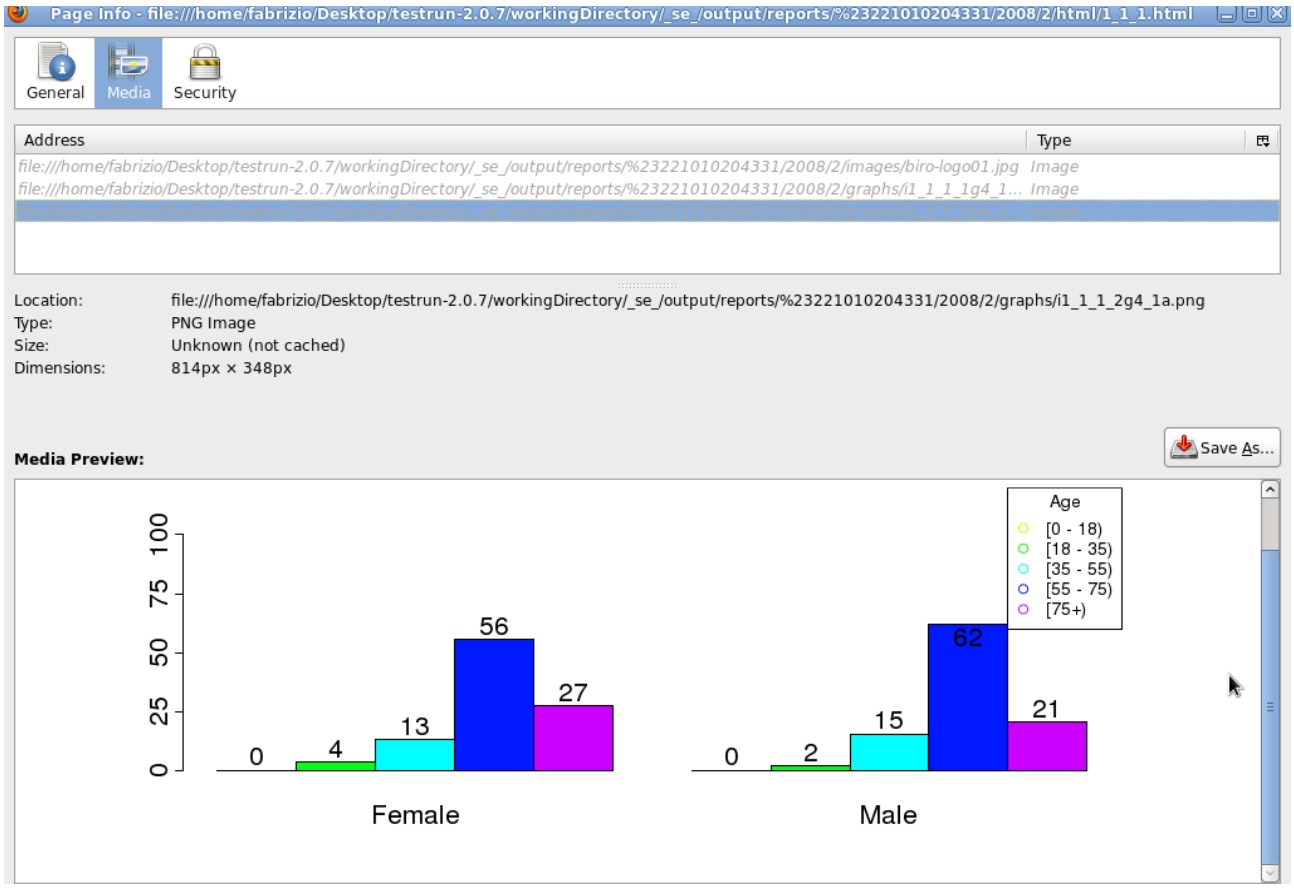


Figure 16: Using Firefox to Browse EUBIROD graphs in the HTML Report

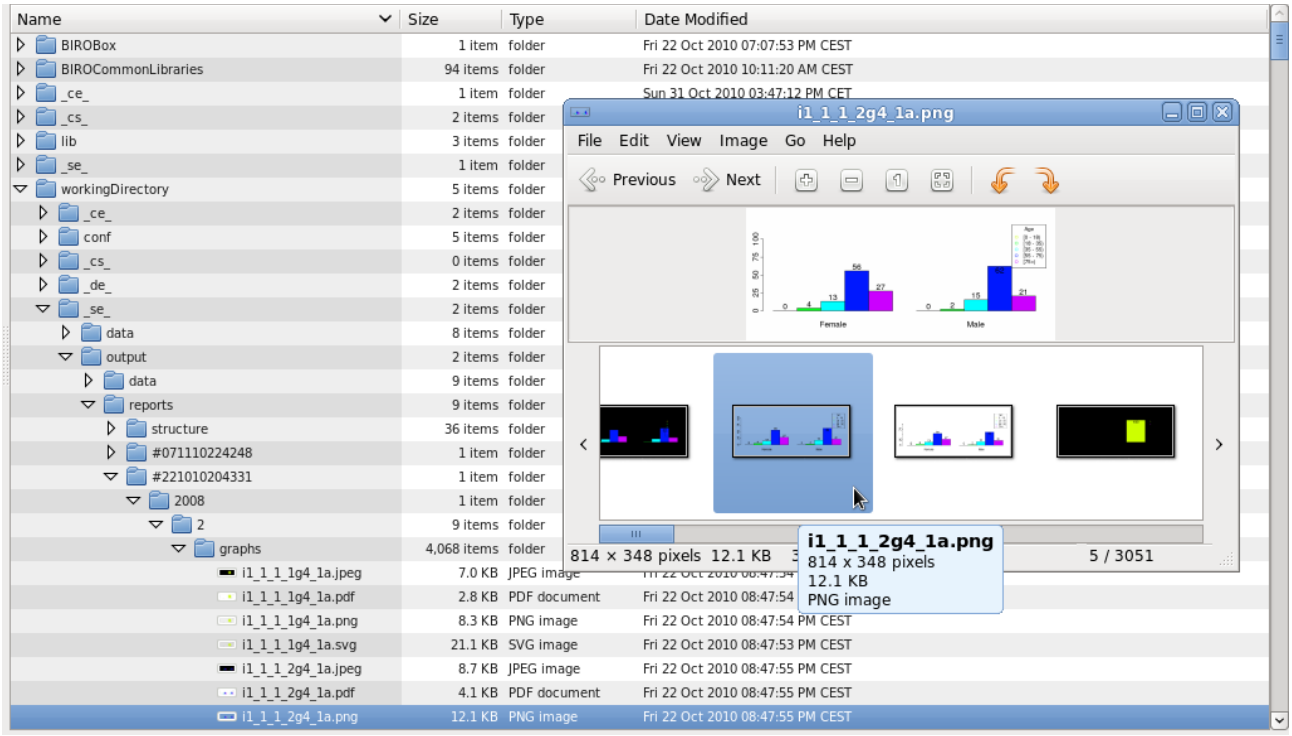


Figure 17: Selecting Images directly from the EUBIROD Graphs Directory

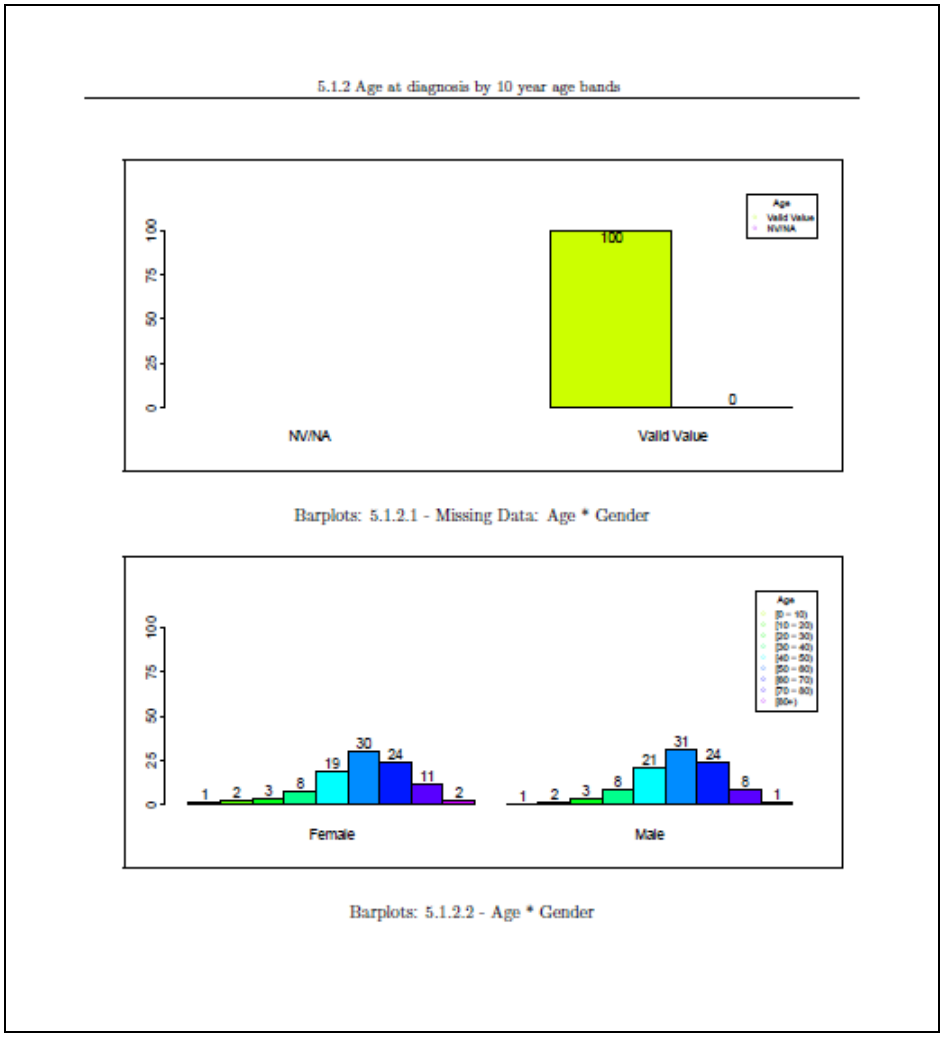


Figure 18: Opening the EUBIROD PDF Report

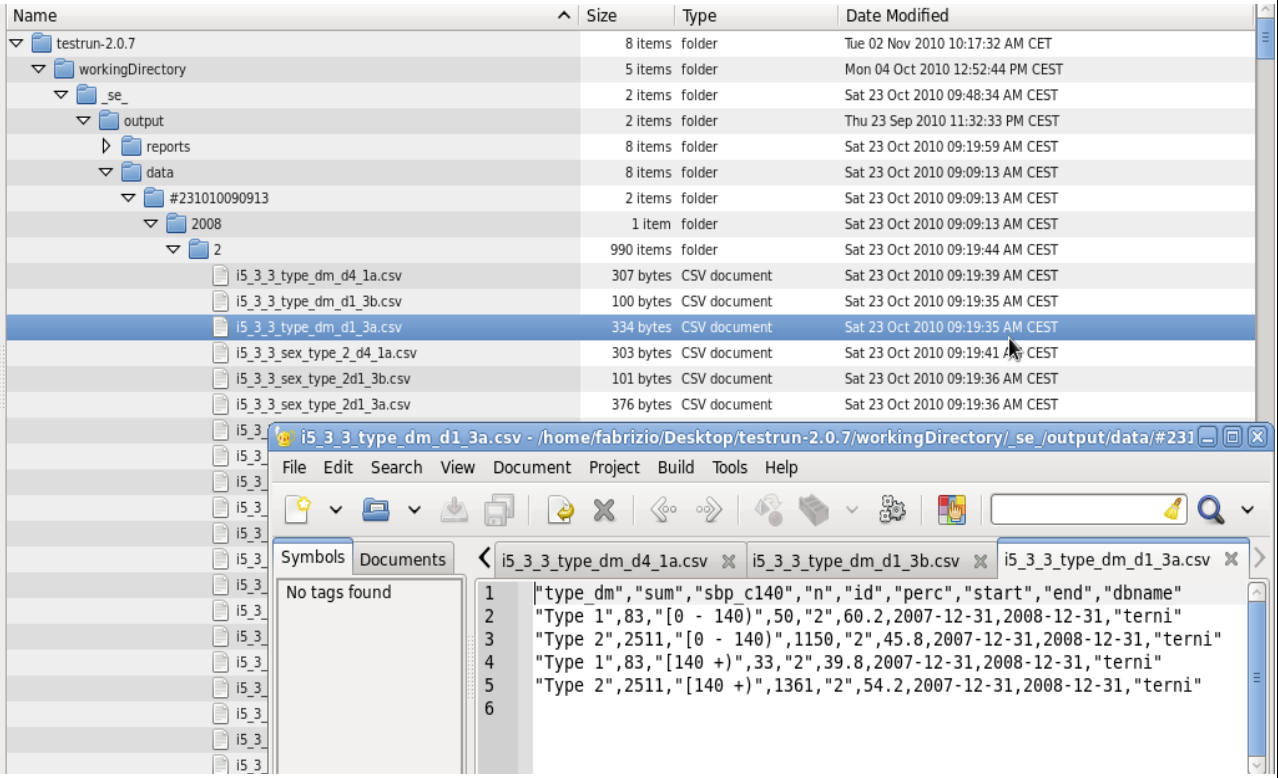


Figure 19: Browsing the EUBIROD Output Data Directory

7.4. Interpreting BIRO Reports

A general epidemiological perspective has been applied to plan all tables and graphical outputs to be produced in EUBIROD.

The structure of the BIRO Reports is shown in **Figure 20**.

The pdf version is the most complete and user friendly, as it includes bookmarks, the list of all BIRO/EUBIROD contributors, a set of basic Help files to help the reader in the interpretation of results, and the parameters used by the statistical routines to produce the report.

For each indicator, the statistical engine produces a root and a body table. In the most complex case where two exposures and the class variable are present, the outputs include a separate table for each exposure and both exposures for each level of the class variable.

For instance, if the two exposures are Age, Gender, and the class variable as usual is Type of Diabetes, the report will include a root and a body table for Age, Gender, Age*Gender for Type 1, Type 2, and Other Type of Diabetes.

In each report, and for each indicator, the table section is followed by a list of graphs for all the variables included in the process. These include barplots for categorical variables, and boxplots, trellis plots for continuous variables. All graphs are stratified by sub source (usually centres or regions) if the option “enable sub data source reporting” is selected.

For risk adjusted indicators, the report includes additional tables of standardized rates and observed minus expected excess/reduction along with 95% confidence intervals.

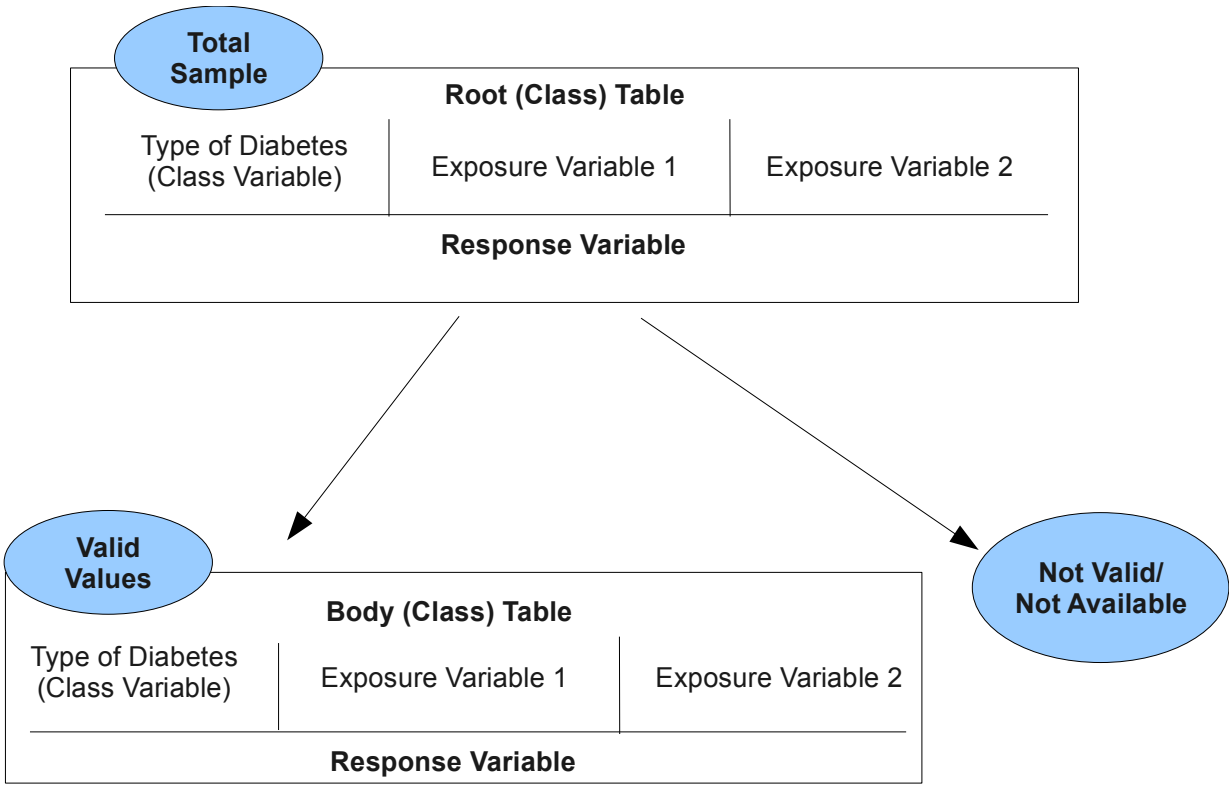
Graphs included in these outputs are barplots and forest plots. Maps and longitudinal trends have been planned but not yet implemented due to the need of identifying a common solution for geographical coding and unique IDs to be used by the central engine (see Discussion).

The BIRO report includes a “**root table**” and a “**body table**” for each tabular output. This was required to assess the impact of missing values on all tables. In fact, unless missing values are reported as a separate class for each reported table (unusual), patterns of observations included in frequency tables may hide the influence of values discarded from the tabulation due to missing values. Furthermore, tables reported for separate indicators would usually show different total number of observations, which can appear very confusing for the reader.

The “**root table**” reports the same total number of observations for all indicators. Observations are then stratified by valid/non valid values for each variable taken in account in the separate tables. Since only observations with valid values for all variables used in the tables are duly reported in frequency tables, only one cell in the “root table” would include the total number of observations passed to the “body table”. This way, the possible influence of the composition of the “root table” on the reported “body table” is clearly displayed to the user.

Tables include the following features:

- all frequency tables produced for each indicator include **one target response (outcome)** and up to **two cross tabulated exposure factors**. If the target response has i categories, exposure 1 has j levels and exposure 2 has k levels, the resulting table has a total of (i*j*k) cells. For example, a binary indicator of low/high level of HbA1c (two categories), stratified by four different age bands and sex (two levels), implies the construction of a table with 2*4*2=16 cells.



Body (Class) Graphs

BARPLOTS
Exposure Variable 1 (Exposure Variable 2)
Data Source
Response Variable=Categorical

TRELLIS / BOXPLOTS
Exposure Variable 1 (Exposure Variable 2)
(Data Source)
Response Variable=Continuous

Standardized (Class) Estimates
(Risk Adjusted Estimators)
Data Source
Response Variable

Standardized (Class) Graphs
BARPLOTS
FOREST PLOTS
Data Source
Response Variable

Figure 20: Structure of the EUBIROD Report

- The table is constructed with the **outcome in rows** and **exposure factors in columns**. A one way table includes only one exposure factor, while a two way table includes two of them. The columns are built by nesting levels of exposure factor 1 within each level of exposure 2. In the previous example, for each level of Hba1c, there will be four columns for males and four for females.
- **each (i,j,k) cell** in the table presents the absolute frequency of observations and the column percentage relative to the specific cross tabulation of exposure factors. In the previous example, the percentage of males/females with a specific level of Hba1c in each age band out of all males/females in the same age band.

  This representation allows the direct computation of the relative risk across different levels of the exposure factors, by dividing a certain percentage for the percentage shown in a different cell.
- The **column marginals** present the total number of observations for the specific exposure (exposure 1 for tables with two exposure factors), with row percentages computed over the grand total.
- The **row marginals** present the total number of observations for the specific level of the target response, with column percentages computed over the grand total.
- All tables include the calculation of the **Chi Square Test** (value and associated p>Chi-Square), to test the association between the exposure and the outcome of interest. When two exposures are included, the Chi-Square can be used to test the association between exposure 1 and the outcome of interest, stratified by levels of exposure 2.

Tables are stratified according to the values of the "**Class variable**", which triggers the creation of n tables for the target response and associated exposures, one for each level of the class variable.

In diabetes indicators, the usual "class variable" of choice is "**Type of diabetes**". Thus this option allowed us to replicate the production of all tables for all indicators for levels: Type 1, Type 2, Other Type.

All **graphical displays** are created according to a common structure for all strata of response, exposure and levels of the class variable.

Outputs can be furtrher stratified through the optional selection of the parameter "**sub_source_id**". Through this parameter, it is possible to display all outputs comparing levels of a certain variable, usually the centre ID. When data from multiple centres are present in the same data source, graphs may be used to benchmark results obtained by different centres against the overall average (ex: regions from different parts of Europe or centres within a region).

Outputs included in the BIRO report are explained in Appendix C and included as a preamble to all BIRO reports.

7.5. Risk Adjustment Methodology

Standardized estimators allow to rigorously compare quality of care and outcomes across different centres, regions or countries taking into account the possible imbalance in the case-mix, which can be systematically associated to systems performance. For example, a centre with older and sicker patients would normally experience higher rates of diabetic complications compared to the average population.

Risk adjustment methods allow standardizing all results against an ideal population usually corresponding to the total population target of the analysis. In the case of EUBIROD, the best comparison would be made against the European population.

The most advanced application of risk adjustment involves the use of multivariate models to assign weights for each risk factor of interest (exposure variable) on the rate of outcomes observed for a specific indicator. Since all risk adjusted indicators in EUBIROD are expressed in terms of binary outcomes (yes/no, low/high, etc), a natural candidate for the multivariate modelling approach is that of logistic regression.

The EUBIROD statistical routines implement the method adopted by the US Agency AHRQ for the calculation of standardized quality health care indicators.

Briefly, this work as follows:

- a multivariate model is run on top of the overall population based upon a specified outcome and a set of target covariates intended as potential risk factors (confounders). In quality of care, these can be assumed to be observed components of the case-mix that are potentially associated to the outcome of interest. Their effect shall be isolated by that potentially related to the quality of care delivered by a specific centre or region, which we may want to monitor or benchmark across a group of providers.
- weights extracted from the multivariate model are applied to each subject in the sample, applying the logistic model to compute an estimated probability of the outcome for that specific subject.
- the sum of the estimated probability across each centre or region is computed as the average “expected rate of events” (as specified for each indicator) for the particular centre.
- the quantity (observed rate/expected rate) is used as a multiplier (penalty if >1, premium if <1) of the average population rate to compute the “standardized rate” for each centre in the overall sample
- the percentage of observed minus expected over the expected number of cases for each centre is used as a measure of the excess/reduction of cases in each centre, compared to the average level
- all risk adjusted measures are published along with 95% confidence intervals, based upon a precise formula of the variance of the estimates.

Graphical display including barplots of standardized rates against the average, and forest plot of O-E/E%, with the related 95% confidence intervals, may offer an immediate representation of the variability of results across the whole sample of centres included in a report.

The technical details used for the calculation of EUBIROD risk adjusted indicators, directly obtained from the AHRQ, are included in **Box 1**.

Box 1. EUBIROD Risk Adjustment Method based on AHRQ Quality Indicators (modified from direct communication received from AHRQ Online Suport, Version 3.0, 23/5/2006)

The EUBIROD Statistical and Central Engine compute risk adjusted indicators using the method implemented by the AHRQ for quality of care indicators.

All of the AHRQ Quality Indicator routines begin with estimating a logit model of a 0/1 outcome variable and a set of subject-level covariates as dependent variables, and using the results to form the expected outcome for each subject (e.g. P=pr(outcome=1)).

I. Notation:

Y_{ij} = 0 or 1, outcome for patient j in centre i.
X_{ij} = covariates (e.g., gender, age, DRG, comorbidity)
P_{ij} = predicted probability from logit of Y on X
= $\exp(X_{ij}\beta)/[1+ \exp(X_{ij}\beta)]$
where β is estimated from logit on entire sample.
e_{ij} = $Y_{ij} - P_{ij}$ = logit residual (difference between actual and expected).
n_i = number of patients in sample at centre i.
α = average outcome in the entire sample[1] (e.g. Ybar).

II. Estimating the Risk Adjusted Rate (RAR) and SE using the *Ratio* Method[2] of Indirect Standardization for each Centre:

1. Estimating RAR:

let $O_i = (1/n_i)\sum(Y_{ij})$ be the observed rate at centre i
let $E_i = (1/n_i)\sum(P_{ij})$ be the expected rate at centre i

RAR_i
$= \alpha(O_i/E_i) = \alpha\,[(1/n_i)\sum(Y_{ij})]/\,[(1/n_i)\sum(P_{ij})]$ (where sum is for j = 1 to j = n_i)
= **population rate * observed/expected at centre i.**

2. Estimating Variance of RAR (SE is the square root):

$Var(RAR_i)$

$= Var[\alpha(O_i/E_i)]$
$= (\alpha/E_i)^2 Var[O_i]$ (since $var(aX) = a^2var(X)$ for any constant a)
$= (\alpha/E_i)^2 Var[(1/n_i)\sum(Y_{ij})]$ (by the definition of O_i)
$= (\alpha/E_i)^2 (1/n_i)^2 Var[\sum(Y_{ij})]$ (since $var(aX) = a^2var(X)$ for any constant a)
$= (\alpha/E_i)^2 (1/n_i)^2 [\sum Var(Y_{ij})]$ (since $var(\sum X_i)=\sum var(X_i)$ if Xi are independent)
$= \mathbf{(\alpha/E_i)^2 (1/n_i)^2 \sum [P_{ij}(1-P_{ij})]}$ (since Y is 0/1, so var(Y) = P(1-P))

[1] For the AHRQ QI, the sample is the entire reference population consisting of the discharges in the States Inpatient Database for the participating states pooled over three years (2001-2003). Therefore, the "average outcome for the entire sample" is the population rate.

[2] Risk-adjusted rate = (Observed rate / Expected Rate) * Population Rate

8. Updating the BIROBox

The distribution of the complete BIRO software using Debian-packages has the advantage that every change made in the software by the development team and its ensuing build with maven and Debian-package manager (dpkg) results in a direct update-notification in the Linux where the software is installed.

The user will be visually notified in the notification area of the top bar in Gnome.

A screenshot of a pending software update can be seen in Figure 21.

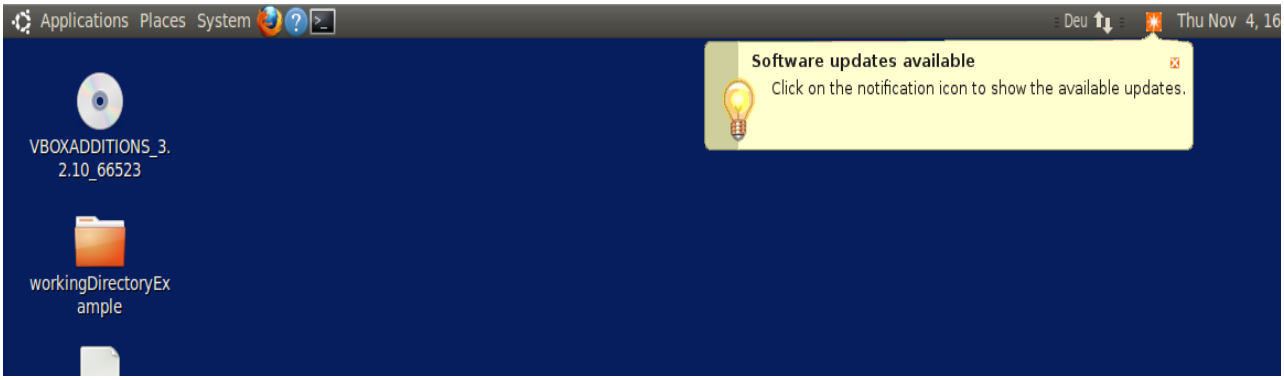


Figure 21 – Update notification of installed software in Gnome Desktop

The software can be easily updated when clicking on the notification area or using the terminal with the command:

```
$ sudo aptitude upgrade
```

9. Support

How can I ask for technical support?

This section explains how the user may report bugs, ask for further clarifications or request technical support for the usage of the BIROBox.

You may report bugs, ask for clarification or technical support by sending an email to the EUBIROD Coordination Centre (eubirod@unipg.it).

If you experienced any problem with the BIROBox configuration and the data import in the database engine section, please attach the content of the <conf> folder located in the Working Directory, and the Data Quality Log file, if available.

If you experienced any problem with the statistical engine, and the run has not been completed, please attach the file statisticalEngineSinkFile.txt located in the statistical engine output folder in your Working Directory. If a problem was noticed, but the run was completed, please attach the .log file in the specific output directory corresponding to the latest run of the Statistical Engine.

Appendix A. BIRO Indicators

The BIRO Report Template includes the following output indicators:

1. **Demographic characteristics**
   - **1.1. Age (Classes)**
2. **Clinical characteristics**
   - **2.1. Diabetes Status**
     - 2.1.1. Type of diabetes
     - 2.1.2. Duration of diabetes (Classes)
   - **2.2. Risk Factors**
     - 2.2.1. Obesity
       - 2.2.1.1. Weight (the most recent episode in 12 months)
       - 2.2.1.2. BMI (the most recent episode in 12 months)
     - 2.2.2. Lifestyle
       - 2.2.2.1. Smoking status (the most recent episode in 12 months)
     - 2.2.3. Clinical measurements
       - 2.2.3.1. Systolic BP (the most recent episode in 12 months)
       - 2.2.3.2. Diastolic BP (the most recent episode in 12 months)
       - 2.2.3.3. Total cholesterol (the most recent episode in 12 months)
       - 2.2.3.4. HDL-cholesterol (the most recent episode in 12 months)
       - 2.2.3.5. Creatinine (the most recent episode in 12 months)
       - 2.2.3.6. HbA1c (the most recent episode in 12 months)
   - **2.3. Diabetes complications**
     - 2.3.1. Retinopathy (the most recent episode in 12 months)
     - 2.3.2. End stage renal failure
     - 2.3.3. Foot ulcer
     - 2.3.4. Amputation
     - 2.3.5. Stroke
     - 2.3.6. Myocardial infarction
     - 2.3.7. Hypertension
3. **Health System**
   - **3.1. Structure (provider level)**
     - 3.1.1. Type of Provider
     - 3.1.2. Average diabetes population
   - **3.2. Structural quality**
     - 3.2.1. Hospital beds per 100,000 population
     - 3.2.2. Physicians employed per 100,000 population
   - **3.3. Processes (individual level)**
     - 3.3.1. Foot examination
       - 3.3.1.1. Foot Examination Done
     - 3.3.2. Eye examination
       - 3.3.2.1. Eye Examination Done
     - 3.3.3. Measurements done
       - 3.3.3.1. BP (at least one measurement in 12 months)
       - 3.3.3.2. Lipids (at least one measurement in 12 months)
       - 3.3.3.3. Microalbumin (at least one measurement in 12 months)
       - 3.3.3.4. HbA1c (at least one measurement in 12 months)
     - 3.3.4. Treatment
       - 3.3.4.1. Antihypertensive Medication (at least one medication in 12 months)
       - 3.3.4.2. Lipid Lowering Treatment (at least one medication in 12 months)
       - 3.3.4.3. ASA Medication (at least one medication in 12 months)
       - 3.3.4.4. Glucose Lowering Treatment
         - 3.3.4.4.1. Glucose Lowering: Diet Only
         - 3.3.4.4.2. Glucose Lowering: Tablets Only
         - 3.3.4.4.3. Glucose Lowering: Insulin Only
         - 3.3.4.4.4. Glucose Lowering: Insulin and Tablets

3.3.4.4.5. Glucose Lowering: Insulin Pump
3.3.5. Management
3.3.5.1. Self monitoring
3.3.5.2. Visit Frequency

4. Population

4.1.1. Total Population
4.1.2. Life expectancy
4.1.3. Mortality Data

5. Risk Adjusted Indicators

5.1. Epidemiology

5.1.1. Prevalence of diabetes mellitus per 1,000
5.1.2. Age at diagnosis by 10 year age bands

5.2. Process Quality

5.2.1. % subjects with one or more HbA1c tests during the last 12 months
5.2.2. % subjects with at least one test for microalbuminuria during the last 12 months
5.2.3. % subjects who received a dilated eye examination or evaluation of retinal photography by a trained caregiver within the last 12 months
5.2.4. % subjects receiving at least one foot examination within the last 12 months
5.2.5. % subjects with smoking status recorded within the last 12 months
5.2.6. % subjects with serum creatinine tested in last 12 months
5.2.7. % subjects with one or more blood pressure measurements within the last 12 months
5.2.8. Percentage of patients with hypertension who receive antihypertensive medication
5.2.9. Oral Therapy
5.2.9.1. Type of oral therapy (distribution of agents): Sulphonylureas
5.2.9.2. Type of oral therapy (distribution of agents): Biguanides
5.2.9.3. Type of oral therapy (distribution of agents): Glucosidase
5.2.9.4. Type of oral therapy (distribution of agents): Glitazones
5.2.9.5. Type of oral therapy (distribution of agents): Glinides
5.2.10. % subjects treated with insulin
5.2.11. % subjects treated with insulin in combination with OADs
5.2.12. % subjects treated with insulin with insulin pump therapy
5.2.13. % subjects with anti hypertensive treatment
5.2.14. % subjects with lipid lowering treatment
5.2.15. % subjects with ASA treatment
5.2.16. % subjects performing selfmonitoring of blood glucose/ urine testing

5.3. Outcome quality - intermediate outcomes

5.3.1. % subjects with most recent HbA1c level greater than 9.0 pct (poor control)
5.3.2. % subjects with most recent HbA1c level greater than 7,5 pct
5.3.3. % subjects with most recent blood pressure less than 140/90 mmHg
5.3.4. % subjects with BMI greater than 30
5.3.5. % subjects with normal value of microalbuminuria in last 12 months (among those tested)
5.3.6. % subjects currently smoking
5.3.7. Former or current foot ulceration

5.4. Outcome quality - terminal outcomes

5.4.1. Annual incidence of dialysis and/or transplantation (renal replacement therapy)
5.4.2. % subjects with end stage renal failure
5.4.3. Annual death rate per 100,000 populations in the general population from all causes, adjusted for standard European population. Annual death rate per 100,000 populations in persons, who have as primary or secondary cause of death, diabetes mellitus, adjusted for standard European population

Appendix B. BIRO dataset Quick Reference

Merge Table

Reference	Field Name	Parameter	Data Type	Categories / Ranges
BIRO001	PAT_ID	Patient ID	String(12)	
BIRO002	DS_ID	Data Source ID	Enumerated	1 = DARTS Dataset, Tayside, Scotland 2 = Umbria Dataset, Italy 3 = FQSD, Austria 4 = Telemed, Romania 5 = Noklus, Norway 6 = Diabetes Register, Cyprus 7 = CDM Program, Malta 8 = NEPI foundation, Sweden 9 = University of Debrecen, Hungary 10 = Scientific Inst. Public Health, Belgium 11 = Adelaide and Meath Hospital, Ireland 12 = CBO, Netherlands 13 = University of Ljubljana, Slovenia 14 = C.Hosp. Luxembourg, Luxembourg 15 = IMABIS Foundation, Spain 16 = Medical University of Silesia, Poland 17 = Havelhohe Hospital, Germany 18 = Hillerod University Hospital, Denmark 19 = Vuk Vrhovac University Clinic, Croatia 20 = DASMAN Centre , Kuwait 21 = IDF, Belgium 22 = Serectrix, Italy
BIRO003	TYPE_DM	Type Of Diabetes	Enumerated	1 = Type 1 2 = Type 2 3 = Other Types of Diabetes
BIRO004	SEX	Sex	Enumerated	1 = Male 2 = Female
BIRO005	DOB	Date of Birth	Date/Time	Range: ≥01/01/1900 - <Current Date
BIRO006	DT_DIAG	Date of Diagnosis	Date/Time	Range: ≥DOB - <Current Date
BIRO007	EPI_DATE	Episode Date	Date/Time	Range: ≥DOB - <Current Date
BIRO008	SMOK_STAT	Smoking Status	Enumerated	1 = Current Smoker 2 = Non-Smoker 3 = Ex-Smoker
BIRO009	CIGS_DAY	Cigarettes per day	Integer	Range: 0 – 100
BIRO010	ALCOHOL	Alcohol Intake	Integer	Range: 0 – 60
BIRO011	WEIGHT	Weight	Real	Range: 5 – 300
BIRO012	HEIGHT	Height	Real	Range: 0.3 – 3
BIRO013	BMI	Body Mass Index	Real	Range: 0.01 – 100
BIRO014	SBP	Systolic Blood Pressure	Integer	Range: 10 – 400
BIRO015	DBP	Diastolic Blood Pressure	Integer	Range: 10 – 300
BIRO016	HBA1C	HbA1c	Real	Range: 2.15 – 25.02
BIRO017	CREAT	Creatinine	Integer	Range: 3 – 1999
BIRO018	MA_TEST	Microalbumin	Enumerated	1 = MA Test Normal 2 = MA Test Abnormal 0 = No MA Test Recorded
BIRO019	CHOL	Total Cholesterol	Real	Range: 0.01 – 50
BIRO020	HDL	HDL	Real	Range: 0.01 – 5
BIRO021	TG	Triglycerides	Real	Range: 0.01 – 100
BIRO022	RETINAL_EXAM	Retinal	Enumerated	1 = Yes

Reference	Field Name	Parameter	Data Type	Categories / Ranges
		Examination		0 = No
BIRO023	RETINA	Retinopathy Status	Enumerated	1 = No Retinopathy 2 = Background Retinopathy 3 = Referable Retinopathy
BIRO024	MACULA	Maculopathy Status	Enumerated	1 = No Maculopathy 2 = Referable Maculopathy
BIRO025	FOOT_EXAM	Foot Examination	Enumerated	1 = Yes 0 = No
BIRO026	PULSES	Foot Pulses	Enumerated	1 = Present 0 = Absent
BIRO027	FTSENS	Foot Sensation	Enumerated	1 = Normal 0 = Abnormal
BIRO028	ESRF	End Stage Renal Therapy	Enumerated	1 = Yes 0 = No
BIRO029	DIALYSIS	Renal Dialysis	Enumerated	1 = Yes 0 = No
BIRO030	TRANSPLANT	Renal Transplant	Enumerated	1 = Yes 0 = No
BIRO031	STROKE	Stroke	Enumerated	1 = Yes 0 = No
BIRO032	ULCER	Active Foot Ulcer	Enumerated	1 = Yes 0 = No
BIRO033	MI	Myocardial Infarction	Enumerated	1 = Yes 0 = No
BIRO034	LASER	Laser	Enumerated	1 = Yes 0 = No
BIRO035	HYPERTENSION	Hypertension	Enumerated	1 = Yes 0 = No
BIRO036	BLIND	Blindness	Enumerated	1 = Yes 0 = No
BIRO037	AMPUT	Amputation	Enumerated	1 = Yes 0 = No
BIRO038	HYPERT_MED	Antihypertensive Medication	Enumerated	1 = Yes 0 = No
BIRO039	DRUG_THERAPY	Hypoglycaemic Drug Therapy	Enumerated	1 = Insulin Only 2 = Tablet Only 3 = Insulin and Tablets 4 = None (Diet Only)
BIRO040	ORAL_THERAPY	Oral Drug Therapy	Enumerated	1 = Sulphonylureas 2 = Biguanides 3 = Glucosidase Inhibitors 4 = Glitazones 5 = Glinides
BIRO041	PUMP_THERAPY	Pump Therapy	Enumerated	1 = Yes 0 = No
BIRO042	NASAL_THERAPY	Nasal Therapy	Enumerated	1 = Yes 0 = No
BIRO043	INJECTIONS	Average Injections	Real	Range: 0 – 20
BIRO044	SELF_MON	Self Monitoring	Enumerated	1 = Urine 2 = Blood Glucose 3 = Both
BIRO045	EDUCATION	Diabetes Specific Education	Enumerated	1 = Yes 0 = No
BIRO046	LDL	LDL	Real	Range: 0.01 – 15
BIRO047	ALC_STAT	Alcohol Status	Enumerated	1 = Current Drinker 2 = Non-Drinker 3 = Ex-Drinker
BIRO048	DMP_ENROL	Patient Enrolment in DMP for Diabetes	Enumerated	1 = Yes 0 = No

Reference	Field Name	Parameter	Data Type	Categories / Ranges
BIRO053	LIPID_THERAPY	Lipid Lowering Therapy	Enumerated	1 = Yes 0 = No
BIRO054	ANTIPLATELET_THERAPY	Anti-platelet Therapy	Enumerated	1 = Yes 0 = No
BIRO055	SULPHONYLUREAS	Sulphonylurea Therapy	Enumerated	1 = Yes 0 = No
BIRO056	BIGUANIDES	Biguanide Therapy	Enumerated	1 = Yes 0 = No
BIRO057	GLUCOSIDASE_INHIBITORS	Glucoseidase Inhibitor Therapy	Enumerated	1 = Yes 0 = No
BIRO058	GLITAZONES	Glitazone Therapy	Enumerated	1 = Yes 0 = No
BIRO059	GLINIDES	Glinide Therapy	Enumerated	1 = Yes 0 = No
BIRO099	SUB_DS_ID	Sub data source identifier (Related to DS_ID)	Enumerated	TBC

Activity Table

Reference	BIRO Name	Parameter	Data Type	Enumerated Values
BIRO001	PAT_ID	Patient ID	String(12)	
BIRO049	AD_START_DATE	Data of commencement of period of patient activity	Date/Time	Range: ≥DOB - <Current Date
BIRO050	AD_START_REASON	Reason for the commencement of activity period	Enumerated	1 = Birth 2 = Diabetes Diagnosis 3 = Transferred In
BIRO051	AD_END_DATE	Data of completion of period of activity	Date/Time	Range: ≥DOB - <Current Date
BIRO052	AD_END_REASON	Reason for the completion of activity period	Enumerated	1 = Death 2 = Transferred Out 3 = Lost to Follow-up

Population Table

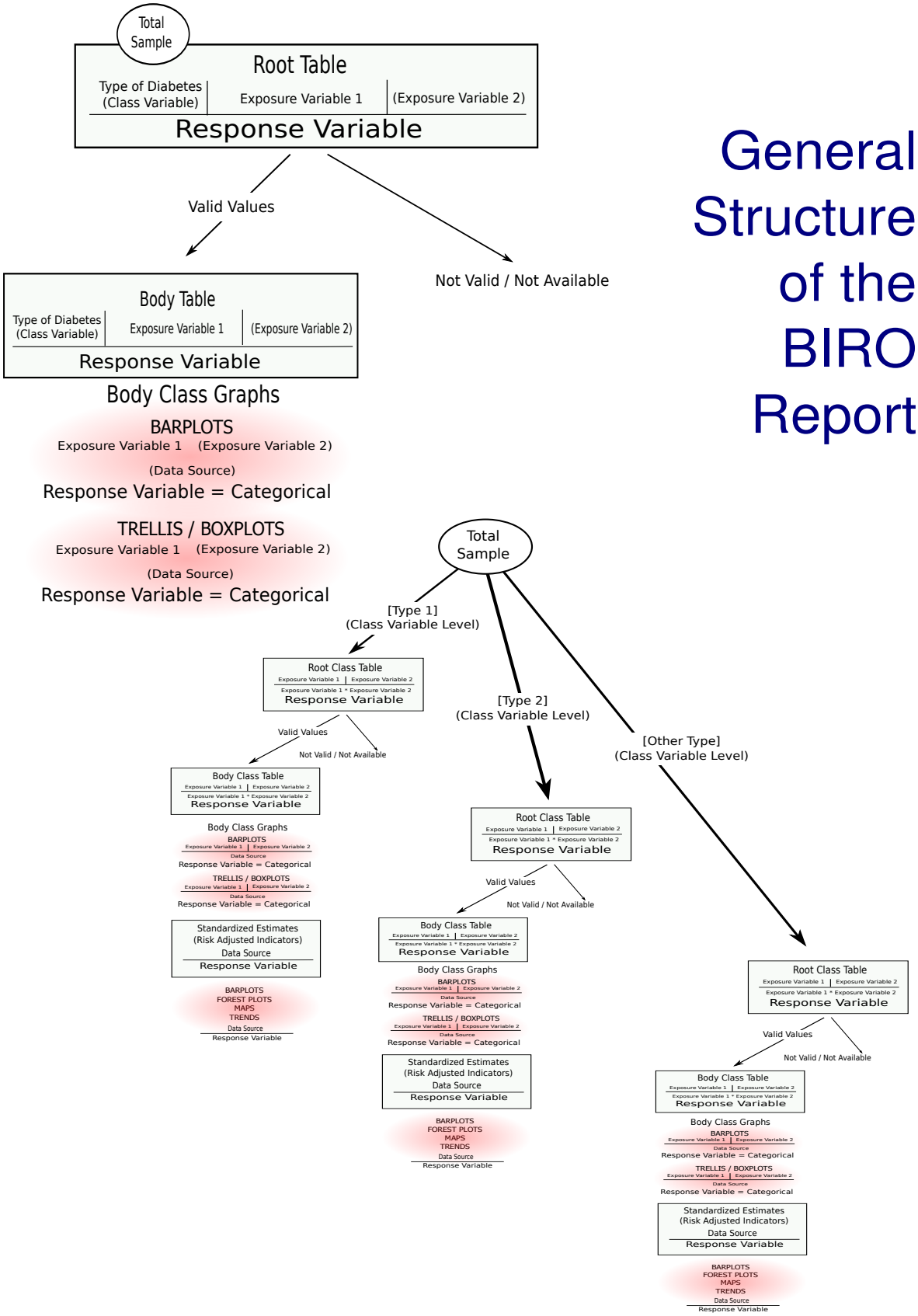
Reference	BIRO Name	Parameter	Data Type	Enumerated Values
BIRO002	DS_ID	Data Source ID	Enumerated	
BIRO099	SUB_DS_ID	Sub Data Source ID	Enumerated	
BIRO300	YEAR	Year	Date/Year	
BIRO310	AGEBAND	Age band	Enumerated	1=0,14 2=15,24 3=25,34 4=35,44 5=45,54 6=55,64 7=65,74 8=75,84 9=85+
BIRO301	POP_M	Total Male Population	Integer	
BIRO302	POP_F	Total Female Population	Integer	
BIRO303	DEATHS_M	Total Deaths in Male Population	Integer	
BIRO304	DEATHS_F	Total Deaths in Female Population	Integer	

Diabetic population table

Reference	BIRO Name	Parameter	Data Type	Enumerated Values
BIRO002	DS_ID	Data Source ID	Enumerated	See Appendix 2
BIRO099	SUB_DS_ID	Sub Data Source ID	Enumerated	
BIRO300	YEAR	Year	Date/Year	
BIRO310	AGEBAND	Age band	Enumerated	1=0,14 2=15,24 3=25,34 4=35,44 5=45,54 6=55,64 7=65,74 8=75,84 9=85+
BIRO003	TYPE_DM	Type of Diabetes	Enumerated	1=Type 1 2=Type 2 3=Other
BIRO305	POP_D_M	Total Diabetic in Male Population	Integer	
BIRO306	POP_D_F	Total Diabetic in Female Population	Integer	

Appendix C. Statistical Engine Help Pages

General Structure of the BIRO Report



HOW TO READ THE BIRO REPORT
INTERPRETING ONE WAY TABLES

ONE WAY TABLES are used to tabulate the frequency of values for a target variable/indicator against a single exposure

Code and Description of the Indicator

Target Variable

2.2.3.1 Systolic BP (the most recent episode in 12 months)

Root Table
includes all observations in the input dataset

Response Variable

Exposure Variable

Valid Value

Not Valid / Not Available

SBP	Gender			
	Valid Value (%)	NV/NA (%)		N (%)
Valid Value	254 (79.9)	0(0.0)		254 (79.9)
NV/NA	64 (20.1)	0(0.0)		64 (20.1)
TOTAL	318(100.0)	0(0.0)		318 (100.0)

Table 2.2.3.1.3 - Missing Data: SBP * Gender

Within Table and Marginal Percentages are expressed as Column Percentages

Row Percentages

GRAND TOTAL for all tables: total number of observatoions in the overall sample

Caption: numbering is the same used for HTML and CSV outputs

Only Valid Values for all variables enter the Body Table

Column percentages allow to spot differences in the distribution of the Exposure Variable for each level of the response variable (ex: high levels of SBP are more frequent among females). The Relative Risk (RR) for a specific class can be computed as [Col(1)%]/[Col(2)%]. Here Males/Females[160+]=13.6/17=0.8, i.e. Females have a 20% increased risk of falling in the upper level of SBP

SBP	Gender			
	Male (%)	Female (%)		N (%)
[0 - 130)	36 (23.4)	25(25.0)		61 (24.0)
[130 - 160)	97 (63.0)	58(58.0)		155 (61.0)
[160+)	21 (13.6)	17(17.0)		38 (15.0)
TOTAL	154(60.6)	100(39.4)		254 (100.0)

Table 2.2.3.1.4 - SBP * Gender

Row percentages highlight the weight of each level of the exposure variable on the total sample

Body Table
includes only Valid observations for all exposure and response variables

	CMH Chi-Square	p.value	df
Value	0.7721	0.6797	2

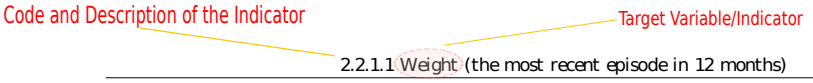
Chi-Square Table
Refers to the Body Table

The Chi-Square test provides a quick measure of the strength of the association between one or more exposures (stratified analysis) and the response of interest.
A p value < 0.05 is computed using the CMH Value together with the associated degrees of freedoms (df). It allows to reject the hypothesis of independence between the columns and the rows of the body table (ex.: the differences between males and female found in the body table do not support an association between gender and systolic blood pressure).

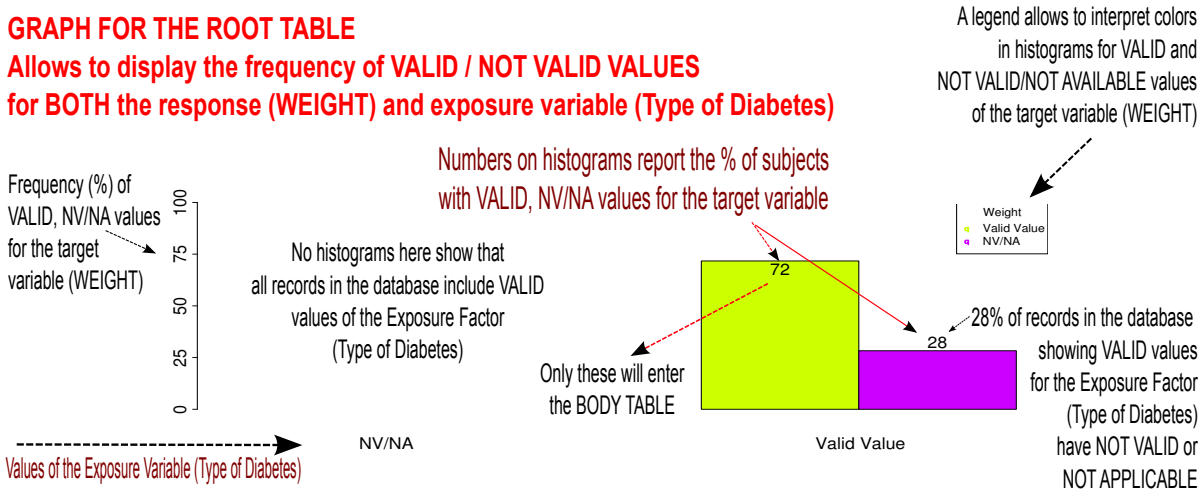
HOW TO READ THE BIRO REPORT

INTERPRETING GRAPHS FOR ONE WAY TABLES

GRAPHS FOR ONE WAY TABLES are produced to provide a graphical display of the content of ONE WAY TABLES

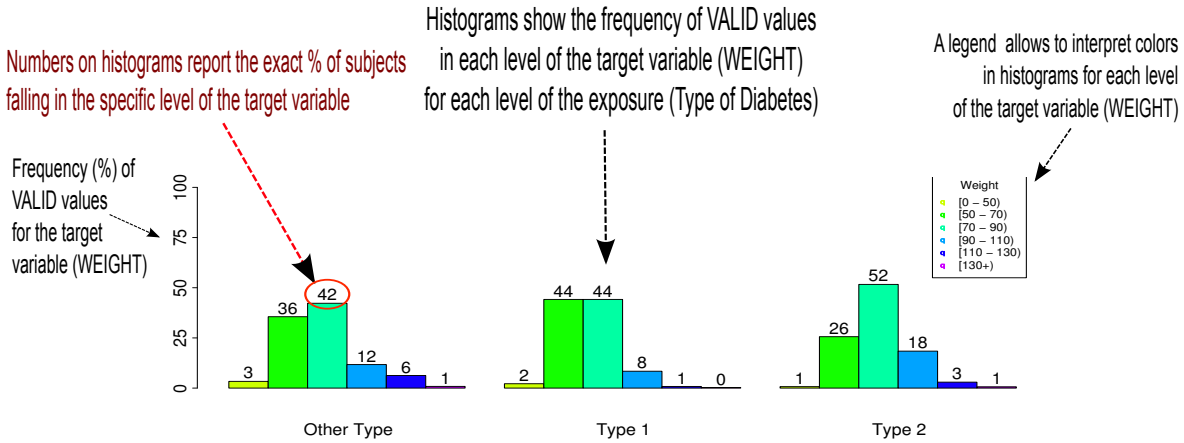


GRAPH FOR THE ROOT TABLE
Allows to display the frequency of VALID / NOT VALID VALUES for BOTH the response (WEIGHT) and exposure variable (Type of Diabetes)



Barplots: 2.2.1.1.1 - Missing Data: Weight * Type of Diabetes

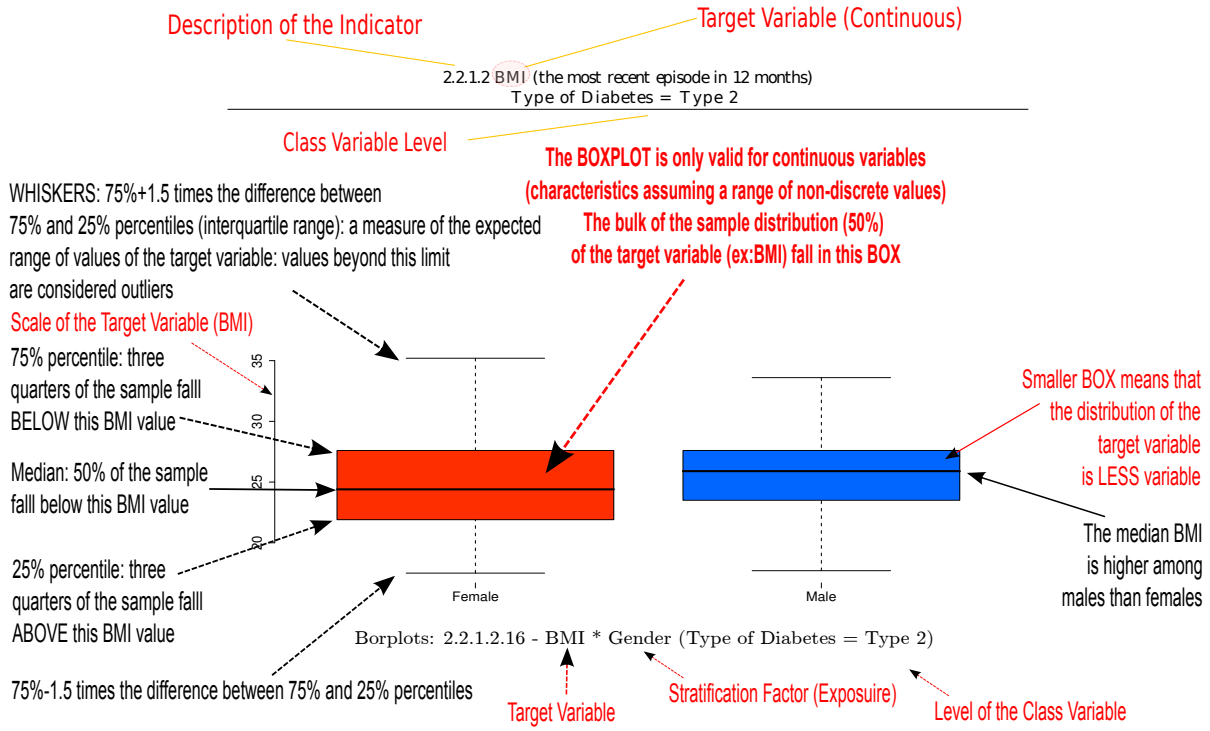
GRAPH FOR THE BODY TABLE
Allows to display the frequency of values for the response variable (WEIGHT) by levels of any exposure variable (Type of Diabetes)



Barplots: 2.2.1.1.2 - Weight * Type of Diabetes

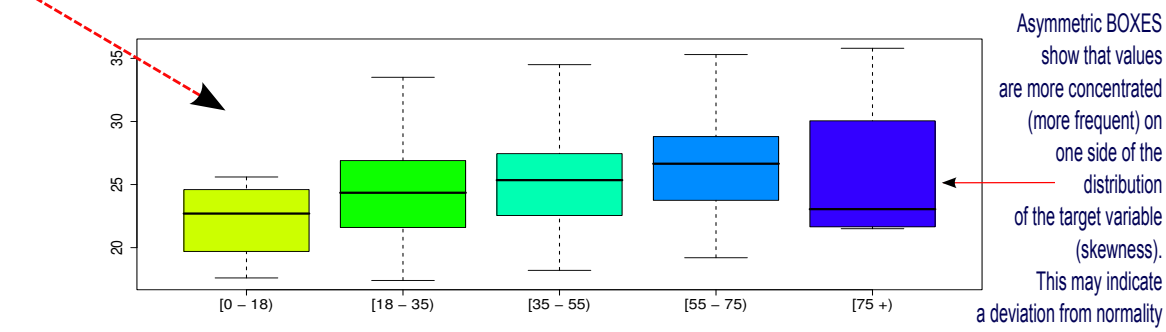
HOW TO READ THE BIRO REPORT
BOXPLOTS

BOXPLOTS are used to provide an effective graphical display of the distribution of a continuous variable



Borplots: 2.2.1.2.16 - BMI * Gender (Type of Diabetes = Type 2)

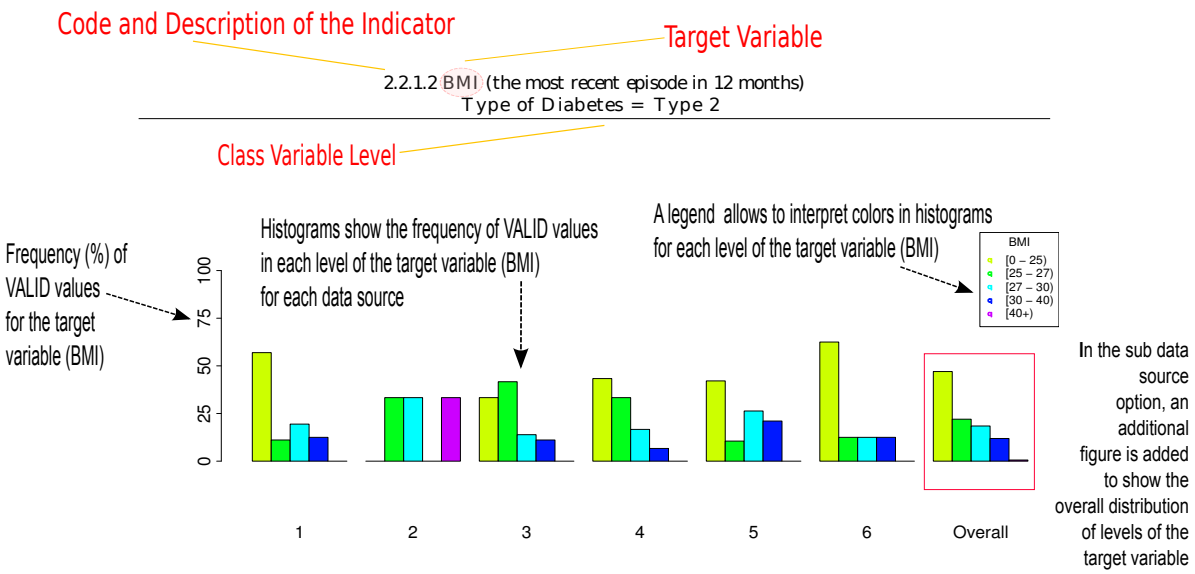
Stratification Factors with many levels (Age Classes) allow exploring direct associations
between increasing levels of the exposure, and values of the target variable.
Here, with the exception of those aged 75+, BMI linearly increases with AGE. Whiskers show that also the variation of BMI is linearly increasing with AGE.



Borplots: 2.2.1.2.17 - BMI * Age (Type of Diabetes = Type 2)

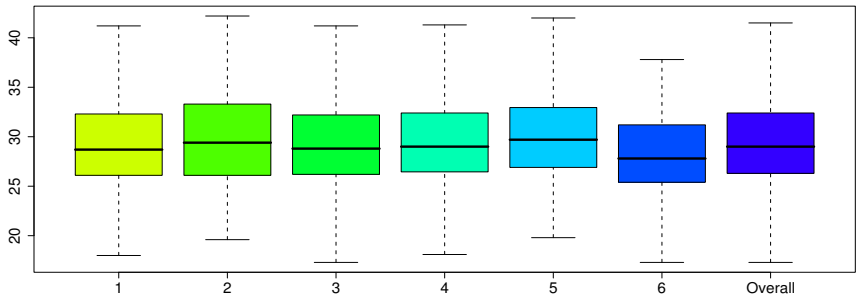
HOW TO READ THE BIRO REPORT
USING THE SUB DATA SOURCE OPTION

The SUB DATA SOURCE option can be used
to compare the distribution of a target response across centres



Barplots: 2.2.1.2.26 - BMI by data source (Age = [35 - 55), Type of Diabetes = Type 2)

In the SUB DATA SOURCE output, BOXPLOTS are used
to compare distributions across centres for continuous response variables



Boxplots: 2.2.1.2 3 - BMI by data source (Type of Diabetes = Type 2)

HOW TO READ THE BIRO REPORT
INTERPRETING TWO WAY TABLES

TWO WAY TABLES are used to tabulate the frequency of values for a target variable/indicator against two exposures

Code and Description of the Indicator

Target Variable/Indicator

5.2.1 % subjects with 1+ HbA1c tests during the last 12 months

Type of Diabetes = Type 2

Class Variable Level

Root Table

Two Way by Class Variable

Response Variable

BOTH values of Exposure Variables are Valid

At least one of the Exposure Variables is Not Valid / Not Available

Exposure Variable 1 * Exposure Variable 2

Valid Value

HbA1c done	Age*Gender				
	Valid Value		NV/NA		
	Valid Value (%)	NV/NA (%)	Valid Value (%)	NV/NA (%)	N (%)
Valid Value	8707 (100.0)	0 (0.0)	0 (0.0)	0(0.0)	8707 (100.0)
NV/NA	0 (0.0)	0 (0.0)	0 (0.0)	0(0.0)	0 (0.0)
TOTAL	8707(100.0)	0(0.0)	0(0.0)	0(0.0)	8707 (100.0)

Table 5.2.1.17 - Missing Data: HbA1c done * Age * Gender (Type of Diabetes = Type 2)

Within Table and Marginal Percentages are expressed as Column Percentages

Row Percentages

GRAND TOTAL for all tables: total number of observations in the overall sample

Caption: numbering is the same used for HTML and CSV outputs

HbA1c done	Age*Gender										
	Male					Female					
	[0 - 18) (%)	[18 - 35) (%)	[35 - 55) (%)	[55 - 75) (%)	[75 +) (%)	[0 - 18) (%)	[18 - 35) (%)	[35 - 55) (%)	[55 - 75) (%)	[75 +) (%)	N (%)
at least one test	0 (0.0)	13 (86.7)	542 (91.4)	2914 (95.3)	962 (92.9)	0 (0.0)	17 (94.4)	339 (89.4)	2230 (93.5)	1159(94.8)	8176 (93.9)
no test	0 (0.0)	2 (13.3)	51 (8.6)	143 (4.7)	74 (7.1)	1 (100.0)	1 (5.6)	40 (10.6)	155 (6.5)	64(5.2)	531 (6.1)
TOTAL	0(0.0)	15(0.2)	593(6.8)	3057(35.1)	1036(11.9)	1(0.0)	18(0.2)	379(4.4)	2385(27.4)	1223(14.0)	8707 (100.0)

Table 5.2.1.18 - HbA1c done * Age * Gender (Type of Diabetes = Type 2)

Body Table

Two Way by Class Variable

Row percentages highlight the weight of each level of the exposure variable on the total sample

Column percentages with two exposures allow to spot differences in terms of relative risk among an exposure and the response of interest for each level of a second exposure (ex: among subjects aged 55-75, the relative risk of not having an HbA1c test done, compared to females is RR=4.7/6.5=0.72)

CMH Chi-Square	
Value	One or more cells have 0 obs

Chi-Square Table

Refers to the Body Table

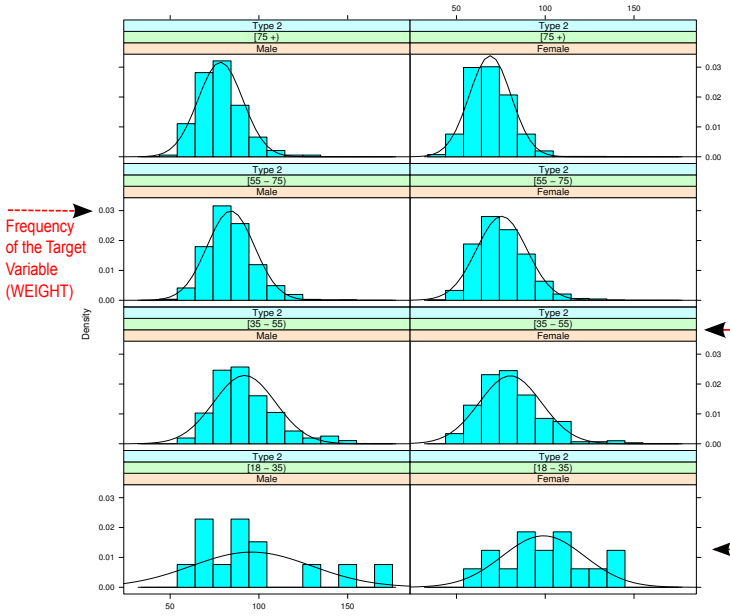
The Chi-Square test cannot be computed when one or more cells have less than 5 observations. This is frequently the case in two-way tables.

HOW TO READ THE BIRO REPORT

TRELLIS GRAPHS FROM TWO WAY TABLES

TRELLIS GRAPHS offer a flexible graphical representation of the distribution of a target CONTINUOUS RESPONSE VARIABLE for different levels of EXPOSURE FACTORS and CLASS VARIABLES

2.2.1.1 Weight (the most recent episode in 12 months)
Type of Diabetes = Type 2



Trellis density plot: 2.2.1.1.2 - Weight * Gender * Age (Type of Diabetes = Type 2)

The density plot explores the distribution of target continuous variable: WEIGHT, WITHIN the Level of the Class Variable "Type of Diabetes"=2

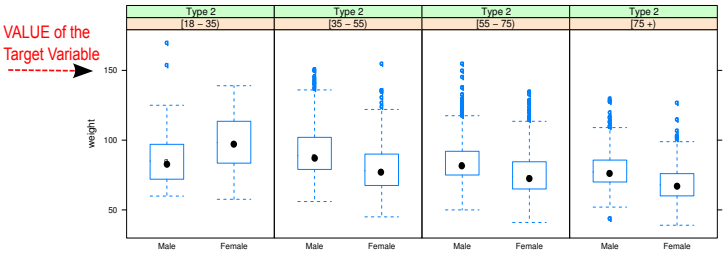
Frequency of the Target Variable (WEIGHT)

Each section of the TRELLIS displays the distribution of the target variable (WEIGHT) for a particular combination of exposure factors (age, gender).

Lines can be used to compare the distribution of one exposure (gender) within the same level of the other exposure (age).

The continuous variable (WEIGHT) is automatically divided in a number of classes of equal range to display the frequency distribution. A curve is superimposed to show the shape of the density and explore the level of heterogeneity among exposure classes.

Here, the shape of the distribution of WEIGHT among young subjects shows a wider variation than higher ages, particularly for males. The distribution in older subjects is fairly normal.



Boxplots: 2.2.1.1.21 - Weight * Gender * Age (Type of Diabetes = Type 2)

VALUE of the Target Variable

BOXPLOTS offer a synthetic view of the distribution of values (see BOXPLOT help for an explanation of the graph).

Here, outlying values outside whiskers are highlighted by dots, showing a higher presence of extreme WEIGHT values in the central classes of age.

The median value decreases with age.

The median WEIGHT of males is constantly higher than females, except for younger subjects, for which holds the opposite.

HOW TO READ THE BIRO REPORT
RISK-ADJUSTED INDICATORS

RISK ADJUSTED INDICATORS include the estimation of expected values and adjusted rates based on multivariate modelling (logistic regression)

Description of the Indicator

% subjects with most recent HbA1c > 7,5 pct
Type of Diabetes = Type 2

Table of Standardized Results

Two Way by Class Variable

Class Variable Level

Sub Data Source

Expected no. Outcomes (based on Logistic Regression Model)

Crude Rate [O/N]

AR 95% CI based on AR Variance Estimation

Measures an Excess or Reduction of Observed Outcomes in a specific centre compared to the Expected % estimated using a reference logistic regression model (internal or external to the local population)

Observed no. Outcomes (Numerator)

Total no. Subjects (Denominator)

Adjusted Rate PR * [CR / (E/N)]

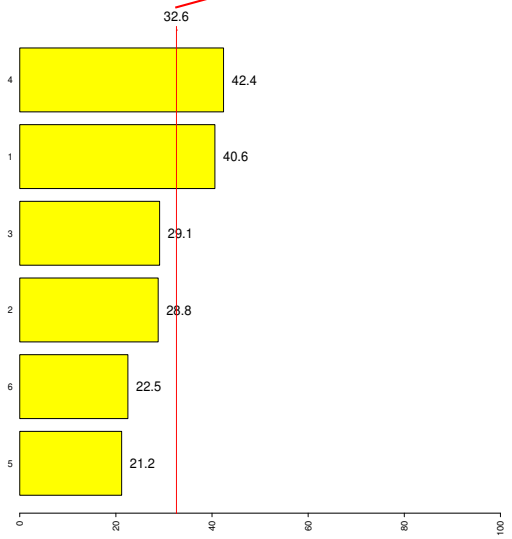
Ranking

	s	O	E	N	CR	AR	95% C.I.	[O-E]/E %	95% C.I. [O-E]/E
1	4	429	330	1020	42.1	42.4	(39.5; 45.3)	30.0	(21.1; 38.9)
2	1	957	768	2357	40.6	40.6	(38.8; 42.5)	24.6	(18.8; 30.4)
3	3	734	824	2530	29.0	29.1	(27.2; 30.9)	-10.9	(-16.5; -5.3)
4	2	228	258	791	28.8	28.8	(25.6; 32.1)	-11.6	(-21.6; -1.7)
5	6	67	97	296	22.6	22.5	(17.2; 27.8)	-30.9	(-47.1;-14.8)
6	5	252	388	1182	21.3	21.2	(18.5; 23.8)	-35.1	(-43.1;-27.0)
	T	2667		8176	32.6				

95% C.I. of O-E/E%
If the interval DOES NOT CROSS ZERO any excess/reduction IS statistically significant at alpha=0.05

Total no. Subjects

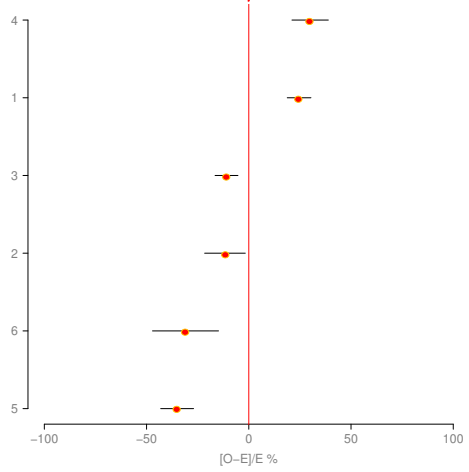
Population Rate (PR)



Barplots: 5.3.2.29 - Adjusted Rates % subjects with most recent HbA1c > 7,5 pct

Graphical Representation of Standardized Rates

Centres are ordered by Descending Adjusted Rates



Forest plots: 5.3.2.1 - % subjects with most recent HbA1c > 7,5 pct

Graphical Representation of O-E/E% with 95% Confidence Intervals

Statistically Significant Excess/Reductions are highlighted by lines not intersecting the zero line