

1. BellaDati User's Documentation	2
1.1 Getting Started	2
1.1.1 Logging in to BellaDati	3
1.1.2 Exploring BellaDati Workspace	4
1.2 BellaDati Concepts	6
1.2.1 Detailed Glossary	6
1.2.2 Permissions and Roles	9
1.3 Data Sets	10
1.3.1 Creating Data Set	12
1.3.2 Importing Data	13
1.3.2.1 Importing from Clipboard	14
1.3.2.2 Importing from File	15
1.3.2.3 Import Settings	17
1.3.2.3.1 Import Results	21
1.3.3 Browsing Data	22
1.3.4 Managing Indicators	23
1.3.4.1 Grouping Indicators	26
1.3.4.2 Translating Indicators	26
1.3.4.3 Transforming Indicator into Attribute	27
1.3.4.4 Formula Reference Guide	28
1.3.5 Managing Attributes	39
1.3.5.1 Defining Drill-down Path	41
1.3.5.2 Members Appearance	42
1.3.5.3 Translating Attributes and Members	43
1.3.5.4 Transforming Attribute into Indicator	44
1.3.6 Joining Data Sets	45
1.3.6.1 Changing join point	47
1.3.7 Structure Backup	47
1.3.8 Sharing Data Sets	49
1.3.9 Watching Data Changes	50
1.4 Data Sources	51
1.4.1 Connecting to Database	53
1.4.2 Connecting to Google	54
1.4.3 Connecting to SalesForce	56
1.4.4 Connecting to Amiando	57
1.4.5 Connecting to FaceBook	58
1.4.6 Connecting to URL	60
1.4.7 Connecting to YouTube	63
1.4.8 Scheduling Import	65
1.5 Reports	66
1.5.1 2.3.1 Report management	66
1.5.2 2.3.2 Table settings	69
1.5.3 2.3.3 Chart settings	77
1.5.4 2.3.4 Indicator settings	84
1.5.5 2.3.5 Time settings	91
1.5.6 2.3.6 Analytical filters	94
1.5.7 2.3.7 Drill-down path	96
1.5.8 2.3.8 Cooperation and sharing	99
1.5.9 2.3.9 Custom content and attachments	103
1.5.10 2.3.10 Exports	105
1.5.11 2.3.11 Formulas	106
1.5.11.1 Report Variables	116
1.6 Dashboards	118
1.6.1 2.4.1 Insertion of views	118
1.6.2 2.4.2 Dashlet adjustment	120
1.6.3 2.4.3 Other dashlets	121
1.6.4 2.4.4 Desktop management	124
1.6.5 2.4.5 Publishing dashboards on Mercato	124
1.7 Administration	126
1.7.1 Administering Users	126
1.7.2 Administering User Groups	128
1.7.3 Managing User Profile	129
1.7.4 Domain Overview and Administration	131

# BellaDati User's Documentation

## Welcome

The BellaDati User's Guide begins with an overview of the key features of BellaDati, and goes on to describe the use of BellaDati as a smart reporting, analytical and a collaborative tool. If you still have a question that has not been answered, [please ask us](#).

## Video tutorials

### List of video-tutorials

1. How to import data

---

2. How to prepare data set

---

3. How to create report

---

3. How to create report

---

4. How to use workspace

---

5. How to manage users

---

6. Public dashboard

---

7. Exports

---

## Getting Started

---

The following pages contain information to help you get started using BellaDati:

- [Logging in to BellaDati](#)
  - [Exploring BellaDati Workspace](#)
- 

11. Conditional formating

12. Members aggregation

How to prepare data set

---

How to connect to external data source

### Video tutorial

Configure to Google Analytics

Configure to Google Docs

Configure databases

Configure path to external data source

Import data sets Google Analytics

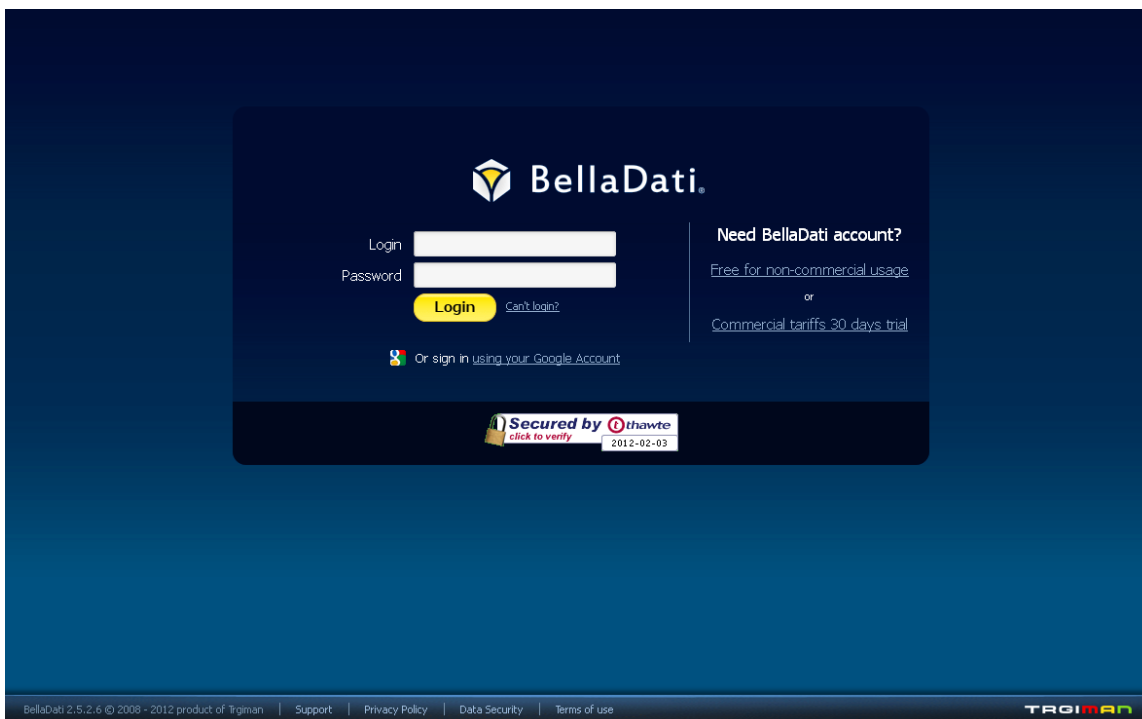
How to analyze the Google World Bank

## Logging in to BellaDati

Many BellaDati functions require correct permissions to be set, therefore anonymous access is not permitted and you are prompted to log in.

The Login panel will be displayed if you are not logged in to BellaDati. There are three things you can do here:

1. Log into BellaDati: To log in to BellaDati, enter your Username and Password and click the Log In button.  
Log in using Google Account: If you already have an account by Google, you can log in to BellaDati simply by clicking on corresponding link and entering login information for existing Google account. Note: If it is the first time you are logging to BellaDati using your Google account, you will be automatically registered to BellaDati.
2. Can't login?: Click this link in case you have lost your password or your BellaDati account has been locked (see login policy below).  
Select the appropriate option and enter your username (usually e-mail address) in the popup. Then click "Submit" button and now you should receive the instructions how to set your new password or unlock your account via e-mail in a few minutes. If you have forgotten your username, you will need to contact your BellaDati administrator for help.
3. Sign up for an account: If you do not have a user account yet, you can create your own user account by clicking one of the links on the left and follow registering procedure. If you would like to get access to existing domain, please contact your BellaDati administrator.





Security note: When you are using BellaDati Cloud, you can verify BellaDati site identity by clicking on the certificate logo here to prevent fraud of your password. Data transfer security is then ensured by HTTPS encryption.

## Login and Password Policy

- Password must be at least 8 characters long and must contain at least one uppercase [A-Z], one lowercase [a-z] and one numeric character [0-9].
- Language specific characters (like diacritic) are not supported.
- When the login fails three times, your BellaDati account becomes locked. An e-mail with verification link will be sent to your registered e-mail. Use this link to unlock your account.



Unlocking link will be valid for 24 hours only! Otherwise you need to create a new request on the BellaDati login page.



BellaDati On-Premise notes:

1. Maximum failed login attempts count can be set differently by domain administrator.
2. Authentication can be performed using LDAP or Active Directory services.

## Next Steps

- [Explore BellaDati workspace](#)
- [BellaDati basic concepts](#)

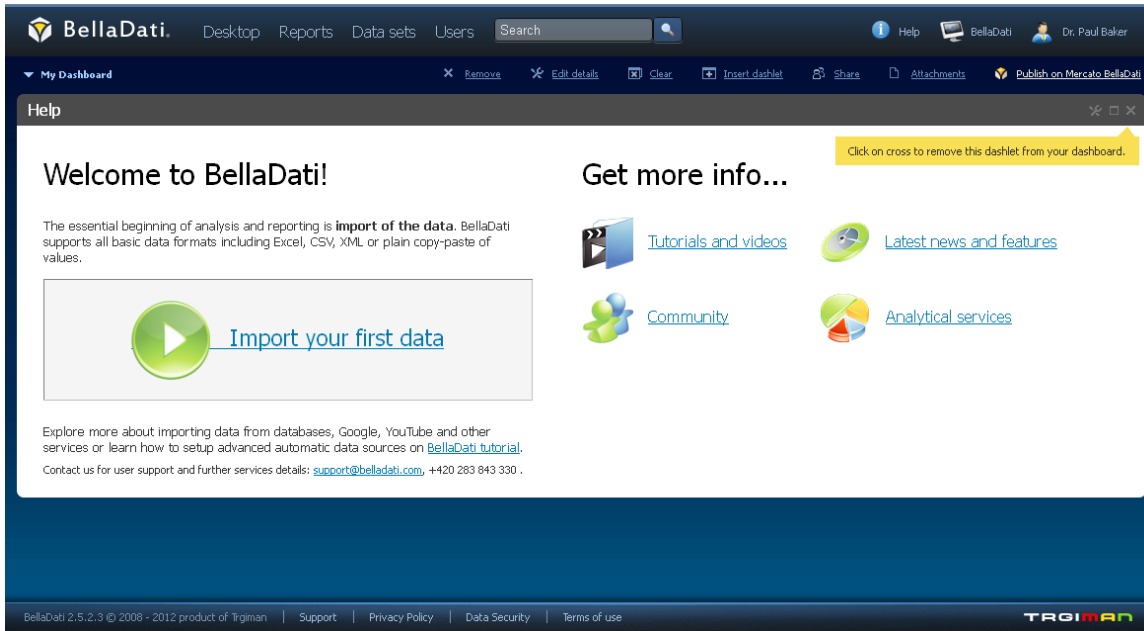
## Exploring BellaDati Workspace

The Dashboard is the first screen you see when you login to BellaDati.

- The menu (navigation bar) is the same on every screen in BellaDati. It contains links which give you quick access to many of BellaDati's most useful functions. Menu structure depends on the roles assigned to you by administrator.
- Main window contents change according to the context of your current work in BellaDati. Dashboard is the default page. An additional submenu is usually displayed on the left or top of the main window to allow executing context specific functions. This will be displayed in corresponding documentation sections.
- Footer provides supplemental function and is also the same on every screen in BellaDati.

On this page:

- [Main menu](#)
- [Footer](#)
- [Next Steps](#)



**i** Please note that your BellaDati screen may look slightly different from this screenshot when you are using BellaDati On-Premise.  
The default content of BellaDati page after login may also differ if you specify particular URL before login.

More detailed description of navigation in BellaDati is following:

## Main menu

Following functions are available for all BellaDati users if not stated otherwise:

- BellaDati icon - always navigates you to default BellaDati page (dashboards)
- Dashboards - navigates to [dashboards page](#)
- Reports - navigates to [reports page](#)
- Data sets - navigates to [data sets page](#) and [data source configuration](#); data manager and domain administrator only
- Users - navigates to [user](#) and [user groups administration](#); domain and system administrator only
- Domains - navigates to domain management; system administrator only
  
- Search field - allows you to search among [reports](#), [data sets](#), [indicators](#) and [attributes](#)
- Help - navigates you to context help (help opens in separate browser's window)
- DomainName - shows your domain details (usage statistics, validity, preferred language etc.); domain and system administrators only
- YourName, Logout - this link navigates you to your profile details, password change and safe logout from BellaDati; domain and system administrators can assign user roles or user groups through this

**i** System administrator role is not required for BellaDati Cloud usage. However this role could be useful for BellaDati On-Premise, especially for large enterprise companies.

## Footer

BellaDati footer displays:

- Installed BellaDati version
- Link to support (this documentation, bug reporting, video tutorials, blog, etc.)
- Privacy policy and data security conditions
- Terms of use (or EULA for BellaDati On-Premise)

## Next Steps

- [BellaDati Concepts](#) and [BellaDati Glossary](#)
- [Permissions and roles](#)
- [Importing data](#)
- [Creating report](#)
- [User administration](#)

## BellaDati Concepts

Key concepts used in BellaDati are described below. See [BellaDati glossary](#) for details.

### Data set

BellaDati has its own integrated data warehouse. This warehouse contains virtual databases that represents data with similar characteristics. These virtual databases are called data sets and reports are established on them. Each data set can be connected to one [data source](#).

Possible actions: [Importing data](#) | [Browsing data](#) | [Exporting data](#) | [Structure backup](#) | [Joining data sets](#) | [Watching data changes](#) | [Cleaning data](#) | [Sharing data](#) | [Defining attributes](#) | [Defining indicators](#) | [Creating and removing datasets](#)

### Data source

Imagine companies has already running their business. Such company has applications and systems installed or use services that generate a lot of data in underlying databases. Moreover employess produces Excel spreadsheets, text or CSV files of data. These databases and files are very suitable data sources for BellaDati analysis. Generally all third party systems BellaDati can import data from are called data sources.

Possible actions: [Connecting to database](#) | [Connecting to URL](#) | [Connecting to FaceBook](#) | [Connecting to Google Services](#) | [Connecting to CRM](#) | [Scheduling automatic upload](#)

### Report

Report is basically a set of tables and charts created mainly for detailed overview and analytical purposes on data stored in data sets. Each report can also contain custom content, comments, attachments and can be easily shared with other BellaDati users or published to corporate intranet or public. Permanent export of each report to PDF, Excel, PNG or Power Point is possible.

Possible actions: [Creating report](#) | [Creating table](#) | [Creating chart](#) | Editing report | [Copying report](#) | [Sharing report](#) | [Exporting report](#) | Commenting report | Attaching files | Publishing report | Searching reports | Filtering data

### Dashboard

Dashboard typically consists of the most important tables or charts from reports and is primarily determined for managers who need quick overview of actual company situation. Each dashboard can be customized by adding special content (dashlets). Dashboards can also be shared for public via web.

Possible actions: [Creating dashboard](#) | [Sharing dashboard](#) | Publishing dashboard | Editing dashboard


### Next Steps

- Explore the [Detailed glossary](#)
- [Permissions and roles](#)

## Detailed Glossary



Base concepts

---

Attribute	<p>Attributes are describing indicators, usually in a form of general terms such as "country", "department", "product", "employee", etc. Attributes consist of members (instances) eg. "Czech Republic", "Sales Department", "Product XY", "John Smith".</p> <p>For example, a Sales record could have attributes such as sales person, store, region, date, etc.</p>
Dashboard	<p>Dashboard typically consists of the most important tables or charts from reports and is primarily determined for managers who need quick overview of actual company situation. Special custom content can be added to every dashboard as a customization. Dashboards can also be shared for public via web.</p>
Data set	<p>A virtual database of BellaDati's integrated data warehouse. Each data set usually represents data with similar characteristics (e.g. invoices, wages, costs) from one data source. Data set consists of data set indicators and attributes. Each report is build on single data set. More data sets can also be joined together.</p>
Data set indicator	<p>Data set indicators are defined within the data set and are available as musters for report indicators. Another settings (like aggregations, appearance, etc.) are not supported. Simply said, the data set indicators represents a raw numerical value (in the OLAP language it is a fact) with basic attributes - name, unit and rounding mode. Values of these indicators are straightforwardly stored in BellaDati's data warehouse directly from imported data.</p>
Member	<p>"Instance" of one attribute. For example, members of the "employee" attribute may be "Jan Novak", "John Smith" etc.</p> 
Record	<p>Particular data row stored in the data set. Each record consists of indicator's value, members and date/time information and has the same structure within single data set</p>
Report	<p>An analytical output, providing answers to user's research according to the gathered data. Each report can contain description, tags and consists of one or more views (charts and pivot tables), comments and possibly attachments.</p>
Report indicator	<p>Report indicators are created in the report from the data set indicators or ad hoc. Unlike the data set indicators, report indicators are supporting wide range of various settings - <a href="#">member aggregation</a>, <a href="#">time aggregation</a>, <a href="#">appearance</a>, <a href="#">conditional formatting</a>, extended <a href="#">formula support</a> with <a href="#">Report Variables</a> etc.</p>

### Detailed glossary

Term	Meaning
Alarm (data watch)	Notifies data manager about changes in value of indicators in a particular data set according fulfillment one or more predefined conditions.
Chart	Graphic representation of analytical data view. There are a lot of chart types in BellaDati - eg. line chart, pie chart, bar chart, stack bar chart, horizontal bar chart, radar chart, horizontal heat map. Each of them offers a different way how to explore particular dimensions (time, indicators and attributes).
Comments	A short message or explaining information attached to report or table cells.
Conditional formatting	Conditional formatting allows user to adjust appearance of the particular indicators in table according to currently displayed values or their changes.
Copy	A clone of the existing report, table or chart. Any future change of the copy does not affect the original report.
Dashlet	A basic item of the dashboard. Dashlets usually represent your existing views (tables, charts) from reports. Another types of dashlets have informational, supplemental or customizing function.
Data source	Data source is typically a database, Microsoft Excel spreadsheet, text file or another services eg. Google Doc spreadsheet, FaceBook or enterprise services like CRMs (SalesForce, Amiando etc.).
Domain	Whole virtual space of the one registrated organization. Users of one domain can't access data in another domains. Domains are completely independent. Separate domains can also be suitable for individual divisions or SBUs of large international companies.

Drill-down	<p>An operation which results in displaying more detailed data with higher granularity according to the chosen drill-down path. If you drill-down a member, you will see its child members. Eg. drill-down of particular affiliate can display its employees.</p> 
Drill-down path	<p>A sequence of attributes which influences the results of the drill-down operation. It can be defined in the data set or chosen ad-hoc in the report.</p> 
Fact	Facts are equal to the data set indicators
Filters	Filters restricts data displayed in views according to selected members only or general member match pattern. Eg. you can set filter to show only the largest cities in a table or chart. General filter settings are also available when sharing data sets.
Formula	Formula is a tool allowing user to define its own new indicators. Despite using common mathematical functions, advanced functions such as regressions or getting different values of existing indicators in time or depending on their aggregation is possible.
Group of indicators	Group of indicator consists of one or more included indicators which have the same context (eg. wage components). Users can effectively work with the whole group like with a single indicator.
iFrame	iFrame is a feature, which allows users to embed their analytical views into the external website using HTML code.
iGoogle	iGoogle is a feature, which allows users to embed their analytical views into their personal iGoogle dashboard.
Joined data set	An abstract data set, which represents data from two or more source data sets. This allows user to analyze mutual dependencies of data from different data sources or can easily assign member IDs to their names (codebook). Data sets can be connected together according specified condition - inner, outer or cross join. Joined data set reflects all the changes in the source data sets.
Member aggregation	Specifies the way, how are the repeating data records of one member aggregated (in one time unit). The aggregation possibilities are: sum, average, minimum, maximum and count (of different members).
Metric	Metrics are equal to the report indicators
News	Dashlet, which shows actual and former changes mainly in reports and data sets within the domain. This dahlet also displays information about the author of those changes.
Pivot table	An analytical view in a form of flexible rows and columns. Users can adjust, what should be displayed on each axis (various attributes, indicators or time units) and choose the structure of the drill-down path for the current table. It is also possible to analyze indicators in pivot table through the conditional formatting.
Predefined import	Predefined import is a tool, which allows users to easily load columns settings from the first import during repeated imports.
Query Scheduler	A tool, which allows to schedule and control regular data synchronization. It's main purpose is to update data warehouse with actual data from data sources.
Roll-up	An operation, which displays less detailed data with lower granularity according to the chosen drill-down path. It's an inverse operation to drill-down.
Sample data	A set of predefined content, consisting of reports, dashboards, data sets and predefined imports.
Sharing	Sharing is a feature, which allows collaboration and cooperation among users. It's possible to share your data sets, reports or dashboards with other users in the same domain. You can also choose to delegate permission to edit your data sets and reports. Besides iFrame and iGoogle are another sharing tools, which allows you to share your views on corporate intranet or publicly via Internet.
Time aggregation	Specifies the way, how are the time units with higher granularity aggregated in the units with lower granularity. If you (for example) gather data in days but display them in months, the time aggregation determines if the "month" units show sum, average, maximum, minimum or count of the included day records.



User group	A set of individual users. The Domain administrator has a right to create, delete or edit these groups. He also adds new users to groups, remove users from groups and assign user roles to the groups. User group is a very useful tool, because users can share their reports or data sets with the whole group instead of choosing of all its individual members.
User roles	User roles are designed to separate access to different functions for different types of users. Available user roles are "Domain administrator", "Report editor" and "Data manager". These roles can be combined without any restrictions. Only the Domain administrator has a right to delegate those roles to other users.
View	Basic element of the report - displays values of the indicators depending on user's criteria. Views represent flexible points of view to the user's data in a form of pivot table, chart or possibly custom content. Views allow you to use the "slice and dice" feature.

## Permissions and Roles

BellaDati distinguishes between two basic permission schemes:

- Assigning user roles
- Permissions assigned by sharing

### User roles

BellaDati implements following user roles:

- General BellaDati user - this role is assigned by default to all BellaDati users and this cannot be changed. Such users have only the very basic access to BellaDati functions: report and dashboard view, user profile editing

This role is usually sufficient for report or dashboard consumers such as general managers or company management members.

He can't create his own reports, but other users (report editors) can share their reports with him - even with permission to edit their reports.

- Report editor - report editor is able to create, edit, comment and share analytical reports. He can create reports only from his own or shared data sets. Therefore it's usual that users with "report editor role" have also the "data manager" role.
- Data manager - data manager cares about the parts of the data warehouse. His job is to prepare and import data into particular data sets, control and edit the source data, create alarms and join existing data sets. He is also able to prepare translation for particular indicators, attributes and members. He is owner of data sets, which he creates during the imports. He can share data sets with other users (report editors) or directly use them if he has also "report editor" role.
- Domain administrator - domain administrator a specific and important user role. He cares about the users and user groups. He is able to create or import users, delete them, change their profile information and passwords and assign user roles to particular users or whole user groups (he may assign "domain administrator" role to other users as well) . He has available all the statistics of his domain. He can even delete all the content of domain (data sets, reports, dashboards) or forbid the publishing of domain content on the web. He is the only one user who sees all the data sets in his domain. Therefore this user role should be assigned to only one competent user.

It's possible to combine all user roles mentioned above. Eg. both roles data manager and report editor assigned at the same time allows such user performing the whole process from setting up a [data source](#), modelling data set, report creation and sharing dashboards.



System administrator: There is also a System administrator role in BellaDati. This user role is not required for BellaDati Cloud usage. However this role could be useful for BellaDati On-Premise or Unlimited Cloud tariffs, especially for large enterprise companies or international business groups that require managing more separated domains (eg. for their SBU).



User roles can be assigned to user groups too. These roles are merged with standard user roles results - particular user has both roles together.  
Only domain or system administrators can reassign user roles.

## Permissions

Permissions are granted to users while sharing data sets or reports. There two levels of shared permissions:

- Read-only access
- Full access

Owner: Each data set, report or dashboard has always assigned one user that has full access and also can manage sharing in addition to that. These user are called owners and usually are the creators of the data set, report or dashboard.



Permissions assigned by sharing particular data sets or reports have priority over standard user roles. This means user with only general user role assigned can have permission to edit particular data set or report which has been shared with him on full access level!

## Next Steps

- [Managing users](#)
- [Managing user groups](#)
- [Managing user profile](#)

## Data Sets

BellaDati BI has its own integrated data warehouse. This warehouse contains virtual databases that represents data with similar characteristics. These virtual databases are called data sets and [reports](#) are established on them. Each data set can be connected to one [data source](#).

In data set following objects are defined and managed:

- [Indicators](#)
- [Attributes](#)
- [Data sources](#)

Following actions can be performed within data sets:

- [Importing data](#)
- [Browsing, editing and exporting data](#)
- [Structure backup and restore](#)
- [Managing joins](#)
- [Watching data changes](#)
- [Cleaning data](#)
- [Sharing data](#)
- [Creating and removing datasets](#)

In this section:

- [Creating Data Set](#)
- [Importing Data](#)
  - [Importing from Clipboard](#)
  - [Importing from File](#)
  - [Import Settings](#)
- [Browsing Data](#)
- [Managing Indicators](#)
  - [Grouping Indicators](#)
  - [Translating Indicators](#)
  - [Transforming Indicator into Attribute](#)
  - [Formula Reference Guide](#)
- [Managing Attributes](#)
  - [Defining Drill-down Path](#)
  - [Members Appearance](#)
  - [Translating Attributes and Members](#)
  - [Transforming Attribute into Indicator](#)
- [Joining Data Sets](#)
  - [Changing join point](#)
- [Structure Backup](#)
- [Sharing Data Sets](#)
- [Watching Data Changes](#)



Only users with [data manager role](#) are allowed to manage the data sets. If you don't have this role, please contact your BellaDati administrator.

## Data set summary window

Data set summary window displays:

- [Calendar scheduler](#)

Typical data set summary window looks like following:

BellaDati. Desktop Reports Data sets Users Search Help BellaDati Dr. Paul Baker

**Campaigns**

Data set summary

**Settings**

- Attributes (3)
- Indicators (3)
- Joins (0)

**Data**

- Import Data
- Data source
- Browse data
- Create alarm
- Erase data

**Action**

- Share with users
- Backup structure
- Remove Data Set

### Data set summary - Campaigns

Owner: MS dynamics CRM Create report

Previous month
October 2010
Next month

Mon	Tue	Wed	Thu	Fri	Sat	Sun
				1/10	2/10	3/10
4/10	5/10	6/10	7/10	8/10	9/10	10/10
		13:23				
11/10	12/10	13/10	14/10	15/10	16/10	17/10
18/10	19/10	20/10	21/10	22/10	23/10	24/10
25/10	26/10	27/10	28/10	29/10	30/10	31/10

**Reports**

Marketing campaigns

Source data set: Campaigns  
Report owner: MS dynamics CRM

BellaDati 2.5.2.3 © 2008 - 2012 product of Trigram | Support | Privacy Policy | Data Security | Terms of use TRIGRAM

## Action list (submenu layout)

- Data set name: Edit is possible by clicking the name (i.e. in-line edit).
- [Attributes settings](#)
- [Indicators settings](#)
- [Joins settings](#)
- Import data: Allows data manager to [import new data from clipboard or files](#).
- Data source: Management of [connection to external databases, URLs or third party systems](#).
- Browse data: [Data browsing](#), editing and export.
- Create alarm: [Data changes watching](#) option.
- Erase data: Deletes all data from the data set.
- [Share](#) with users: Grants access to the data for other users or user groups, including data filter settings.
- [Backup structure](#): Stores the whole structure of the data set and related reports to XML file.
- Remove data set: Allows removing the whole data set and related reports from BellaDati.

## Creating Data Set

Only users with [data manager role](#) are allowed to create and manage the data set.


Open the Data sets page from main menu on the top of the screen.

1. Click "Create data set" in the left submenu. A popup window appears.
2. Enter name of the new data set.

Data set name must be unique in the whole domain. In case the name isn't unique, warning message will be displayed and the new data set will not be created.

New empty data set contains no indicators and attributes defined. There are two ways how to create them:

- Define indicators and attributes while [importing data](#). If you already have existing data file, we recommend this option to set up the data set more interactively.
- Define [indicators](#) and [attributes](#) manually. The mapping to the imported data will be provided during the import settings stage.

 The data set can be created also using the [data import](#).

## Next steps

- [Import data](#) from your files or [data sources](#)
- Manage [indicators](#) and [attributes](#)
- Continue by [creating report](#) if your data set is ready

# Video tutorial

[How to prepare data set](#)

---

## Importing Data

Data import process allows user to fill BellaDati's data warehouse. Data can be imported also from various sources. There are two types of import:

- Manual import from data files or clipboard content
- Automatic scheduled import from [data sources](#)

### Manual import

You can import from following sources manually by uploading data file into BellaDati:

- [Clipboard](#) content from various applications like MS Excel, web browser, command line, etc.
- Data [file](#) in various formats - from CSV and XML to XLS/XLSX, in ZIP archive as well.
- Adding row using [data browser](#)

### Automatic import

Automatic import enables connecting to external sources and load the data periodically. It is available for following [data sources](#):

- [SQL Databases](#)
- Content available on [URL](#) using the HTTP protocol
- Web services: [Google Analytics](#) | [Google Spreadsheets](#) | [SalesForce](#) | [Facebook](#) | [Amiando](#) | [YouTube](#)

The screenshot shows the BellaDati interface. At the top, there's a navigation bar with 'BellaDati', 'Desktop', 'Reports', 'Data sets', 'Users', a search bar, and user information. The main content area is titled 'Data set summary - Campaigns' with a 'Create report' button. Below this is a calendar for October 2010. A 'Marketing campaigns' report is visible. An 'Import Data' dialog box is open in the foreground, allowing the user to import data from a file. The dialog includes options for import type and file content, a file selection field, and a 'Procházet...' button. The background page also shows a left-hand menu with sections for 'Settings', 'Data', and 'Action'.

You can start the data import in already existing data set or together with [creating new data set](#). To execute data import wizard, click on "Import data" item in the left menu in data set. When you are importing data to a new data set, you must specify it's unique name first. See [creating data set](#) for more details.

## Next Steps

- Configure the [import settings](#) and data mapping
- [Creating report](#) and dashboards
- [Browse imported data](#)
- [Manage indicators](#) and [attributes](#)

## Importing from Clipboard

Clipboard provides a simple way to quickly analyze small piece of your data in BellaDati. Typically you may like analyzing data directly from Microsoft Excel or a table on a webpage.

1. Select desired area in Excel or a web page.
2. Copy it to clipboard (CTRL+C).
3. Open the data sets page and select "Import data" in the left menu, enter the data set name, select "Copy and Paste".
4. Paste (CTRL+V) the clipboard content into text field and click "Continue"

Check import settings now - most settings should be already set correctly automatically (header, encoding, separator).

BellaDati. Desktop Reports Data sets Users Search Help BellaDati Dr. Paul Baker

**Import Data**

Name:

Import type:

Insert text here:

Date	City	Company Name	Country/Region	First Name	Last Name	Rating	Annual Revenue	No.
11.10.2010	Cedar City	Wholesale Bikes	U.S.	Aaron	Con	Hot	100000.0	27.0
11.10.2010	Fort Wayne	Wheel Gallery	U.S.	Abraham	Szwarcengin	Hot	200000.0	30.0
11.10.2010	Dallas	Active Cycling	U.S.	Adrian	Dmitrasacu	Cold	200000.0	73.0
11.10.2010	Campbellville	Lots of Bikes	Storehouse	U.S.	Alan	Bremer	Cold	200000.0
11.10.2010	Indianapolis	The Cycle Store	U.S.	Alvin	Torre	Warm	20000.0	15.0
11.10.2010	La Verne	Sports Merchandise	U.S.	Andrew	Cuscini	Warm	200000.0	83.0
11.10.2010	Seattle	Sample Bike Store	U.S.	Harry	Johnson	Warm	20000.0	5.0
11.10.2010	Hillington	Lease-A-Bike	Shop	U.S.	Bernard	Duerer	Warm	200000.0
11.10.2010	Alb. S. tin	More Bikes!	U.S.	Bonnie	Leppo	Warm	200000.0	97.0
11.10.2010	Wood Dale	New and Used Bicycles	U.S.	Eranson	Jones	Hot	60000.0	13.0
11.10.2010	Camarillo	Extras Sporting Goods	U.S.	Brian	Johnson	Hot	20000.0	2.0
11.10.2010	New York	Outdoor Distributors	U.S.	Carol	Brink	Hot	150000.0	39.0

[CallCenterInvoices](#) Updated from data source on

[Campaigns](#) Updated by MS dynamics CRM on Oct 6, 2010 1:23:04 PM

## Next Steps

- Define the [import settings](#) and provide the data mapping

## Importing from File

The following file formats are supported by BellaDati:

- Plain text file (or CSV - comma separated values)
- Microsoft Excel spreadsheet (XLS, XLSX) - Office 2003, 2007 and 2010 (previous versions not guaranteed)
- XML files
- ZIP files containing one or more supported file formats above

In Import Data popup:

- Open the data sets page and select "Import data" in the left menu and select "Data file". If you are importing into existing data set, open the desired data set and click on "Import data".
- Choose Import type to "Data file" and select right file content according to your file format.
- After selecting the data file, you need to wait until the file is uploaded



Please note, that default maximum file size to import is 20MB. BellaDati Unlimited tariff or BellaDati On-Premise may have different file size limits. You can compress the file size when importing it in a ZIP archive (see below).

When you are importing plain text file (CSV), please continue directly to [Import settings](#) page. In another cases, please continue on following chapters:

### Importing from Microsoft Excel

After uploading XLS/XLSX file you will be prompted to select the desired spread sheet list in next window.



List selection will not appear when your Excel spreadsheet contains only single list.

## Importing from XML

In the XML importing guide, you will be prompted to select the row tag, which represents repeatable data sentence. The following example illustrates it on XML file containing employees:

```
<employees>
  <employee>
    <name>Jan</name>
    <surname>Novak</surname>
  </employee>
  <employee>
    <name>Petr</name>
    <surname>Novak</surname>
  </employee>
  ...
</employees>
```

In this case, the row tag is `<employee>`.

The screenshot shows the BellaDati web interface with a modal dialog titled "XML file import". The dialog is divided into two main sections:

- 1. Select the row tag:** This section shows an XML structure on the left and a preview of the selected row tag on the right. The selected row tag is "item". The preview shows the following data: "VALID: 01-01-1998", "ID\_PERS: 748824", "BASIC\_INFO: PERSONAL\_AREA: 2422, PERSONAL\_GROUP: 10, PERSONAL\_RANGE: 2", "EMPL\_NAME: Bohuslav Kotál", "EMPL\_BIRTHDATE: 01-11-1965", "EMPL\_SEX: M", "EMPL\_HANDICAP:".
- 2. Add custom columns:** This section has a "Tag" dropdown set to "item" and an empty "Attribute" field. A yellow "Add" button is below the dropdown. Below this, it says "No custom columns has been selected."

At the bottom right of the dialog is a yellow "Submit" button.


1. Row tag: Select repeating tag in XML structure. Check extracted content in the preview on the left.
2. Optionally, you can add custom columns repeatedly: Select items and/or attributes when XML structure is not straightforward.

## Importing from ZIP

Importing data compressed as ZIP archive represents an effective way how to reduce imported file size and also upload times significantly. It can contain the following file formats:

- Plain text (CSV)
- Microsoft Excel (XLS, XLSX)
- XML

Please follow corresponding chapters above to continue importing Microsoft Excel or XML file formats.

 New data set will not be created until the import process will have been successfully completed.

## Next Steps

- [Import settings](#)



# Import Settings

Import settings page allows you to control ETL (Extract-Transform-Load) operations and verify the structure of data being imported. The main task is to define the mapping of tabular data to [attributes](#), [indicators](#) and date/time dimensions.

Following adjustments and actions are available:

- First row is header: Use the texts in the first row as names for corresponding columns; only for plain text (clipboard), CSV or Excel
- Excluded rows: Allows you to exclude some rows from the beginning of the file imported (eg. additional information, not data).; only for plain text (clipboard), CSV or Excel
- Encoding: Select appropriate encoding for the source file (UTF-8, ISO-8859-1, Win-1250, Win-1252, Auto are available); only for plain text, CSV or XML
- Separator: Auto-detection (the most frequent separator is the semicolon ";"), otherwise select character that separates each column (comma, tab, semicolon, space, vertical bar, custom); only for plain text (clipboard) and CSV
- Fill the empty cells: Generally for the whole import, or this substitution can be performed individually for particular columns.
- Apply import template: See "import templates" chapter below.
- Use default settings: Resets all import settings to defaults.

Another functions are:

- Data cleaning and transforming using the transformation scripting
- Assigning imported columns to existing attributes or indicators
- Renaming columns
- Column merging
- Adding new columns
- Preview changes

On this page:

- [Column Settings](#)
  - [Time Format](#)
- [Filling of Empty Cells](#)
- [Merging Columns](#)
- [Transformation Scripts](#)
  - [Reusing Transformation Scripts](#)
- [Import Templates](#)
- [Data Categories](#)
- [Data Overwriting Policy](#)
- [Import Progress](#)
- [Next Steps](#)



Automatic encoding detection is not always reliable. We recommend to check for strange characters in the preview.

BellaDati. Desktop Reports Data sets Users Search

Help BellaDati Dr. Paul Baker

Import Data Fill the empty cells... Apply import template... Use default settings **Continue...**

**Columns available**

Search:  Search

All (28) Attribute (12)

Indicator (13) Time (1)

Exclude (2)

Proceed action...

- 1 item
- 2 VALID (dd-MM-yyyy)
- 3 ID\_PERS
- 4 BASIC\_INFO
- 5 PERSONAL\_AREA
- 6 PERSONAL\_GROUP
- 7 PERSONAL\_RANGE
- 8 EMPL\_NAME
- 9 EMPL\_BIRTHDATE
- 10 EMPL\_SEX
- 11 EMPL\_HANDICAP
- 12 EMPL\_EDUCATION
- 13 EMPL\_NATIONALITY
- 14 EMDI\_EMPLOYMENT\_DATE

Preview selected

**Data preview**

1	2	3	4	5	6
item	VALID (dd-MM-yyyy)	ID_PERS	BASIC_INFO	PERSONAL_AREA	PERSONAL_GROUP
01-01-1998	748824			2422	10
01-01-1998	748825			502	2
01-01-1998	748827			3120	13
01-01-1998	748828			1691	7
01-01-1998	748829			502	2
01-01-1998	748830			1451	6
01-01-1998	748831			1931	8
01-01-1998	748832			3142	13
01-01-1998	748833			1222	5
01-01-1998	748834			240	1

(The table shows only the first 10 rows of import and first 15 selected columns.)

BellaDati 2.5.2.3 © 2008 - 2012 product of Trigram | Support | Privacy Policy | Data Security | Terms of use **TRIGRAM**



Availability of the adjustments on the screenshot above could vary depending on file format imported. Options are stated for manual import, see [data sources](#) for specific information about automated imports.

## Column Settings

If you want to change the type of particular column, click on the name of the selected column in the list of columns (in the left side of the import screen). It's also possible to change meaning of more columns to one type in just one step - just mark selected columns by clicking in the checkboxes next to them and then select their meaning from menu above.

There are four possible meanings of columns:

**Time** - time index of particular rows. It can be displayed in a lot of different time formats (also depending on language - for more information see the related part of this chapter). There can be chosen only one time column during the import.

**Attribute** - defines categories of the drill-down path. It's usually a short text (e.g. affiliate, product, customer, employee, division etc.). Every attribute column creates exactly one attribute in the data set. Those attributes can be freely combined in the drill-down paths.

**Indicator** - indicators are usually the numeric data, which are the main point of the user's interest.

**Don't import** - these columns won't be imported at all (it's useful if column contains no, invalid or unimportant data).

Preview of marked columns may be displayed by clicking on the "Preview selected" button. In this way you can get a better view into your data and their meaning settings. If your data contain too much columns, you can use a search label above the column list to find appropriate column and check its settings. Under this searching field is displayed statistics, which shows number of particular types of columns.

## Time Format

Every time column has a specific type of format. This format should be automatically detected during import. However it's possible, that you have your time data in some very specific format. In this case you can use the list of available format in different languages.

If you don't choose from available formats, you can also define your own specific custom format for your data. In this case, you should choose your language from the list below and enter a code, which describes your data format according to these meanings (note, that the number of characters influences the interpretation of the code):

Code	Meaning	Number of characters in code
y	Year	Two characters (yy) represents two digits year number (89). Otherwise is the code interpreted as four digits year number (1989).

M	Month in year	Three or more characters (MMM) are interpreted as text representation of month (e.g. "January" or "Jan"). In other cases are characters interpreted as number of month in year (1-12).
d	Day in month	Number of characters (d) in code should be equal to minimal number of digits in source data. It's always a numeric format.
E	Day in week	Number of character determines, if the day is displayed in its full name (EEEE - "Monday") or in its shortcut (EE - "Mo").

Separator character should be equal to separator character contained in source data (space, dot, semicolon, etc.). If your source data contains time in more separated columns (months, days, years), it's necessary to merge those columns first (described in previous part of this chapter). Next table shows some combination of source data and appropriate time code.

Source data	Appropriate code
09/15/10	MM.dd.yy
26/03/1984	dd/MM/yyyy
15.September 2010	dd.MMMM yyyy
15 Sep 10	dd MMM yy
Wed 15 09 10	EE dd MM yy
Sep 15, 2010	MMM d, yyyy

### Filling of Empty Cells

It's usual that imported data contains empty cells. It's usually necessary to replace this empty cells with own values (e.g. "0", "none", "N/A" etc.). If you want to do this, you have two possibilities, how to fill in these empty cells:

1. globally - fill empty cells with chosen value in all columns (located below batch column settings)
2. locally - fill empty cells with chosen value in particular column (located in the window of particular column settings)

Global changing is available in the top blue line directly under encoding settings. After clicking just type the value, which will be entered in all empty cells in your data.

Local changing is available after clicking on column name in the list. There you can enter your own value for empty cells (but only for this particular column). You can freely combine these two methods - for example you can fill in all the empty cells with "0" value, but one particular attribute column can be refilled with "N/A" value.

### Merging Columns

Merging of columns is useful tool especially in two cases:

- time is separated in more columns (days, months and years in different columns)
- two columns representing one entity (e.g. first name and surname of one person in two separated columns)

You can merge two or more columns after clicking on the symbol of chain in the list of columns (next to column, which you want to merge with another).

Then you have to choose another column for merging with the selected one. You should define a separator character which will be displayed between values of those two columns (space, comma, dot, etc.). Then just confirm your choice or merge selected columns with another one.

If it's necessary to disconnect merged columns, you have to click on the symbol of broken chain in list of columns. Then just click on "Drop merge" link and columns will be separated back at their former positions. Result of merging in our example should look like this:

Another way to merge columns is by [transformation scripts](#).

### Transformation Scripts

Transformation scripts allow advanced data transformations during import. These scripts are based on Groovy programming language syntax.

Transformation scripts allows you the following:

- Modify values stored in BellaDati data warehouse according defined functions and conditions.
- Create new columns (date/time, attributes, indicators) with transformed or combined values from other columns. Values in different cells are indexed from 0 and displayed near column names within import settings screen.
- Perform advanced calculations in date/time (eg. period of some action undertaken between two dates).

Basic script commands:

- `value()` - returns actual value of the current cell
- `value(index)` - returns value of cell on desired (indexed) position in the actual row
- `name()` - returns name of the column
- `name(index)` - returns name of the column at desired position
- `format()` - returns value of the format in actual column (only time and indicator column types)
- `actualDate()` - returns actual date in dd.MM.yyyy format
- `actualDate('MM/dd/yyyy')` - returns actual date in chosen format (e.g. MM/dd/yyyy)
- `excludeRow()` - excludes the row



These transformations are applied for each import including scheduled automatic imports from [Data Sources](#).

Go to [Transformation scripting guide](#) for more details.

## Reusing Transformation Scripts

Previously defined transformation scripts can be invoked for your convenience. You can choose from any existing transformation scripts used in all data sets you have access to. Transformation script list is available after clicking on "Use existing script" link in column settings.

## Import Templates

This function allows you to reuse import settings from previous imports or different data sets. It is available by clicking on "Apply import template" link on the top of the page.

In the popup, you can:

- Select data set
- Select import template assigned to this data set according requested date and import status
- Display import template details (column settings)
- Sort import templates



Applying the template will overwrite all current import settings.

You can choose from any existing import settings used in all data sets you have access to. These templates are created automatically after the import has been successfully finished.

## Data Categories

Basically BellaDati data warehouse currently supports these categories of data according to date/time information present:

- Data containing date, attributes and indicators
- Data containing date & time, attributes and indicators
- Data containing only attributes and indicators - without any date/time information: This is automatically detected and you have two options:
  - Import data without any date information.
  - Add to all records date stamp manually during import.



1. If your file contains more than one date/time column, please set the other date/time columns type to attributes.
2. Indicators are not mandatory. Record containing only attributes may be sufficient in some cases (survey results analysis with count functions ).
3. Data containing only time information are also importable, however their analysis on the report level is limited.

## Data Overwriting Policy

When there are already existing data in the data set, you can choose the following options what to do with these data:

- No overwrite: Imported data will be appended to existing (default).
- Delete data with the same attributes values: Deletes all existing records with the same combination of attributes as in the imported data.
- Delete data with the same date/time: Deletes all existing records containing the same date/time as imported data.
- Delete data with the same attributes values and time: Deletes all existing records with the same combination of date/time and attributes together as in the imported data.
- Delete all data before import.

## Import Progress



Import of lot of data may take a long time to complete.

Data are being imported asynchronously, therefore BellaDati functions are still available during import. The user can be logged out during the import too.

Data set summary page shows actual import progress bar with estimated time and percentage.

Before import finishes, you are able to:

- Cancel running import: All data related to this import will be erased from BellaDati data warehouse.
- Notify by e-mail: An e-mail will be send to you after the import has been finished.

## Next Steps

- Check the [import results](#) after the import
- [Create report](#)
- [Browsing data](#)
- [Transformation scripting - detailed guide](#)

## Import Results

You can display the detailed results for each import. Each import can reach the following status:

- **green** - successfully finished import without errors
- **orange** - finished import containing errors
- **red** - aborted or unfinished import
- **gray** - deleted import
- **blue** - scheduled future import (only in case of external data sources)



Records with errors has not been imported!



Quick access to the most recent import is possible via top blue light bar - hover over the date nearby "Updated by".

BellaDati. Desktop Reports Data sets Users Search Help BellaDati Dr. Paul Baker

**Data set summary - Import Results** Create report  
 Owner: Dr. Paul Baker Created: Jan 27, 2012 3:26:17 PM Updated by Dr. Paul Baker on Jan 27, 2012 3:26:18 PM

**Importing data**  
 0%  
[Cancel](#)  
 Notify by email after import is completed:

**Import detail**

Select column for details:

Column name	Errors count
<b>Datum</b>	1
First name	0
Surname	0
Discount	0
Column 5	0
Formula	0
Column 7	0
Change	0
Column 9	0
Total	0

**Datum**  
 Imported records count:  
 Total: 30  
 Empty: 0  
 Not valid: 1  
 The date has wrong format (1)

**Errors detail**  
 Errors occurred on rows:  
 2

Calendar view for January 2012:

Mon	Tue	Wed	Thu	Fri
2/1	3/1	4/1	5/1	6/1
9/1	10/1	11/1	12/1	13/1
16/1	17/1	18/1	19/1	20/1
23/1	24/1	25/1	26/1	27/1 15:26
30/1	31/1			

BellaDati 2.5.2.5 © 2008 - 2012 product of Trigram | Support | Privacy Policy | Data Security | Terms of use TRIGRAM

If errors occurred during import, you can diagnose the causes in the import detail popup:

- The column name with the error is highlighted in red and errors count is displayed.
- Details for each column (click on it's name):
  - Total: Successfully imported records count.
  - Empty: Records count containing no values in selected columns (these records are imported however).
  - Number (only for indicators): Record counts with numeric values.
  - Not valid: Records with errors count (these records are not imported).
  - Error details: Displays records number in source file to find problem.



Typical errors during import are:

- Date/time format mismatch: Check and set the proper format for date/time column in [Import Settings](#).
- You are trying to assign an indicator not numeric values: Consider setting such column as attribute.

## Next Steps

- [Schedule](#) the automatic import
- Define the [Import Settings](#) during the data import or data source configuration

## Browsing Data

Browsing existing data within data set is available after clicking to "Browse data" in left submenu.

**Leads**  
[Data set summary](#)

**Settings**  
[Attributes \(10\)](#)  
[Indicators \(3\)](#)  
[Joins \(0\)](#)

**Data**  
[Import Data](#)  
[Data source](#)  
[Browse data](#)  
[Create alarm](#)  
[Erase data](#)

**Action**  
[Share with users](#)  
[Backup structure](#)  
[Remove Data Set](#)

**Browse data** From-To: [ ] - [ ] Search Order by: [ ]

[Add row](#) [Save as CSV](#) [Save as ZIP](#)

Rows count: 125

	Date	City	Company Name	Country/Region	First Name	Indus
<input type="checkbox"/>	10/6/2010	Cedar City	Wholesale Bikes	U.S.	Aaron	
<input type="checkbox"/>	10/6/2010	Fort Wayne	Wheel Gallery	U.S.	Abraham	
<input type="checkbox"/>	10/6/2010	Dallas	Active Cycling	U.S.	Adrian	
<input type="checkbox"/>	10/6/2010	Campbellsville	Lots of Bikes Storehouse	U.S.	Alan	
<input type="checkbox"/>	10/6/2010	Indianapolis	The Cycle Store	U.S.	Alvin	
<input type="checkbox"/>	10/6/2010	La Vergne	Sports Merchandise	U.S.	Andrew	
<input type="checkbox"/>	10/6/2010	Seattle	Sample Bike Store	U.S.	Barry	
<input type="checkbox"/>	10/6/2010	Millington	Lease-a-Bike Shop	U.S.	Bernard	
<input type="checkbox"/>	10/6/2010	AU.S.tn	More Bikes!	U.S.	Bonnie	
<input type="checkbox"/>	10/6/2010	Wood Dale	New and Used Bicycles	U.S.	Brannon	
<input type="checkbox"/>	10/6/2010	Camarillo	Extras Sporting Goods	U.S.	Brian	
<input type="checkbox"/>	10/6/2010	New York	Outdoor Distributors	U.S.	Carol	
<input type="checkbox"/>	10/6/2010	Santa Fe	Wide World Importers	U.S.	Caroline	
<input type="checkbox"/>	10/6/2010	Randolph	Tallspin Toys	U.S.	Cindy	
<input type="checkbox"/>	10/6/2010	Fort Worth	Discount Bicycle specialists	U.S.	Connie	
<input type="checkbox"/>	10/6/2010	Vacaville	Travel Sports	U.S.	Daniel	
<input type="checkbox"/>	10/6/2010	Bellevue	Journey Sporting Goods	U.S.	Darren	
<input type="checkbox"/>	10/6/2010	Seattle	Good Toys	U.S.	David	
<input type="checkbox"/>	10/6/2010	Saint Louis	Sure and Reliable Sporting Good	U.S.	David	
<input type="checkbox"/>	10/6/2010	Jefferson City	Cycles and Scooters	U.S.	Deanna	
<input type="checkbox"/>	10/6/2010	Norwood	Trendy Department Stores	U.S.	Donald	
<input type="checkbox"/>	10/6/2010	Upland	Bike Products and Accessories	U.S.	Doris	
<input type="checkbox"/>	10/6/2010	Dallas	Cash and Carry Bikes	U.S.	Dorothy	
<input type="checkbox"/>	10/6/2010	Fontana	Unsurpassed Bikes	U.S.	Edna	
<input type="checkbox"/>	10/6/2010	Fullerton	Fitness Cycling	U.S.	Forrest	
<input type="checkbox"/>	10/6/2010	Phoenix	Great Bikes	U.S.	Francois	
<input type="checkbox"/>	10/6/2010	Vista	Traillblazing Sports	U.S.	Frank	
<input type="checkbox"/>	10/6/2010	Dallas	The Cracker Box	U.S.	Gabrielle	
<input type="checkbox"/>	10/6/2010	Westminster	Curbside Sporting Goods	U.S.	Gladys	
<input type="checkbox"/>	10/6/2010	Newark	Handy Bike Services	U.S.	Grant	
<input type="checkbox"/>	10/6/2010	El Segundo	Optimal Bikes	U.S.	Greg	
<input type="checkbox"/>	10/6/2010	Dallas	Recreation Supplies	U.S.	H.	
<input type="checkbox"/>	10/6/2010	Daleville	Leading Bike Distributors	U.S.	Hannah	
<input type="checkbox"/>	10/6/2010	Camarillo	Little Bicycle Supply Shop	U.S.	Helge	
<input type="checkbox"/>	10/6/2010	Essex	Mobile Bikes	U.S.	Marlene	

Data browser allows you to perform following actions:

- Edit data - individually per record (row)
- Delete row
- Order displayed data by: date or any other attribute
- Filter displayed data:
  - by date interval
  - by attributes members filtering
- Add new record (row)
- Data export: Save data in CSV: Data can be saved in CSV directly or in ZIP format to save download time a space on your storage.

**!** Please note, due to browsers' performance issues, maximum 2000 rows of data is displayed in data browsing window. Use filters to display only desired subset of all data stored.

**!** Deleting one row is performed immediately after clicking corresponding icon! There is no confirmation message displayed before.

**Next Steps**

- [Importing data](#)
- [Creating report](#)

**Managing Indicators**

Managing the indicators is a part of BellaDati's data warehouse modelling. Values of particular indicators represent the point of the examination interest. Indicators have usually numeric values (wages, costs, incomes etc.).

We are distinguishing two types of indicators:

- data set indicators are defined within the data set and are available as masters for report indicators. Another settings (like aggregations,

appearance, etc.) are not supported. Simply said, the data set indicators represents a raw numerical value (in the OLAP language it is a fact) with basic attributes - name, unit and rounding mode. Values of these indicators are straightforwardly stored in BellaDati's data warehouse directly from imported data.

**i** Each data set indicator is represented by its code (unless defined by formula). This code always begins with "M\_" prefix and serves as unique identification for usage in formulas. Note: Indicator code is not editable and is assigned during it's creation.

- [report indicators](#) are created in the report from the data set indicators or ad hoc. Unlike the data set indicators, report indicators are supporting wide range of various settings - [aggregations](#), [appearance](#), [conditional formatting](#), extended [formula support](#) with [Report Variables](#) etc. Report indicators can contain also non-numerical values.

	Action	Code / Formula	
Budget Allocated		M_BUDGET_ALLOCATED	Remove
Estimated Revenue		M_ESTIMATED_REVENUE	Remove
Expected Response Percentage		M_EXPECTED_RESPONSE_PERCENTAGE	Remove

- **Formula indicators:** Their value is calculated according assigned mathematical formula or another operation. Values of indicators created by formula are not imported to data set, they are evaluated from formula. Formula indicators can be defined on report level as well. See [Formula Reference Guide](#) for details.

**i** Data set indicators contain numeric values only. Formula indicators on the other side can contain also textual values or member counts.

## Creating data set indicator

New indicator can be created by following ways:

- Within data set (data set indicators)
- During the [import](#)
- Transforming an [attribute](#) into an data set indicator
- When inserting data row manually

## Data set indicator settings

When you click on the name of particular indicator name in the data set, you can adjust a lot of different parameters of this indicator. You can define or change:



- Indicator name
- Indicator unit
- Decimal format (see chapter below)
- Type of rounding
- Assign indicator to group
- [Formula](#). This function is available only when the indicator is defined by formula or when creating new one.

Those adjustments will take effect in each occurrence of this indicator (data sets, reports, dashboards).



When the data set is a part of [join](#), you can choose if you want to propagate new indicator to joined data set (by checking the propagate option in dialog window).

## Decimal format

Decimal format is a useful tool to:

- insert separate characters into indicator
- insert additional characters into unit
- adjust number of decimal positions

Indicators without adjusted decimal format are displayed with a comma after each three positions of digits (thousands, millions etc.) and with dot between whole number and decimal places. You can use prepared help window in BellaDati for some examples, how to define the decimal format. Basic description is also in the table below:

Code	Meaning
#	Represents one digit
,	Separator of digit places (thousands, millions etc.). It's displayed as a comma in English localization.
.	Decimal separator. It's displayed as a dot in english localization.
%	Multiply value by 100 and add % symbol
"	If you want to display any of described operational character (., #), you have to place it between " characters. It's a useful function, when you deal with shortcuts which ends with "." character.

Decimal format examples:

Actual value	Display	Code
1234	1234	#
1234	\$1234	\$#
1234.56	\$1 235	##,###
1234.56	1 234,6	#,### #
1234.56	\$1 234, 56	##,###.##
1234	\$1 234 k	##,### k
0.56	56 %	# %
0.5612345	56.12345%	#####%



You can display brief decimal format help by clicking on "Show help" link in the popup window.

## Removing Indicator

Click on the cross icon (Remove link) at the end of corresponding row in indicators' list. Then confirm the removal in popup.



Indicator removal will affect all reports and dashboards where the indicator has been used!

## Next Steps

- [Grouping indicators](#)
- [Creating report](#)
- [Importing data](#)
- [Browsing data](#)
- [Transforming indicator into attribute](#)

## Grouping Indicators

Indicators grouping enables better structuring and organizing of your indicators. Each indicator can be assigned to the indicator group and each group can be nested to another group.

The main advantage of indicator groups is that you handle all contained indicators as a single object.

Each group contains embedded functions like sum or average from the nested indicators or sub-groups.



Indicator groups should contain indicators with similar characteristics - eg. financial indicators, performance indicators etc.

There are three ways how to create indicator groups:

- manually on the data set indicators page
- in the [indicator settings](#) dialog window
- within the report

## Next Steps

- [Managing indicators](#)
- [Creating report indicators](#)

## Translating Indicators

Translation is useful tool when you have imported data in one language but then you want to display your data in another language - eg. for presentation to managers. Then if you change the language in user profile or switch it in your browser, all the translated indicators will be displayed in new alternative language. Indicators without translation remain displayed the same way (unchanged).





Logout and repeated login to BellaDati may be needed to reflect language change.

The screenshot shows the BellaDati web application interface. The top navigation bar includes 'BellaDati', 'Desktop', 'Reports', 'Data sets', 'Users', and a search bar. The left sidebar contains sections for 'Campaigns', 'Settings', 'Data', and 'Action'. The main content area is titled 'Indicators' and displays a table of indicators. A 'Translation' popup window is open, allowing the user to create a new translation for the 'Estimated Revenue' indicator. The popup includes a 'Language' dropdown set to 'Czech', a 'Translation' input field, and an 'Add' button. Below the input field is a table showing existing translations for 'de' (Erwartet Umsatz). The main indicator table in the background lists 'Budget Allocated', 'Estimated Revenue', and 'M\_EXPECTED\_RESPONSE\_PERCENTAGE' with 'Remove' buttons for each.

Go to desired data set via data set menu.

1. Select "Indicators" in the left submenu.
2. Click on the translation icon at next to the name of selected indicator. A popup will appear. All existing translations of actual are displayed in the table below.
3. Select the target language in "Language" drop-down box.
4. Type translated name into "Translation" field.
5. Click "Add" button. The new translation will be added to the table below.
6. Choose another language and repeat the process or click "Close" to close popup.

 To edit existing translation, please remove the actual translation first by clicking the cross icon at the end of corresponding row. Then add the translation again according instructions above.

 Indicator translation will not be reflected in views, where the indicator has been renamed before! See report indicator editing for details.

## Next Steps

- Localization settings in [user profile](#)

## Transforming Indicator into Attribute



Please use this function carefully especially when the data has been already imported into the data set. Changing an existing indicator to attribute may affect data structure mapping while using [import template](#) during next data imports.

Also be sure no [formulas](#) are based on this indicator before performing the transformation. Otherwise some data in reports and/or dashboards may become unavailable.

The screenshot shows the BellaDati web interface. The top navigation bar includes 'BellaDati', 'Desktop', 'Reports', 'Data sets', 'Users', a search bar, and user information for 'Dr. Paul Baker'. The main content area is titled 'Indicators' and contains a table with the following data:

	Action	Code / Formula	
Budget Allocated		M_BUDGET_ALLOCATED	Remove
Department		M_DEPARTMENT	Remove
Estimated Revenue		M_ESTIMATED_REVENUE	Remove
Expected Response Percentage		M_EXPECTED_RESPONSE_PERCENTAGE	Remove

A modal dialog box titled 'Change to attribute' is open in the foreground. It contains a warning icon and a checked checkbox with the text 'Are you sure to transform indicator to an attribute?'. Below the checkbox are two buttons: 'Back' and 'Continue'.

After transformation the original indicator is no longer available and you will find it as an attribute under [Attributes](#) overview. All existing values of the indicator have been transformed to [members](#). The attribute gets new code with "L\_" prefix.

## Next Steps

- [Managing attributes](#)
- [Importing data](#)
- [Transforming attribute into indicator](#)

## Formula Reference Guide



This summary provides an overview of all formulas that can be used in reports (as report indicators) or data sets (as predefined data set indicators).

This reference guide has nothing to do with the transformation scripting. If you are interested in data cleaning and transformation, visit the [transformation scripting guide](#) instead.

### Indicators and codes

Each data set indicator is specified by its unique code starting with M\_ (M as Measure). Accessing the calculated indicator's value is possible by typing this code directly into the formula. For example:

```
M_NAME_1 + M_NAME_2
```

Another way how to get the value of the indicator is to use the `value()` function:

```
value('M_NAME_1') + value('M_NAME_2')
```

Both examples have the same result.

On this page:

- Indicators and codes
  - Drill-down (members) aggregation
  - Date-Time aggregation
  - Drill-down and date-time aggregation
  - Counting of level members
- Datetime functions
  - Date strings
  - Changing datetime context
  - Advanced functions
  - Nested expressions
- Working with datetime constants
  - Accessing context datetime
  - Accessing actual date
- Math functions
  - Regression functions
- Special functions
  - Passing parameters to time formula
- Report variables

### Drill-down (members) aggregation

Members aggregation determines the way how to count values in the case that exists more records for one selected member in a single time unit. Aggregation type is specified by adding the appropriate suffix to the indicator's code. When not specified, the `SUM` aggregation is applied.

Suffix	Description	Example
@SUM	Calculates the sum of all values for the selected drill-down attribute	M_NAME@SUM
@MIN	Calculates the minimum of all values for the selected drill-down attribute	M_NAME@MIN
@MAX	Calculates the maximum of all values for the selected drill-down attribute	M_NAME@MAX
@AVG	Calculates the average of all values for the selected drill-down attribute	M_NAME@AVG
@DC	Calculates the distinct count of all values for the selected drill-down attribute	M_NAME@DC

### Date-Time aggregation

The date-time aggregation specifies the way, how to count values in the case that you display indicator values in higher time units (lower granularity data) than the time units in which are data stored in the system (higher granularity data).

Suffix	Description	Example
@SUMT	Calculates the sum of all values for the selected time interval	M_NAME@SUMT
@MINT	Calculates the minimum of all values for the selected time interval	M_NAME@MINT
@MAXT	Calculates the maximum of all values for the selected time interval	M_NAME@MAXT
@AVGT	Calculates the average of all values for the selected time interval	M_NAME@AVGT
@DCT	Calculates the distinct count of all values for the selected time interval	M_NAME@DCT

### Drill-down and date-time aggregation

Both drill-down and date-time aggregation can be specified simultaneously.

```
M_NAME@SUM@SUMT
```

Both aggregation methods can be used in any other formulas when relevant.

### Counting of level members

Each drill-down level is represented by particular members, for example the level City contains members like Berlin, Paris, New York etc. To get the count of these members, use the following syntax:

```
L_NAME_COUNT //L_NAME represents the level code (prefix L like level)
L_NAME_DCOUNT //to get the unique count use the _DCOUNT suffix instead
```

### Datetime functions



For the purpose of this reference guide: Date refers only to year, months, days (and quarters, weeks) and their combinations. Time refers to hours, minutes and seconds and their combinations. For combination of date and time we strictly use datetime term.

### Date strings

The date string parameters are entered absolutely (dd.MM.yyyy or yyyy-MM-dd) or relatively (time variables) by operators:

```
date +/- n[d|w|m|q|y]
```

where

- date is date in dd.MM.yyyy or yyyy-MM-dd format, or one of actualyear, actualmonth, actualquarter, actualweek, actualday, now
- n represents the count of:
- d days, w weeks, m months, q quarters or y year.

Examples:

```
'2011-01-01 + 1q - 2m + 1d' means Jan 1, 2011 + 1 quartal - two months + 1 day, that represents Feb 2, 2011
'actualYear + 1m + 1d' , (used on 8 Mar, 2011) means 1 Feb, 2011 + 1 month + 1 day, that represents 2 Feb, 2011 as well
```

Another way how to create the date strings is following:

```
date(day: dd, month: MM, year: YYYY)
date(d: dd, m: MM, y: YYYY)
date(week: w, year: YYYY)
date(quarter: q, year: YYYY)
```

### Changing datetime context

What is the datetime context? Consider following example:

	01/2011	02/2011	03/2011
formula indicator	1000	1200	1300

Formula is evaluated for each column - in this example, in columns are values evaluated for particular months. During the evaluation of value 1000, the datetime context was 01/2011, then during the processing of value 1200, the context was 02/2011 etc.

Function	Description
<pre>dateAt(String dateString, { expression })</pre>	<p>Changes the context of the evaluated expression to dateString date. Example:</p> <pre>dateAt('2011-01-01', {   i1 = M_NAME_1   i2 = M_NAME_2   if (i1 &gt; i2) {     return i1;   }   return i2; });</pre>
<pre>dateInterval(String from, String to, { expression })</pre>	<p>Changes the date context of the expression and evaluates it aggregated in the specified interval from - to. Example:</p> <pre>dateInterval('2011-01-01', '2011-12-31' {   i1 = M_NAME_1   i2 = M_NAME_2   if (i1 &gt; i2) {     return i1;   }   return i2; });</pre> <p>Values for the indicators M_NAME_1 and M_NAME_2 are aggregated for the whole period.</p>
<pre>timeAt(String timeString, { expression })</pre>	<p>Changes the context of the evaluated expression to timeString date. Example:</p> <pre>timeAt('13:33:23', {   i1 = M_NAME_1   i2 = M_NAME_2   if (i1 &gt; i2) {     return i1;   }   return i2; });</pre>
<pre>timeInterval(String from, String to, { expression })</pre>	<p>Changes the time context of the expression and evaluates it aggregated in the specified interval from - to. Example:</p> <pre>timeInterval('00:00:01', '23:59:59' {   i1 = M_NAME_1   i2 = M_NAME_2   if (i1 &gt; i2) {     return i1;   }   return i2; });</pre> <p>Values for the indicators M_NAME_1 and M_NAME_2 are aggregated for the whole period.</p>

#### Obsolete functions



These functions may be removed in further releases

Function	Description
<code>value(String dateString, String indicator)</code>	Loads the indicator's value at specified date.
<code>value(String dateString, int drill_down_level)</code>	Loads the indicator's value at specified date aggregated for N previous levels.
<code>value(String dateFrom, String dateTo, String indicator)</code>	Loads cumulative indicator's value for specified date interval.
<code>value(String dateFrom, String dateTo, String indicator, int drill_down_level)</code>	Loads cumulative indicator's value for specified date interval aggregated for N previous levels.

### Advanced functions

Function	Description												
<code>cumulateFromDate(String startDate, String indicator)</code>	<p>This function gradually adds the current value to the cumulated value. Example:</p> <p><input type="checkbox"/></p> <p>   01/2011   02/2011   03/2011  </p> <table border="1"> <tr> <td>M_NAME_1</td> <td>1000</td> <td>1200</td> <td>1300</td> <td></td> <td></td> </tr> <tr> <td><code>cumulateFromDate('2011-01-01', 'M_NAME_1')</code></td> <td>1000</td> <td>2200</td> <td>3500</td> <td></td> <td></td> </tr> </table>	M_NAME_1	1000	1200	1300			<code>cumulateFromDate('2011-01-01', 'M_NAME_1')</code>	1000	2200	3500		
M_NAME_1	1000	1200	1300										
<code>cumulateFromDate('2011-01-01', 'M_NAME_1')</code>	1000	2200	3500										
<code>cumulateFromTime(String startTime, String indicator)</code>	<p>This function gradually adds the current value to the cumulated value. Example:</p> <p><input type="checkbox"/></p> <p>   00:01   00:02   00:03  </p> <table border="1"> <tr> <td>M_NAME_1</td> <td>1000</td> <td>1200</td> <td>1300</td> <td></td> <td></td> </tr> <tr> <td><code>cumulateFromTime('00:01', 'M_NAME_1')</code></td> <td>1000</td> <td>2200</td> <td>3500</td> <td></td> <td></td> </tr> </table>	M_NAME_1	1000	1200	1300			<code>cumulateFromTime('00:01', 'M_NAME_1')</code>	1000	2200	3500		
M_NAME_1	1000	1200	1300										
<code>cumulateFromTime('00:01', 'M_NAME_1')</code>	1000	2200	3500										



<pre>prev(String indicatorCode)</pre>	<p>Value of the passed indicator calculated for previous date or time value (e.g. previous month, day, hour, ...). Example:</p> <p><input type="checkbox"/></p> <p>   01/2011   02/2011   03/2011  </p> <table border="1" data-bbox="717 373 1195 474"> <tr> <td>M_NAME_1</td> <td>1000</td> <td>1200</td> <td>1300</td> <td></td> <td></td> </tr> <tr> <td>prev('M_NAME_1')</td> <td></td> <td>1000</td> <td>1200</td> <td></td> <td></td> </tr> </table>	M_NAME_1	1000	1200	1300			prev('M_NAME_1')		1000	1200		
M_NAME_1	1000	1200	1300										
prev('M_NAME_1')		1000	1200										
<pre>prev(String indicatorCode, int prevLevelAgg)</pre>	<p>Value of the passed indicator calculated for previous date or time value (e.g. previous month, day, hour, ...). Parameter <code>prevLevelAgg</code> represents number of previous levels which should be aggregated.</p>												
<pre>next(String indicatorCode)</pre>	<p>Value of the passed indicator calculated for next date or time value (e.g. previous month, day, hour, ...). Example:</p> <p><input type="checkbox"/></p> <p>   01/2011   02/2011   03/2011  </p> <table border="1" data-bbox="717 989 1195 1089"> <tr> <td>M_NAME_1</td> <td>1000</td> <td>1200</td> <td>1300</td> <td></td> <td></td> </tr> <tr> <td>next('M_NAME_1')</td> <td>1200</td> <td>1300</td> <td></td> <td></td> <td></td> </tr> </table>	M_NAME_1	1000	1200	1300			next('M_NAME_1')	1200	1300			
M_NAME_1	1000	1200	1300										
next('M_NAME_1')	1200	1300											
<pre>next(String indicatorCode, int prevLevelAgg)</pre>	<p>Value of the passed indicator calculated for next date or time value (e.g. previous month, day, hour, ...). Parameter <code>prevLevelAgg</code> represents number of previous levels which should be aggregated.</p>												

For changing the whole context of the evaluated expression, you can use following functions:

<pre>prev(String period, { expression })</pre>	<p>Changes the context of the expression to desired previous date period. The value of the period parameter is one of: DAY, WEEK, MONTH, QUARTER, YEAR. Example:</p> <p><input type="checkbox"/></p> <p>   01/2011   02/2011   03/2011  </p> <table border="1" data-bbox="604 1680 1195 1780"> <tr> <td>M_NAME_1</td> <td>1000</td> <td>1200</td> <td>1300</td> <td></td> <td></td> </tr> <tr> <td>prev(MONTH) { M_NAME_1 }</td> <td></td> <td>1000</td> <td>1200</td> <td></td> <td></td> </tr> </table>	M_NAME_1	1000	1200	1300			prev(MONTH) { M_NAME_1 }		1000	1200		
M_NAME_1	1000	1200	1300										
prev(MONTH) { M_NAME_1 }		1000	1200										

```
next(String period, { expression
})
```

Changes the context of the expression to desired next date period. The value of the period parameter is one of: DAY, WEEK, MONTH, QUARTER, YEAR

The following example works with values, which are loaded for one year before the actual table/chart datetime entry:

```
return prev(YEAR) {
  M_NAME_1 / M_NAME_2 * 100
}
```



You may simply enter only the first letter of time unit (in case of time context changing formulas) instead of their full names (Y,Q,M,W,D), for example prev(Y){}

### Nested expressions

Date and time functions can be combined (if applicable), eg.: Value of M\_INDICATOR at 8th December 2010, 9:04:02AM:

```
dateAt ('2010-12-08') { timeAt('09:04:02') { M_INDICATOR } }
```

### Working with datetime constants

#### Accessing context datetime

Function	Description
String contextDay()	Returns the formula's context day in dd.MM.yyyy format.
String contextWeek()	Returns the beginning of the context's week in dd.MM.yyyy format. Example:
String contextMonth()	Returns the beginning of the context's month in dd.MM.yyyy format.
String contextQuarter()	Returns the beginning of the context's year in dd.MM.yyyy format.
String contextYear()	Returns the formula's context week start in dd.MM.yyyy format.
int dateDayOfYear()	Returns the day of year from the context date
int dateDayOfMonth()	Returns the day of month from the context date
int dateDayOfWeek()	Returns the day of week from the context date
int dateMonth()	Returns the number of month from the context date
int dateYear()	Returns the number of month from the context date
int timeHour()	Returns the number of hour from the context time
int timeMinute()	Returns the number of minute from the context time
int timeSecond()	Returns the number of second from the context time

Examples:

	31.1.2011 00:01:00	1.2.2011 00:01:00	2.2.2011 00:01:00
contextDay()	31.1.2011	1.2.2011	2.2.2011
contextWeek()	31.1.2011	31.1.2011	31.1.2011
contextMonth()	1.1.2011	1.2.2011	1.2.2011
contextQuarter()	1.1.2011	1.1.2011	1.1.2011

contextYear()	1.1.2011	1.1.2011	1.1.2011
dateDayOfYear()	31	32	33
dateDayOfMonth()	31	1	2
dateDayOfWeek()	1	2	3
dateMonth()	1	2	2
dateYear()	2011	2011	2011
timeHour()	0	0	0
timeMinute()	1	1	1
timeSecond()	0	0	0

### Accessing actual date

Function	Description
String actualDay()	Returns the actual day in dd.MM.yyyy format
String actualWeek()	Returns the actual week in dd.MM.yyyy format
String actualMonth()	Returns the actual month in dd.MM.yyyy format
String actualQuarter()	Returns the actual quarter in dd.MM.yyyy format
String actualYear()	Returns the actual year in dd.MM.yyyy format

### Math functions

Function	Description
double abs(double a)	Returns the absolute value of a double value.
float abs(float a)	Returns the absolute value of a float value.
int abs(int a)	Returns the absolute value of an int value.
long abs(long a)	Returns the absolute value of a long value.
double acos(double a)	Returns the arc cosine of a value; the returned angle is in the range 0.0 through pi.
double asin(double a)	Returns the arc sine of a value; the returned angle is in the range -pi/2 through pi/2.
double atan(double a)	Returns the arc tangent of a value; the returned angle is in the range -pi/2 through pi/2.
double atan2(double y, double x)	Returns the angle theta from the conversion of rectangular coordinates (x, y) to polar coordinates (r, theta).
double cbrt(double a)	Returns the cube root of a double value.
double ceil(double a)	Returns the smallest (closest to negative infinity) double value that is greater than or equal to the argument and is equal to a mathematical integer.
double copySign(double magnitude, double sign)	Returns the first floating-point argument with the sign of the second floating-point argument.
float copySign(float magnitude, float sign)	Returns the first floating-point argument with the sign of the second floating-point argument.
double cos(double a)	Returns the trigonometric cosine of an angle.
double cosh(double x)	Returns the hyperbolic cosine of a double value.

<code>double exp(double a)</code>	Returns Euler's number e raised to the power of a double value.
<code>double expm1(double x)</code>	Returns $e^x - 1$ .
<code>double floor(double a)</code>	Returns the largest (closest to positive infinity) double value that is less than or equal to the argument and is equal to a mathematical integer.
<code>int getExponent(double d)</code>	Returns the unbiased exponent used in the representation of a double.
<code>int getExponent(float f)</code>	Returns the unbiased exponent used in the representation of a float.
<code>double hypot(double x, double y)</code>	Returns $\sqrt{x^2 + y^2}$ without intermediate overflow or underflow.
<code>double IEEERemainder(double f1, double f2)</code>	Computes the remainder operation on two arguments as prescribed by the IEEE 754 standard.
<code>double log(double a)</code>	Returns the natural logarithm (base e) of a double value.
<code>double log10(double a)</code>	Returns the base 10 logarithm of a double value.
<code>double log1p(double x)</code>	Returns the natural logarithm of the sum of the argument and 1.
<code>double max(double a, double b)</code>	Returns the greater of two double values.
<code>float max(float a, float b)</code>	Returns the greater of two float values.
<code>int max(int a, int b)</code>	Returns the greater of two int values.
<code>long max(long a, long b)</code>	Returns the greater of two long values.
<code>double min(double a, double b)</code>	Returns the smaller of two double values.
<code>float min(float a, float b)</code>	Returns the smaller of two float values.
<code>int min(int a, int b)</code>	Returns the smaller of two int values.
<code>long min(long a, long b)</code>	Returns the smaller of two long values.
<code>double nextAfter(double start, double dir)</code>	Returns the floating-point number adjacent to the first argument in the direction of the second argument.
<code>float nextAfter(float start, double dir)</code>	Returns the floating-point number adjacent to the first argument in the direction of the second argument.
<code>double nextUp(double d)</code>	Returns the floating-point value adjacent to d in the direction of positive infinity.
<code>float nextUp(float f)</code>	Returns the floating-point value adjacent to f in the direction of positive infinity.
<code>double pow(double a, double b)</code>	Returns the value of the first argument raised to the power of the second argument.
<code>double random()</code>	Returns a double value with a positive sign, greater than or equal to 0.0 and less than 1.0.
<code>double rint(double a)</code>	Returns the double value that is closest in value to the argument and is equal to a mathematical integer.
<code>long round(double a)</code>	Returns the closest long to the argument.
<code>int round(float a)</code>	Returns the closest int to the argument.
<code>double scalb(double d, int scaleFactor)</code>	Return $d \times 2^{\text{scaleFactor}}$ rounded as if performed by a single correctly rounded floating-point multiply to a member of the double value set.
<code>float scalb(float f, int scaleFactor)</code>	Return $f \times 2^{\text{scaleFactor}}$ rounded as if performed by a single correctly rounded floating-point multiply to a member of the float value set
<code>double signum(double d)</code>	Returns the signum function of the argument; zero if the argument is zero, 1.0 if the argument is greater than zero, -1.0 if the argument is less than zero.
<code>float signum(float f)</code>	Returns the signum function of the argument; zero if the argument is zero, 1.0f if the argument is greater than zero, -1.0f if the argument is less than zero.

<code>double sin(double a)</code>	Returns the trigonometric sine of an angle.
<code>double sinh(double x)</code>	Returns the hyperbolic sine of a double value.
<code>double sqrt(double a)</code>	Returns the correctly rounded positive square root of a double value.
<code>double tan(double a)</code>	Returns the trigonometric tangent of an angle.
<code>double tanh(double x)</code>	Returns the hyperbolic tangent of a double value.
<code>double toDegrees(double angrad)</code>	Converts an angle measured in radians to an approximately equivalent angle measured in degrees.
<code>double toRadians(double angdeg)</code>	Converts an angle measured in degrees to an approximately equivalent angle measured in radians.
<code>double ulp(double d)</code>	Returns the size of an ulp of the argument.
<code>float ulp(float f)</code>	Returns the size of an ulp of the argument.
<code>long factorial(int value)</code>	Returns the factorial of passed value.

### Regression functions

Function	Description
<code>linereg(String indicatorCode)</code>	Linear regression
<code>polyreg(2, String indicatorCode)</code>	Quadratic regression
<code>polyreg(3, String indicatorCode)</code>	General polynomial regression

### Special functions

Function	Description
<code>Double aggregatePrevLevel(int countOfPrevLevels, { expression } )</code>	<p>Returns the aggregated value of the embedded expression. The aggregate computed for the specified upper level. For example:</p> <pre> total = aggregatePrevLevel(1, {     cumulateFromTime(actualYear(), {         'M_NAME_1' + 'M_NAME_2'     }); }); current = cumulateFromTime(actualYear(), {     'M_NAME_1' + 'M_NAME_2' }); return current / total; </pre>
<code>Double members(String path, { expression } )</code>	<p>Computes the expression for desired drill-down member values. Example</p> <pre> return members(' [L_LEVEL_1=VAL][L_LEVEL_2="VAL1,VAL2"] ', {     M_NAME_1 * M_NAME_2 }) </pre> <p>This example returns the value of indicator <code>M_NAME_1</code> aggregated for specified members.</p>

<pre>Double membersSum({ expression } )</pre>	<p>Computes the expression for particular members and calculates the sum particular results. Following example shows the difference of using multiplication within the <code>membersSum()</code> method and outside:</p> <p><input type="checkbox"/></p> <pre>    Ind1   Ind2   membersSum( { M_IND_1 * M_IND_2 } )   M_IND_1 * M_IND_2  </pre> <table border="1" data-bbox="820 430 1230 579"> <tr> <td>Member</td> <td></td> <td>5</td> <td>30</td> <td>80</td> <td>150</td> <td></td> </tr> <tr> <td>Member</td> <td>Child1</td> <td>3</td> <td>20</td> <td>60</td> <td>60</td> <td></td> </tr> <tr> <td>Member</td> <td>Child2</td> <td>2</td> <td>10</td> <td>20</td> <td>20</td> <td></td> </tr> </table>	Member		5	30	80	150		Member	Child1	3	20	60	60		Member	Child2	2	10	20	20	
Member		5	30	80	150																	
Member	Child1	3	20	60	60																	
Member	Child2	2	10	20	20																	
<pre>Double membersSum(String levelCode, { expression } )</pre>	<p>Computes the expression for particular members of the defined <code>levelC</code> and calculates the sum from particular results.</p>																					
<pre>Double membersAvg({ expression } )</pre>	<p>Computes the expression for particular members and calculates the average value from particular results.</p>																					
<pre>Double membersAvg(String levelCode, { expression } )</pre>	<p>Computes the expression for particular members of the defined <code>levelC</code> and calculates the average value from particular results.</p>																					
<pre>Double membersMin({ expression } )</pre>	<p>Computes the expression for particular members and calculates the minimal value from particular results.</p>																					
<pre>Double membersMin(String levelCode, { expression } )</pre>	<p>Computes the expression for particular members of the defined <code>levelC</code> and calculates the minimal value from particular results.</p>																					
<pre>Double membersMax({ expression } )</pre>	<p>Computes the expression for particular members and calculates the maximum value from particular results.</p>																					
<pre>Double membersMax(String levelCode, { expression } )</pre>	<p>Computes the expression for particular members of the defined <code>levelC</code> and calculates the maximum value from particular results.</p>																					
<pre>String memberValue()</pre>	<p>Returns the value of current member. Example:</p> <p><input type="checkbox"/></p> <pre>   Population   memberValue()  </pre> <table border="1" data-bbox="820 1512 1156 1661"> <tr> <td>Prague</td> <td>1200000</td> <td>Prague</td> <td></td> </tr> <tr> <td>Berlin</td> <td>3000000</td> <td>Berlin</td> <td></td> </tr> <tr> <td>London</td> <td>7825200</td> <td>London</td> <td></td> </tr> </table>	Prague	1200000	Prague		Berlin	3000000	Berlin		London	7825200	London										
Prague	1200000	Prague																				
Berlin	3000000	Berlin																				
London	7825200	London																				

**Passing parameters to time formula**

It is possible to define the time entry by formula. Following functions are applicable for these kind of formulas only.

Function	Description
<code>void set(String name, Object value)</code>	Sets the parameter value.
<code>Object get(String name)</code>	Loads the parameter value.
<code>Object indicator()</code>	Returns indicator from the context of the row of current table. This function returns also values of formula defined indicators (defined in report). Example: <div style="border: 1px dashed blue; padding: 5px; margin: 10px 0;"> <pre>value(actualYear(), 'actualMonth - 1m - 1d', indicator())</pre> </div>

Consider following example - we have several indicators with codes `M_NAME_1`, `M_NAME_2` and `M_NAME_3`. These indicators are used in formulas 1 - 3. The time area is defined by time formulas 1 - 5.

	Time formula 1: <code>M_NAME_1</code>	Time formula 2: <code>M_NAME_2</code>	Time formula 3: <code>M_NAME_3</code>	Time formula 4: <code>s = get('suffix'); return value('M_NAME' + s);</code>	Time formula 5: <code>value(actualYear(), 'now-1m', indicator())</code>
Formula 1: <code>set('suffix', '_1'); return M_NAME_1;</code>	1000	2000	3000	1000	1000
Formula 2: <code>set('suffix', '_2'); return M_NAME_2;</code>	1000	2000	3000	2000	2000
Formula 3: <code>set('suffix', '_3'); return M_NAME_3;</code>	1000	2000	3000	3000	3000
<code>M_NAME_1</code>	1000	2000	3000	N/A	1000
<code>M_NAME_2</code>	1000	2000	3000	N/A	2000
<code>M_NAME_3</code>	1000	2000	3000	N/A	3000

## Report variables

Report variables are accessible using the `@` prefix. For example:

```
dateAt(@dateParameter, {
  return M_NAME_1;
});
```

Variables can be used in formulas and in custom date/time interval definition.

## Managing Attributes


Attributes definition is a part of the BellaDati's data warehouse modelling. All attributes have an unique code beginning with "L\_" prefix (L as level) that serves as unique identifier (eg. for counts, filtering, custom members definition, etc.).


Instance of particular attribute is called [member](#).

## Creating Attribute

Attributes can be created in three different ways:


- during the [data import](#)
- manually on the data set attributes page
- during [drill-down path definition](#)

 Attribute name must be unique in the whole data set. Otherwise warning message will be displayed and the attribute will not be saved.

 When the data set is a part of [join](#), you can choose if you want to propagate new attribute to joined data set (by checking the propagation option in dialog window).

## Editing Attribute

You can modify the attribute by clicking on it's name in the list.

- 
1. All changes in attributes name will influence immediately all existing [Reports](#) and [Dashboards](#) containing affected attributes.
  2. Changing of attribute name does not affect it's code
  3. Changing of attribute name will not affect attribute's names in joined data sets.

## Deleting Attribute

The attribute will be deleted with all corresponding data.





Attribute removal will influence immediately all existing [Reports](#) and [Dashboards](#) (including filters and custom members). All data assigned to this attribute will be deleted.

## Next Steps

- [Defining drill-down path](#)
- [Creating report](#)
- [Importing data](#)
- [Browsing data](#)
- [Transforming attribute into indicator](#)

## Defining Drill-down Path

Predefined drill-down path specifies the meaningful sequence might be useful for further drill-down operation in report (see attached video tutorial below). Each data set can contain more drill-down paths. The simplest drill-down path is linear, however BellaDati supports more complex structures like trees.



Predefined drill-down paths are available int the report when performing ad-hoc drill-down (by clicking on + in the table), or in the drill-down settings.

The screenshot shows the BellaDati web application interface. The top navigation bar includes 'BellaDati', 'Desktop', 'Reports', 'Data sets', 'Users', and a search bar. The user is logged in as 'Dr. Paul Baker'. The left sidebar contains sections for 'Campaigns', 'Settings' (Attributes, Indicators, Joins), 'Data' (Import, Source, Browse, Alarm, Erase), and 'Action' (Share, Backup, Remove). The main content area is titled 'Attributes' and lists 'Campaign Type', 'Department', 'Name', and 'Status Reason'. Below this is a 'Predefined drill down paths' section with a tree diagram showing 'Campaign Type' branching into 'Status Reason' and 'Name'. A modal window 'Insert attribute into drill down path' is open, showing a dropdown for 'Use attribute' and a text input for 'Add new attribute' containing 'Product'. 'Back' and 'Save' buttons are visible in the modal.

1. Go to desired data set via data set menu.
2. Select "Attributes" in the left submenu.



Predefined drill-down paths significantly simplifies [creating reports](#) by analysts and performing drill-down operation in reports and on dashboards. Therefore we recommend defining these paths on the data set level.

## Adding new drill-down path

1. Click "Add drill down path" at the bottom of the page. A popup will appear.
2. Select first attribute in "Use attribute" drop-down box and click on "Save" button.
3. Click the green plus icon on the right of existing attribute in drill-down path to add another attribute and repeat steps above to create whole drill-down path.



By clicking on green plus icon in the middle of existing drill-down path, you can branch out current drill-down path to more sub-paths.



You can create an new attribute during drill-down path definition by filling in the field "Add new attribute" in the popup. See [managing attributes](#) for details.

### Editing drill-down path

- Removing attribute: Hoover over the desired attribute, click on the right top cross symbol and confirm this action in popup.

### Next Steps

- [Managing attributes](#)
- [Importing data](#)
- [Creating report](#)

## Video tutorial

Drill-down path in report

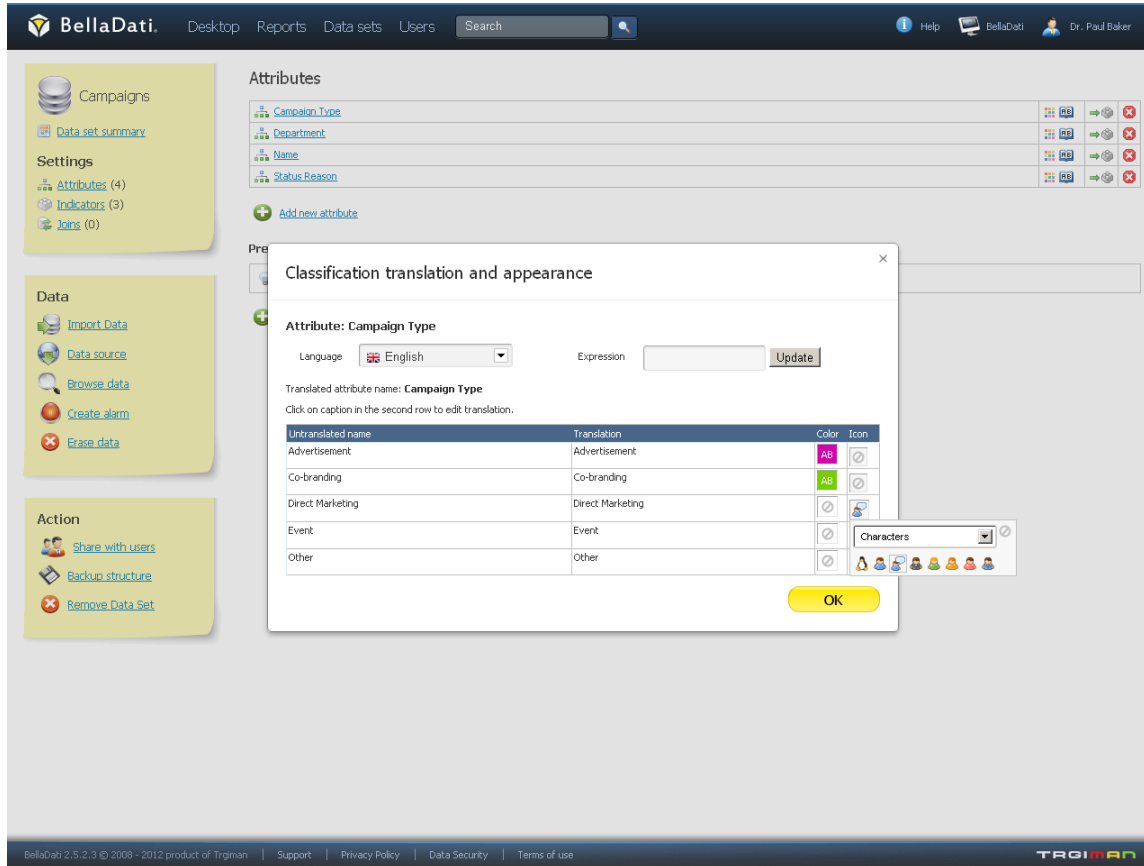
---

## Members Appearance

This function allows you to adjust following appearance options for individual attribute [members](#):

- color
- icon

Both can be assigned to particular member simultaneously.



1. Go to desired data set via data set menu.
2. Select "Attributes" in the left submenu.
3. Click on the translations and appearance icon at the end of the row of selected indicator. A popup will appear. Actual color and icon assignment is displayed in the table below.
4. Color: Click icon in "Color" column in the row of corresponding member. A color selection tool will appear. Click on desired color. Note: Clicking on "Default" field will reset member color to default (none).
5. Icon: Click icon in "Icon" column in the row of corresponding member. An icon selection tool will appear. Select desired category in drop-down box and then click required icon. Note: Clicking on left top symbol will reset member color to default (no icon).
6. Click "OK" to close popup.



When there are lot of members, you can filter them by using "Expression" field. Just type in the part of the names of requested members and then click "Update" button. Empty field means no member filter is active.



All changes in member appearance will influence immediately all existing [Reports](#) and [Dashboards](#) displaying these members.

## Next Steps

- [Translating attributes and members](#)

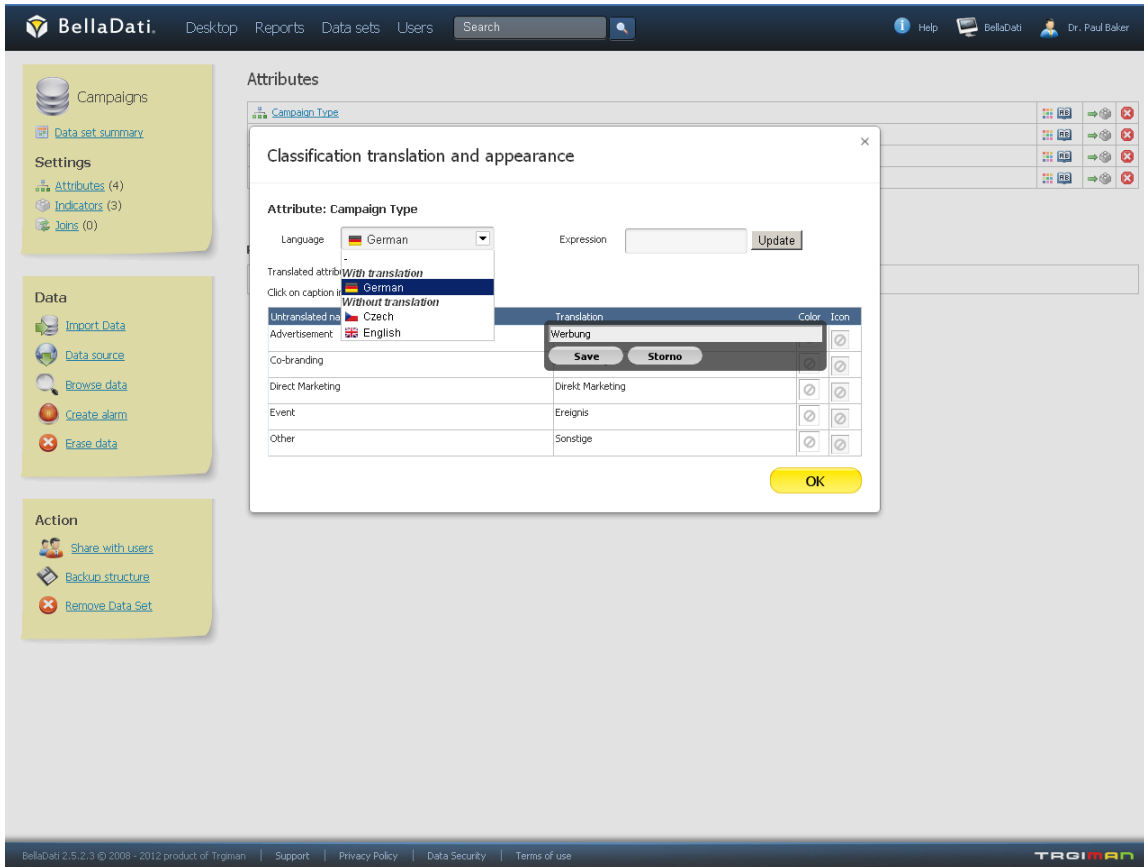
## Translating Attributes and Members

This function allows you to add or edit translations for:


- [Attributes](#)
- Attribute [members](#)


Translation is useful tool when you have imported data in one language but then you want to display your data in another language - eg. for


presentation to managers. Then if you change the language in [user profile](#) or switch it in your browser, all the translated attributes and members will be displayed in new alternative language. Attributes and members without translation remain displayed the same way (unchanged).



1. Go to desired data set via data set menu.
2. Select "Attributes" in the left sub-menu.
3. Click on the translations and appearance icon at the end of the row of selected attribute. A popup will appear.
4. Select the target language in "Language" drop-down box. Now you see actual attribute and its' members translations in the selected language (indicated also in drop-down box - With translation section).
5. Edit the attribute translation by clicking it's name in "Translated attribute name:" row via in-line edit function (bold). Then click "Save" button.
6. Edit members translation by clicking their names on corresponding row in "Translation" column and in-line editing. Then click "Save" button.
7. Click "OK" to close popup.

 Attributes and members translation is available in reports as well.

 When there are lot of members, you can filter them by using "Expression" field. Just type in the part of the names of requested members and then click "Update" button. Empty field means no member filter is active.

 All attribute and member translations will influence immediately all existing [Reports](#) and [Dashboards](#) using these attributes and members.

## Next Steps

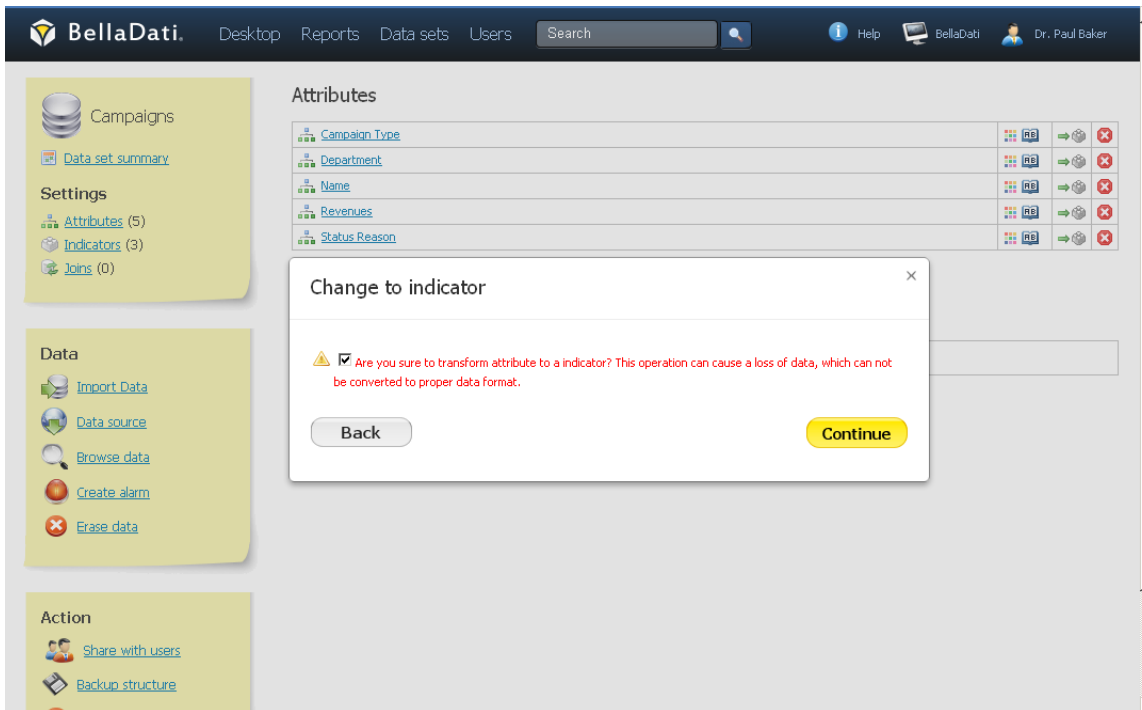
- [Members Appearance](#)
- Language setting in [user profile](#)

## Transforming Attribute into Indicator



Please use this function carefully especially when data has already been imported to the data set. Transforming an existing indicator into attribute may affect data structure mapping while using import template during next data imports.

Also be sure that no formulas are based on this indicator before performing the transformation. Otherwise some data in reports and/or dashboards may become unavailable.



The original attribute is no longer available now and you will find it as an indicator under Indicators left submenu. The indicator gets new code with "M\_" prefix.



Only numeric members will be transformed to indicator values! Other members will be ignored and corresponding indicator values will be empty. This operation is usually irreversible therefore.

### Next Steps

- [Managing indicators](#)
- [Importing data](#)
- [Transforming Indicator into Attribute](#)

## Joining Data Sets

Joined data sets allow you to analyze data from more data sets together and therefore use data from more [data sources](#) in a single report. This function might help you to compare internal company data to external - business intelligence then easily becomes competitive intelligence (CI).



The principle is similar to joining SQL database tables.

One data set could be joined with more data sets (eg. join by department ID to get department full name, address, country, total sales and then join by product ID to get products name, price, weight and mass). Each join is called join point.

# Video tutorial

## Joining of data sets

---

### Join condition

Each join has following parameters:

- Name: Joined data set name.
- Join with: Name of the source data set to join with.
- Compare type: Date/time in both records are taken in account or not. Note: Codebook records usually do not contain time information.
- Join condition: Standard join types available - match always depends on attributes.
  - Left outer join: Record in the target data set is not mandatory.
  - Inner join: Record in the target data set is mandatory.
  - Cross join: No attribute match required.

The resulting joined data set will contain all the attributes and indicators from all source data sets.



1. Data in the joined data set will be the intersection of all joins.
2. Data in joined data sets are updated automatically when data are changed or imported to the source data sets. You cannot import data directly to the joined data sets.



1. Please note, that the permissions setting based on data filter is not available joined data sets!
2. You cannot join a data set with already joined data set.
3. Data join is also available on data source level (database). Please consider this option when you plan to analyze millions records of data. Joining on database level may provide better performance.

Joined data set supports these functions the same way like in ordinary data set:

- [Data browsing and export](#)
- [Data changes watching](#)
- [Structure backup](#)

## Next Steps

- [Browse and export](#) data
- [Watching data changes](#)
- [Create report](#)
- [Backup and restore](#) joined data set
- Modifying [join points](#)

## Changing join point

You are allowed to perform following actions on existing joined data set:

- Changing of join type
- Changing of date/time matching condition
- Deleting the join point (or deleting all join points)
- Adding new join point



All changes will be propagated to joined data and related reports or dashboards immediately.

## Structure Backup

This function allows you to backup existing structures in BellaDati and migrate them to another instance (eg. from Cloud to On-Premise). The following structures are included in this backup:

- Data sets: attributes, indicators, indicator groups and their settings (appearance, format, formulas), data set [owners](#), sharing settings
- Data source settings
- Alarms
- Joins
- Reports: Views (tables structure, chart types and their structure), report indicators, report [owners](#), sharing settings.

BellaDati. Desktop Reports Data sets Users Search Help BellaDati Dr. Paul Baker

**Data set is a virtual database. It can be incrementally extended with more columns, rows.**

Common use it to fill it with data from one data source like XML or databases. Within the data sets you can create the individual attributes, composing them as drill down path branching and define indicators. These indicators can also be combined into formulas.

**Action**

- Create data set
- Import Data
- Load from XML
- Backup structure
- Data changes watch

**Data sets**

Name:  Search  Show permissions details

Name:  Updated from data source on

**Backup structure**

Select data sets to export:

Name	Selected
<input type="checkbox"/> AAA	
<input type="checkbox"/> Accounts	
<input type="checkbox"/> Alfa 13	
<input type="checkbox"/> Alfa-mionet 13	
<input type="checkbox"/> Approximation laptops	
<input type="checkbox"/> A-sklad-eng	
<input type="checkbox"/> BBB	
<input checked="" type="checkbox"/> CallCenterInvoices	
<input checked="" type="checkbox"/> Campaigns	
<input checked="" type="checkbox"/> CashFlow	

1 2 3 4 5 6 7 8 9 10 11 ... 14 15

Back Continue

Updated by [Jiri Urbanek](#) on Sep 16, 2010 11:59:47 AM

1 2 3 4 5 6 7 8 9 10 11 ... 14 15

BellaDati 2.5.2.5 © 2008 - 2012 product of Trigram | Support | Privacy Policy | Data Security | Terms of use **TRIGRAM**

**i** There are two ways of this structure backup:

- Selected data set and reports based on its data backup.
- Bulk backup of more selected data sets, their settings and related reports.

**w** When exporting data sets created by joins, all basic data sets will also be automatically exported.

## Exporting XML Structure

**w** Data are not exported. Use separate [exporting data](#) function to backup data!

**w** Import settings (templates), dashboards, users and user groups are not included in XML structure backup. We recommend to create users, user groups and assigning roles to them before importing XML backup. See also [\[user import|User import\]](#) feature.

## Importing XML Structure

A wizard is available during XML structure import. The preview of imported data sets and report parameters is displayed.

Several checks are performed during XML structure import:

- User and user groups: You can select another users or user groups from the existing accounts in the domain. Note: All data sets and reports must have owners!
- Indicator and attribute assign.

You can manually exclude some data sets and reports from the imported XML structure.





Existing XML structure can be modified - see [Setup Data Model using XML](#) for details.

## Next Steps

- [Browsing and Exporting data](#)
- [User import](#)

## Sharing Data Sets



Data set sharing functions are only available for the owners of the particular data set.

Data set sharing functions allows you to perform following actions:

- Grant access to data for selected [users](#) or [user groups](#)
- Allow access to data for all users in the domain globally
- Optionally notify users about granted access to data sets
- Restrict access to shared data at data level by defining the data filter for users and user groups

When you are setting up data set sharing for individual users or user groups, please distinguish following two [access levels](#):

- Read-only access: Only reports or dashboards can be created.
- Full access: All functions except sharing and data set deleting are available.

As soon as the data set is shared with the user, he is able to create report and then dashboards based on shared data. In addition to that, users with full access can also manage the data set in the same way as it's owner except sharing settings or complete removal.

The screenshot shows the BellaDati web interface. The top navigation bar includes 'Desktop', 'Reports', 'Data sets', 'Users', a search bar, and user information for 'Dr. Paul Baker'. The main content area is titled 'Data set summary - Campaigns' with the owner 'MS dynamics CRM'. A 'Create report' button is visible. A 'Share with users' dialog box is open, showing a list of users: 'Jiri Urbanek' and 'Carl Keys', each with a 'Remove' button. Below the list, there is an 'Add User' section with a text input field for 'Name, user login:', checkboxes for 'Allow editing' and 'Notify users', and an 'Add' button. A 'Close' button is at the bottom of the dialog. The left sidebar contains various menu items like 'Campaigns', 'Settings', 'Data', and 'Action'. The footer contains version information and links for 'Support', 'Privacy Policy', 'Data Security', and 'Terms of use'.

## Data Set Filter

You can restrict access for selected users or user groups at data level. Eg. large companies can have more SBUs requiring the same data

however each SBU management should have access only to the data concerned with their SBU. On the other side general management can still access aggregated data for all SBUs. Data set filter function allows you to restrict sharing only to records that contains selected members (eg. SBU or department name). All aggregations for these users is then available only on these data.

Data set filter functions exactly the same as for [report](#).



Data set member filter is not available for global data set sharing.

Always thoroughly plan member filter settings in connection with [user roles and permissions](#)! Otherwise data leakage may occur when sharing reports or dashboards based on this data set.

## Next Steps

- [User administration](#)
- [User group administration](#)
- [Creating and sharing Reports](#)

## Watching Data Changes

This function is also known as alarms. Data watching function allows to monitor actual data in the data set and launch alerts when the values fulfill predefined conditions.

One data set can contain more alarms.

Alarm parameters:

- Name
- Watch period (repeating interval): Data for this period are aggregated and checked against the condition repeatedly every first day after selected period (1st February for aggregated data of January, then 1st March for aggregated data of February etc.).
- Continuous check: Condition is checked every day. Data are aggregated for time period above, however this period is sliding - eg. on 10th February for aggregated data of 9th January to 9th February.
- Send e-mail: When alert is launched, the user will be also notified by e-mail. By standard, user is notified in alert actions list or via recent changes dashlet on dashboard.
- Condition: Alarm condition. (See also "conditional formatting".)
  - Indicator nad it's aggregation.
  - Condition: equal, lower, greater, decrease , increase.
  - Value: Comparison to that absolute value.
  - Compared to: previous value (previous watch period or previous year).

Following actions are available in alarms list:

- Delete: Completely removes alarm from data set.
- Disable (deactivate) alarm: Condition checking and alerts will be disabled when the alarm is suspended.
- Activate alarm: Changes suspended alarm to active state - condition checking will be restored.
- Launch now (on demand condition checking): Allows user to execute condition checking for the particular alarm manually (see also "Continuous check above"). This is mainly for testing purposes.

The screenshot shows the BellaDati web interface. At the top, there is a navigation bar with 'BellaDati', 'Desktop', 'Reports', 'Data sets', and a search bar. The user is logged in as 'Jiri Urbanek'. The main content area displays 'Data set summary - Revenues' with a 'Create report' button. A 'Create alarm' dialog box is open, showing the following details:

- Name:** Revenue too low
- Watched period:** Month
- Alarm conditions:** Notify me when: Revenue (Mio EUR) (Sur); Condition: lower than; Value: 1000000
- Options:**  Continuously check the current values of the actual time period;  Send me an e-mail on alert.

Buttons for 'Back' and 'Create' are visible at the bottom of the dialog. The footer of the page includes 'BellaDati 2.5.2.4 © 2009 - 2012 product of Trigram' and the TRIGRAM logo.

## Data Sources

Data sources are generally all third party systems BellaDati can import data from. BellaDati contains a set of functions and wizards to help you establishing connection to these data sources, managing them and also provides diagnostic tools. Part of the data sources management is also automatic import scheduler.

BellaDati currently supports these data sources:

- [SQL databases](#)
- [Content available on URL using the HTTP protocol](#) (web services, shared files, etc.)
- [Google Analytics and Google Spreadsheets](#)
- [FaceBook](#)
- [SalesForce](#)
- [Amiando](#)
- [YouTube](#)



Only users with data manager role or full access permission to the data set can set up and control data sources. See [BellaDati permissions and roles](#) for details.

Database

Data set summary

Settings

- Attributes (0)
- Indicators (0)
- Joins (0)

Data

- Import Data
- Data source
- Browse data
- Create alarm

Action

- Share with users
- Backup structure
- Remove Data Set

This data set is not connected to data source.

Select data source type you want the data set to connect to, or use existing data source [data source](#)

<b>SQL Database Connector</b> Create business reports and dashboards directly from the warehouse data in a minute thanks to import wizard with column type recognition. Build intelligence over your plain data with a ROLAP core and in-memory reporting speed. Start using your data, compare history, reveal trends. Share the results with management in a user-interactive environment.	<b>Reporting for Google Spreadsheets</b> Provide users with an agile spreadsheet editing environment while enabling management to see professional and aggregated values. Transform drafts and valuable data divided into teams' documents into interactive and reliable charts. Automatic format detection and spreadsheets-to-reports are a part of the wizard.
<b>Site success analytics for Google Analytics</b> Monitor real ROI and marketing influence of your sales within a unified business intelligence environment. This app helps you to analyse site traffic in interactive KPI-driven marketing reports.	<b>Content from URL</b> Documents of CSV, XLS/XLSX or XML types available on public website or your corporate network will become the data source. Also choose this option in case you want to load data through Web Services provided by other applications (eg. SAP).
<b>Social Marketing Analytics for your Facebook Pages</b> Monitor visitor's behavior of your social site pages. You can combine popularity indicators with your other marketing data available. Identify your most successful marketing strategy from impressions and traffic count indicators segmented by content type, regions and user's profile.	<b>Management-level dashboards for Salesforce</b> SalesForce data source provides an insight into performance and status of your sales pipeline. Browse aggregated campaign results with forecasts with a drill down up to particular sales team members and opportunities.
<b>Youtube</b> Access statistics about visitors of your YouTube videos.	<b>Event Management Insights for Amiando</b> Take control over the budget and revenue of your online events managed in Amiando. Data source provides real-time reports presenting finance indicators, sold ticket amounts by types, regions and referral sources info. Discover popularity trends, predict attendance to eliminate costs.

## Modifications and Operations

General actions:

- Load data: Launches instant synchronization ([overwriting policy](#) and repeating interval can be set).
- Check availability: Allows you to verify if the data source is available.
- Import settings: Allows you to change [import mapping](#) to reflect data source structure changes.
- Schedule: Links to [Synchronization scheduler](#).
- Delete: Delete the data source and all its settings. Data already imported will remain intact.
- Cancel scheduled executions: You can cancel future planned synchronization.



Each data source has specific configurable parameters - see details for particular data source.

## Next Steps

- [Creating data set](#)
- [Import settings](#)

## Video tutorial

Connection to external data source

Connecting to Google Analytics

Connecting to Google Docs

Connection to database

How to analyze data from World Bank

## Connecting to Database

### Supported databases

Depending on application usage (cloud or on-premise) BellaDati is able to connect to these SQL databases:

Name	Supported versions	Driver versions
Oracle	10.1.0.5, 10.2.0.1-10.2.0.5, 11.1.0.7, 11.2.0.1, 11.2.0.2	v6-11.1.0.7, v6-11.2.0.1, v6-11.2.0.2, v14-10.1.0.5, v14-10.2.0.1, v14-10.2.0.2, v14-10.2.0.3, v14-10.2.0.4, v14-10.2.0.5
MySQL	3.1.3 or higher	5.1.13
PostgreSQL	8 or higher	9.0-801-jdbc4
Microsoft SQL Server	MSSQL 2008, MSSQL 2008 R2, MSSQL 2008 Express	sqljdbc4
SAP Max DB	7.3 or higher	7.4.4 Build 003-000-002-502



Database support varies by BellaDati Cloud tariff or On-Premise integration environment.



Connection to many more database vendors is possible using general JDBC drivers. Please consult our [support](#) about particular solutions.

## Video tutorial

Connection to database

## Connection parameters

You must specify the following parameters:

- host: IP address or domain name
- port
- database (sid): database name
- user
- password
- driver: If different drivers are required for various database versions, you can select the right version here (eg. Oracle).

Additional parameters can be specified.



Connection parameters above may vary according the database vendor. Please, refer to your database vendor`s documentation for details.

Connection to the database will be checked immediately - if a problem arises, you will be informed via error message. Please also check the firewall settings - if BellaDati can connect to the data source.

In the next step, provide the SQL command to load the data.

## Modifications

Following data source parameters can be modified within the existing data source via data set summary:

- Connection parameters
- SQL statement



When changing SQL query to extract more columns from the database, use Reset values function and then perform new import settings; otherwise the additional columns will not be imported.

## Next Steps

- Take me back to [data sources](#)
- [Import settings](#)
- [Scheduling Import](#)

## Connecting to Google

BellaDati provides connector to favorite Google services: Google Docs and Google Analytics

Video tutorial

## Connecting to Google Analytics

---

## Connecting to Google Docs

---

### Authentication

Before accessing Google services, authentication is required.



The login data you enter are stored encrypted and not shared with any other application. This login and password is used only to enable data transfer from secured Google API and is not allowed to extract any other personal information from your Google Account.

### Google Analytics

1. Select site
2. Select metrics and attributes



Some metrics and attributes combinations are not allowed.  
See [Google Analytics documentation](#) for details.

### Modifications

The following modifications are available for the existing Google Analytics data source:

- From/To date interval: Influences the period data are imported within (see entering date/time parameters section below).
- Authentication revocation

### Entering date/time parameters

- you can enter time (date) absolutely in two different formats: dd.MM.yyyy (e.g. 1.12.2010), or yyyy-MM-dd (e.g. 2010-12-01)
- it's also possible to enter date relatively:
  - now - represents actual date
  - actualyear - represents the first day of actual year (1.1.20XX). For example actualyear selected on 21.9.2010 represents date 1.1.2010
  - actualquarter - represents the first day of actual quarter (1.1.20XX, 1.4.20XX, 1.7.20XX, 1.10.20XX). For example actualquarter selected on 21.9.2010 represents date 1.7.2010
  - actualmonth - represents the first day of actual month (1.1.20XX, 1.2.20XX, ...). For example actualmonth selected in 21.9.2010 represents date 1.9.2010
  - actualweek - represents first day of actual week (Monday). For example actualweek selected on 21.9.2010 represents date

20.9.2010 (Monday of this week in calendar)

- relative and absolute enterig of date can be adjusted by operators using this syntax: date +/- n[d|w|m|q|y], where n is integer, d represents day, w represents week, m represents month q represents quartal and y represents year. We can for example define time in this way: actualyear + 2m -4d. Today is 21.9.2010, so this value represents 1.1.2010 + 2 months - 4 days, which means date 25.2.2010.

## Google Spreadsheets

1. Select one of your spreadsheets.
2. Select the desired list.

## Modifications

Following actions can be performed for existing Google data source>

- Google Spreadsheet change
- Revoke granted access - available on the data source overview page

## Next Steps

- Take me back to [data sources](#)
- [Import settings](#)
- [Scheduling Import](#)

## Connecting to Salesforce

BellaDatI has connector to Salesforce CRM implemented. Basic steps to connect to Salesforce are the following:

The screenshot displays the BellaDatI web application interface. At the top, there is a navigation bar with the BellaDatI logo and menu items: Desktop, Reports, Data sets, Users, and a search bar. A sidebar on the left contains sections for 'SalesForce' (with a 'Data set summary' link), 'Settings' (with links for 'Attributes (0)', 'Indicators (0)', and 'Joins (0)'), 'Data' (with links for 'Import Data', 'Data source', 'Browse data', and 'Create alarm'), and 'Action' (with links for 'Share with users', 'Backup structure', and 'Remove Data Set').

The main content area shows a message: "This data set is not connected to data source." Below this is a "Select data source settings template" dialog box. The dialog contains the following text: "BellaDatI contains predefined data areas for the analyse. Select one of them, or insert your own SOQL query." It lists five options with radio buttons: "Analyze your Campaigns", "Analyze your Opportunities records with Owner, Account and Campaign details", "Analyze your Campaign details - campaign members, leads and contacts", "Analyze your accounts related with opportunities", and "Set SOQL query directly (for advanced users only)". A yellow "Continue" button is at the bottom of the dialog.

Below the dialog, there are four analytics cards: "Social Marketing Analytics for your Facebook Pages" (with Facebook logo), "Management-level dashboards for Salesforce" (with Salesforce logo), "Youtube" (with YouTube logo), and "Event Management Insights for Amiando" (with Amiando logo). Each card provides a brief description of the analytics it offers.

At the bottom of the page, there is a footer with the text: "BellaDatI 2.5.2.5 © 2008 - 2012, product of Trigram | Support | Privacy Policy | Data Security | Terms of use" and the Trigram logo.

## Authentication



First you have to login to Salesforce and grant access to data stored there.



The login data you enter are stored encrypted and not shared with any other application. This login and password is used only to enable data transfer from secured Salesforce API and is not allowed to extract any other personal information from your Salesforce account.

## Data Area Selection

Select desired area of Salesforce data to analyse or general SOQL query:

- Campaigns
- Opportunities records with Owner, Account and Campaign details
- Campaign details - campaign members, leads and contacts
- Accounts related with opportunities
- Set [SOQL](#) query directly (for advanced users only)

## Extracting SOQL Columns

You may require only a subset of columns returned by executing SOQL query. This function allows you to define which columns will be finally imported to BellaDati's data warehouse.

## Modifications

Following specific data source parameters can be modified within the existing data source via data set summary:

- SOQL query
- Extracted SOQL columns
- Salesforce authentication revoke: You can terminate BellaDati's access to your data in Salesforce.

## Next Steps

- Take me back to [data sources](#)
- [Import settings](#)
- [Scheduling Import](#)

## Connecting to Amiando

To import data from Amiando, perform the following steps:

1. [Get Amiando API key](#) and enter it in BellaDati.
2. Choose which columns do you want to import in [import settings](#).

BellaDati Desktop Reports Data sets Users Search Help BellaDati Dr. Paul Baker

This data set is not connected to data source.

Select data source type you want the data set to connect to, or use existing data source [data source:](#)

**SQL Database Connector**  
Create business reports and dashboards directly from the warehouse data in a minute thanks to import wizard with column type recognition. Build intelligence over your plain data with a ROLAP core and in-memory reporting speed. Start using your data, compare history, reveal trends. Share the results with management in a user-attractive environment.

**Reporting for Google Spreadsheets**  
Provide users with an agile spreadsheet editing environment while enabling management to see professional and aggregated values. Transform drafts and valuable data divided into teams' documents into interactive and reliable charts. Automatic format detection and spreadsheets-to-reports are a part of the wizard.

**Data source settings**

To allow BellaDati using data from Amianto, you have to obtain and enter the Amianto API key below:  
[How to get Amianto API key?](#)

API key  
XA0Q0dsTndFGSFRvJkKuxXoClJk90y0GEs2GvHchLTOi8

Save

BellaDati 2.5.2.5 © 2008 - 2012 product of Trigram | Support | Privacy Policy | Data Security | Terms of use

BellaDati selects predefined columns from Amianto only. Please contact our [support](#) for options how to obtain another data from Amianto.

## Specific Modifications

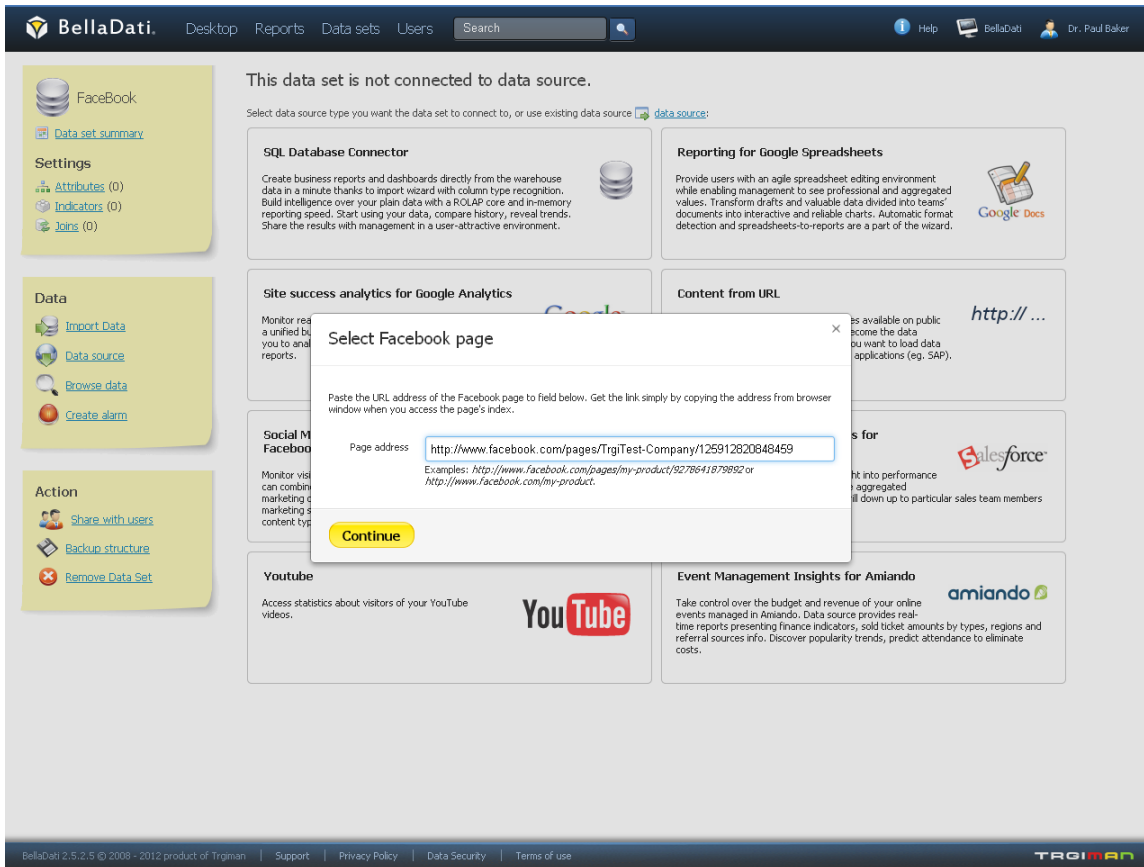
- API key change
- Import settings: Reset import columns settings

## Next Steps

- Take me back to [data sources](#)
- [Import settings](#)
- [Scheduling Import](#)

## Connecting to FaceBook

BellaDati allows you to connect and analyze data from FaceBook easily. The connection wizard is provided.



Please perform the following steps to connect to Facebook:

## Authentication

You have to grant BellaDati access to the statistics of your FaceBook account.



BellaDati Social Analytics App will access the Page Insights data only. Aggregated data are stored in BellaDati according to the Facebook API terms of use. Authentication is provided by Facebook servers thus we do not store your login fields.

## Page And Column Selection

1. Select the desired page: Enter the URL of the desired FaceBook page (this is usually a product or company page).
2. Select the requested columns via [Import settings](#).

## Data Source Settings and modifications

- Date from and to (dd.MM.yyyy): Sets time period within to import data (see entering date/time parameters section below).
- Reset import columns settings

## Entering date/time parameters

- you can enter time (date) absolutely in two different formats: dd.MM.yyyy (e.g. 1.12.2010), or yyyy-MM-dd (e.g. 2010-12-01)
- it's also possible to enter date relatively:
  - now - represents actual date
  - actualyear - represents the first day of actual year (1.1.20XX). For example actualyear selected on 21.9.2010 represents date 1.1.2010
  - actualquarter - represents the first day of actual quarter (1.1.20XX, 1.4.20XX, 1.7.20XX, 1.10.20XX). For example actualquarter selected on 21.9.2010 represents date 1.7.2010
  - actualmonth - represents the first day of actual month (1.1.20XX, 1.2.20XX, ...). For example actualmonth selected in 21.9.2010 represents date 1.9.2010

- actualweek - represents first day of actual week (Monday). For example actualweek selected on 21.9.2010 represents date 20.9.2010 (Monday of this week in calendar)
- relative and absolute enterig of date can be adjusted by operators using this syntax: date +/- n[d|w|m|q|y], where n is integer, d represents day, w represents week, m represents month q represents quartal and y represents year. We can for example define time in this way: actualyear + 2m -4d. Today is 21.9.2010, so this value represents 1.1.2010 + 2 months - 4 days, which means date 25.2.2010.

## Next Steps

- Take me back to [data sources](#)
- [Import settings](#)
- [Scheduling Import](#)

## Connecting to URL

Connecting to URL has similarities with [file import](#). In addition URL imports can be scheduled to execute automatically and repeatedly. URL popup offers advanced section for authentication and specific HTTP headers settings.

### Basic Settings

Enter the following parameters to connect to URL source:

- URL: Web address
- File content: Select the file format - CSV (text file), Excel (XLS, XLSX), XML or ZIP

Then continue setting like for [file import](#).

### Date and time Variables

If you need to change the URL or HTTP post content dynamically, you can use predefined variables. BellaDati currently supports functions to get date, time or timestamp in user defined formats:

Name	Description	Examples
<code>\$date(dateString)</code>	Evaluates the <code>dateString</code> and outputs the date in <code>yyyy-MM-dd</code> format. The <code>dateString</code>	<code>\$date(now + 5d -4w)</code> <code>\$date(2011-01-01 + 5d -4w)</code> <code>\$date(actualMonth -1d)</code>
<code>\$date(dateString, format)</code>	Works like <code>\$date(dateString)</code> , but output format is controlled by <code>format</code> parameter	<code>\$date(now + 5d -4w, dd-MM-yyyy)</code> <code>\$date(2011-01-01 + 5d -4w, MMyyyy)</code> <code>\$date(actualMonth -1d, yyyy-dd-MM)</code>
<code>\$time(timeString)</code>	Evaluates the <code>timeString</code> and outputs the resulting time in <code>HH:mm:ss</code> format	<code>\$time(now)</code> <code>\$time(actualhour)</code> <code>\$time(actualminute)</code>
<code>\$time(timeString, format)</code>	Works like <code>\$time(timeString)</code> , but output format is controlled by <code>format</code> parameter	<code>\$time(now, HH:mm:ss)</code> <code>\$time(actualhour, MMss)</code> <code>\$time(actualminute, HHmmss)</code>
<code>\$timestamp()</code>	Returns the current time stamp value	<code>\$timestamp()</code>

### *DateString* and *TimeString*

#### DateString

- now - represents actual date
- actualyear - represents the first day of actual year (1.1.20XX). For example actualyear selected on 21.9.2010 represents date 1.1.2010
- actualquarter - represents the first day of actual quarter (1.1.20XX, 1.4.20XX, 1.7.20XX, 1.10.20XX). For example actualquarter selected on 21.9.2010 represents date 1.7.2010

- actualmonth - represents the first day of actual month (1.1.20XX, 1.2.20XX, ...). For example actualmonth selected in 21.9.2010 represents date 1.9.2010
- actualweek - represents first day of actual week (Monday). For example actualweek selected on 21.9.2010 represents date 20.9.2010 (Monday of this week in calendar)
- relative and absolute enterig of date can be adjusted by operators using this syntax: date +/- n[d|w|m|q|y], where n is integer, d represents day, w represents week, m represents month q represents quartal and y represents year. We can for example define time in this way: actualyear + 2m -4d. Today is 21.9.2010, so this value represents 1.1.2010 + 2 months - 4 days, which means date 25.2.2010.

#### TimeString

- now - represents actual time
- actualhour - represents the actual hour at 0 minutes and 0 seconds.
- actualminute - represents the actual minute at 0 seconds
- actualsecond - represents the actual second

#### Connecting to SOAP web Services

BellaDati is able to connect resources available on network via the HTTP protocol. Except this simple usage, we can connect also more complex resources available as Web services. Web services are using the SOAP protocol, which is based on the plain HTTP protocol. The SOAP message comes in standard XML format, which is in BellaDati perfectly supported. Here is an example how to do it:

1. Enter the endpoint URL of your web service and choose the [XML](#) file format.
2. Open the advanced settings, choose the POST method and set the following parameters:
  - a. SOAPAction - value is contained in the WSDL file, which describes your web service. It is defined in the soapAction tag, e.g.:  
`<soap:operation soapAction="http://www.sap.com/Z_HSI_HRP04_RZH_READ_DATA"/>`
  - b. Content-Type - set the value to text/xml
3. Insert the POST content in the depicted structure:

## Set data source URL

Enter the URL address you want to load data from. Then select the file type. Section Advanced settings allows you to specify parameters of connection to Web Services (HTTP Post content is used to configure WSDL/SOAP). See documentation for details.

URL	<input type="text" value="http://demo.belladati.com/Z_HSI_HRP04_RZH_READ_DATAPortType/HRP04MockService"/>
File content	<input type="text" value="XML file"/>

### [Advanced settings...](#)

#### Authentication settings

The login data entered are stored encrypted, not shared with any other application and used only to access selected service.

Login	<input type="text"/>
Password	<input type="text"/>

#### HTTP headers settings

HTTP method type	<input type="text" value="Post"/>	
Name	Value	
<input type="text" value="SOAPAction"/>	<input type="text" value="http://www.sap.com/Z_HSI_HRP04_RZH_READ_"/>	<a href="#">Remove</a>
<input type="text" value="Content-Type"/>	<input type="text" value="text/xml"/>	<a href="#">Remove</a>
<a href="#">+</a> <a href="#">Insert</a>		

#### HTTP Post content


```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="urn:sap-com:document:sap:rfc:functions">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:Z_HSI_HRP04_RZH_READ_DATA>
      <!--You may enter the following 5 items in any order-->
      <!--Optional:-->
      <PCV_LAST_PERNR?></PCV_LAST_PERNR>
      <!--Optional:-->
      <PIV_FIRST_PERNR?></PIV_FIRST_PERNR>
    </urn:Z_HSI_HRP04_RZH_READ_DATA>
  </soapenv:Body>
</soapenv:Envelope>
```

[Continue](#)

## Authentication

The following authentication methods are supported:

- Basic and Digest HTTP Access Authentication ([RFC2617 standard](#)).

 REST API supports OAuth standard with security token.

## Next Steps

- Take me back to [data sources](#)
- [Import settings](#)
- [Scheduling Import](#)

## Video tutorial

[Connection to external data source](#)

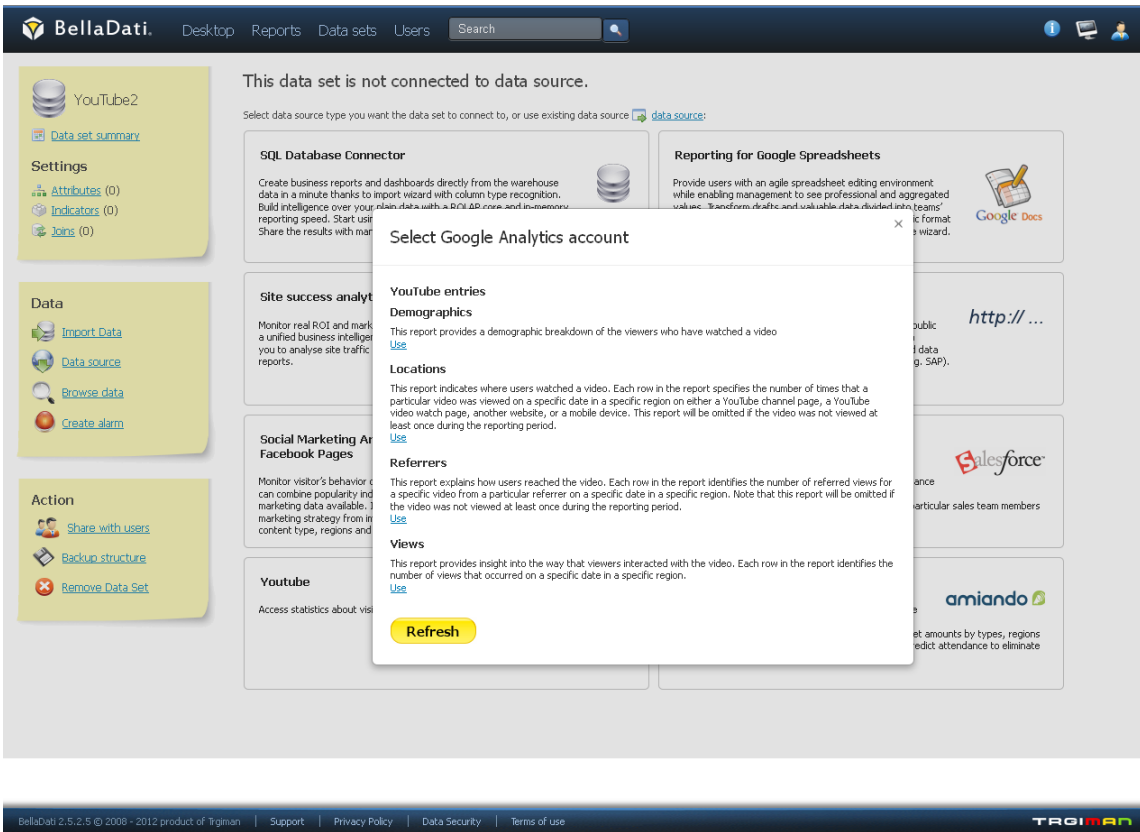
---

[How to analyze data from World Bank](#)

---

## Connecting to YouTube

BellaDati provides connector to YouTube.



## Authentication

You have to grant BellaDati access to your data on YouTube first. Please log in using your Google account credentials.



The login data you entered are stored encrypted and not shared with any other application. This login and password is used only to enable data transfer from secured Google API and is not allowed to extract any other personal information from your Google Account.

## YouTube Entries Selection

Now select one of the YouTube entries (data areas):

- **Demographics:** This report provides a demographic breakdown of the viewers who have watched a video.
- **Locations:** This report indicates where users watched a video. Each row in the report specifies the number of times that a particular video was viewed on a specific date in a specific region on either a YouTube channel page, a YouTube video watch page, another website, or a mobile device. This report will be omitted if the video was not viewed at least once during the reporting period.
- **Referrers:** This report explains how users reached the video. Each row in the report identifies the number of referred views for a specific video from a particular referrer on a specific date in a specific region. Note that this report will be omitted if the video was not viewed at least once during the reporting period.
- **Views:** This report provides insight into the way that viewers interacted with the video. Each row in the report identifies the number of views that occurred on a specific date in a specific region.

Then continue [Import settings](#).

## Modifications

The following parameters and actions can be performed for the existing YouTube data source:

- Revoke access to Google services

## Next Steps



- Take me back to [data sources](#)
- [Import settings](#)
- [Scheduling Import](#)

## Scheduling Import

BellaDati's integrated scheduler allows you to control, monitor and schedule automatic import from [data sources](#). The overview is represented by a navigable calendar.

The screenshot shows the BellaDati web interface. At the top, there is a navigation bar with 'BellaDati', 'Desktop', 'Reports', 'Data sets', 'Users', and a search bar. Below this, a sidebar on the left contains sections for 'SalesForce', 'Settings', 'Data', and 'Action'. The main content area displays a 'Data set summary - Salesforce' with a 'Create report' button. Below the summary is a calendar for January 2012. A 'Schedule execution' dialog box is open, showing the following configuration:

- Run when: 1/26/2012 18:00:00
- Overwrite existing data:
- Overwrite data according rule: Delete data with the same attributes values and time
- Repeating interval: Week

The calendar shows a grid of dates from 23/1 to 31/1. The 26th of January is highlighted with a green circle and a clock icon, indicating a scheduled import. A tooltip for the 26th shows the time 16:02 and a cancel icon (X).

Actions available:

- Schedule import
  - Set the exact time of automated execution
  - Configure the [data overwriting policy](#) if required
  - If you need the import to be repeated after specific period, configure the repeating interval (day, week, month, quarter, year). Leave it empty, to execute the import once.
- Tooltips actions (only for already finished imports):
  - [Browse imported data](#)\*
  - Display the [import results](#) summary: Useful for import statistic or problems diagnostics.
  - Delete imported data: Deletes all data created within the particular import.
  - Cancel import (cross symbol): You can cancel future scheduled imports.

Additional tooltip information for executed imports:

- Download the original file
- Execution date and time
- Status: Completed, Running
- Overwrite existing data: yes, no
- Scheduled by: author name



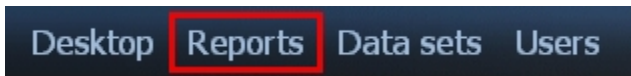
1. Synchronizations can be scheduled only in the future. If no scheduling action is available, please verify whether the calendar displays the right period.
2. Calendar shows the last month where an import has been performed. If no import has occurred yet, the month with the nearest scheduled import will appear. Otherwise actual month is displayed.

## Next Steps

- Configuring the [import settings](#)
- Checking the [import results](#)
- Managing the [data sources](#)
- [Browsing, editing and exporting data](#)
- [Creating Report](#)

## Reports

This chapter is focused on reports and analytical activities. You need to have "Report Editor" role assigned by your domain administrator if you want to prepare or edit existing reports. Features, described in this chapter, are available via "Report" button in upper blue application menu.



### 2.3.1 Report management

#### 2.3.2 Table settings

#### 2.3.3 Chart settings

#### 2.3.4 Indicator settings

#### 2.3.5 Time settings

#### 2.3.6 Analytical filters

#### 2.3.7 Drill-down path

#### 2.3.8 Cooperation and sharing

#### 2.3.9 Custom content and attachments

#### 2.3.10 Exports

#### 2.3.11 Formulas

- [Formula Reference Guide](#)
- [Report Variables](#)














### 2.3.1 Report management

- [Report Search and Management](#)
- [Navigation Dashlets on Workspace](#)
- [Tags](#)

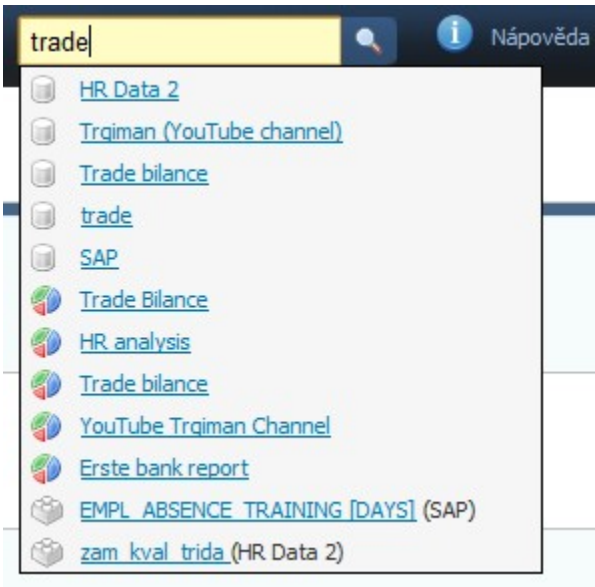
## Report Search and Management

You can find a list of reports for logged user displayed on the reports page (accessible using link "Reports" in blue top main menu). This brief list might contain more pages. Within each item, information about source data set, owner, recent changes, views and report description are displayed. You can reach particular report, data set or user using according links.

### Search

 <a href="#">O2 Enterprise invoices - phone calls</a> Enterprise phone calls in June. Datasource: O2  25.08.2010 15:06:47 changed by  <a href="#">Trqiman_Support</a>	Source data set:  O2 Invoices Owner:  <a href="#">Trqiman_Support</a>
 <a href="#">Copy - Phone Calls Report</a> Get a clear view how money are spend on phones by employees. In what day tariff calls are performed. Discover if you are using more of GPRS, UMTS or other kind of services and how much does it cost.	Source data set:  CallCenterInvoices Owner:  <a href="#">Martin Trqiňa</a>
 <a href="#">CAS PIA phone calls</a>  27.09.2010 15:29:01 changed by  <a href="#">Trqiman_Support</a>	Source data set:  CAS PIA phone calls Owner:  <a href="#">Trqiman_Support</a>

It's possible to search reports in two different ways. The first possibility is to use searching window in the top blue menu. It's accessible from all parts of the application. It is designed to find not only reports, but also data sets, indicators and attributes, which contain entered text. Those entities can be differentiated by the according symbol next to searching results.



The second possibility is to use more detailed search engine, which is situated left next to the list of reports. You can specify your own searching filters here. In filters you can specify what kind of entity you want to get (data set, report, indicator, attribute) and also your access rights for those entities. It's possible to choose one or more choices in each list (by holding CTRL and clicking on the chosen list items).

**Filter settings**

**Search text**  
calls

**Search in**  
Reports  
Data sets  
Indicators  
Attributes

**Access rights**  
My  
Shared  
In responsibility

Show

Meanings of the access rights categories are following:

- My - displays a list of entities which are owned by the current user
- In responsibility - shows entities that were assigned to current user by another user to share them with access rights to modify these reports.
- Shared - shows entities shared with the current user by another user (without an access right to modify them).

If you search some report or data set really often, you can save (by the button above) and reuse again your searching filter. You can also browse the list of filters, which displays number of found entities next to each particular filter name.

**Saved filters**

[Calls searching](#) 3

[Profit results](#) 3

[+ New filter](#)

## Navigation Dashlets on Workspace

Specialized dashlets can be added to the Workspace. These dashlets serve as better navigation among vast number of reports. These dashlets can be found within category "Others" in "Insert dashlet" menu on Workspaces page. There are following dashlet types:

- Most visited - displays the most accessed reports. The larger the number of accesses, the bigger font is displayed (tagcloud).
- Owned reports - all reports logged user is author of. These matches "My" category on reports overview page.
- Shared reports - all shared reports without access rights to modify them. These matches "Shared" category on reports overview page.

See chapter 2.4.2 for more information about using workspace and dashlets.

## Tags

Tags serve for better and faster navigation among your reports. Each of your reports is tagable. You may insert tags by clicking on the field below the name of the report.

## Business overview

More...

Click to insert a description

Tags: Click to insert tags

Data set: business data



Tags should reflect some basic characteristics or content of particular reports. Search engine (available in top blue menu) will find all reports which are marked by selected tags. These tags are also displayed in the list of reports below the report name. If you click on some tag, all other reports marked with the same tag will be displayed.

**Financial analysis**  
Tags: [Profit](#) [Costs](#) [ROI](#) [Invoices](#) [Plans](#)  
04.03.2011 12:27:09 changed by [Denis Neumann](#)

### 2.3.2 Table settings

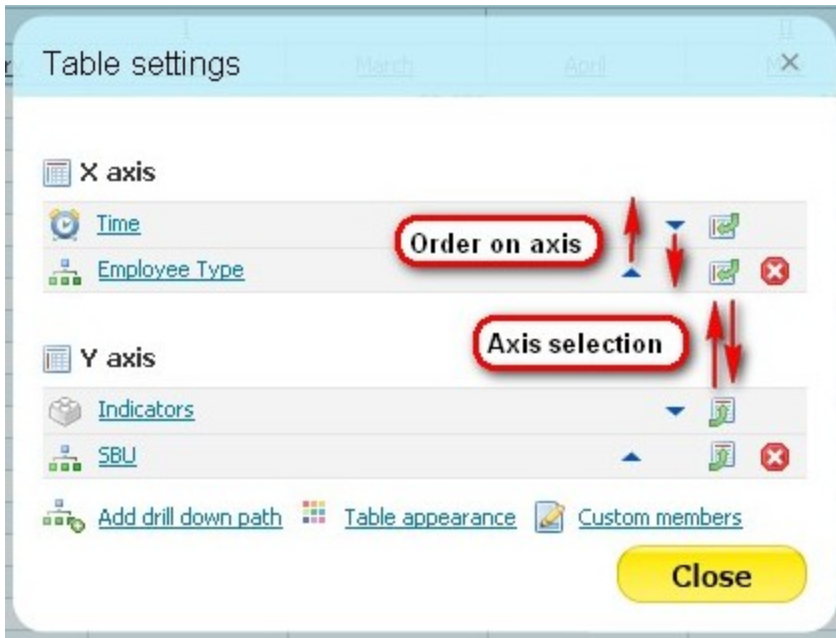
- [Table View Settings](#)
- [Time Defined by Formula](#)
- [More Time Intervals in Single Table](#)
- [Member Sorting According to Indicator Values](#)
- [Alphabetical Member Sorting](#)
- [Custom Members](#)
- [Blank Members](#)
- [Reverse time interval](#)
- [Copy Table](#)
- [Shifting of Aggregation Level](#)
- [More Advanced Table Settings](#)
- [Ad-hoc charts](#)
- [Video tutorials](#)

#### Table View Settings

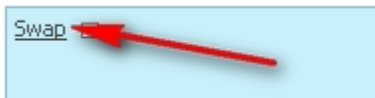
Tables in BellaDati are very flexible when taking in account versatile possibilities of how to present data. The most simple table content consists of one indicator and a single time interval (one day at least). Tables naturally allow displaying more time intervals, indicators and even drill-down paths. All these options can be combined together.

TIP: Despite there are no particular restrictions of elements count, it is recommended to keep their count as low as possible to ensure table lucidity.

More drill-down paths can be added into the table by clicking on "Add drill down path" in Table settings popup window. To add time intervals, use "Time" link. More indicators can be added by clicking on "Indicators" link. You can change the order of individual elements between X and Y axes using the appropriate symbol in Table settings window (see fig.). Simultaneously the order of these elements on each axis can be modified by clicking small blue arrow icons in the same popup.



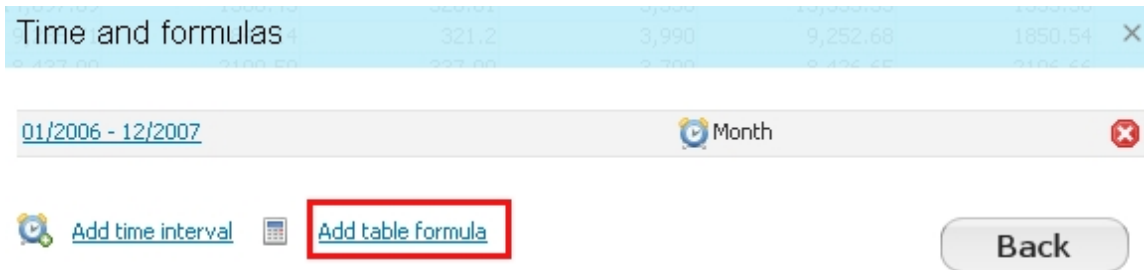
This procedure leads to arbitrary table depiction, which is detailed sufficiently and well-arranged reasonably at the same time from user's point of view and clarity of indicators' values.



If you just want to quickly change elements on horizontal and vertical axis, click on the "Swap" in the upper left corner of the table.

### Time Defined by Formula

In addition to mentioned ways, how to define time in the table, it's also possible to define time dimension of table by formulas. This function is available in table settings window.



There you can use whole family of time formulas. One of the most useful function is "value at time". This function displays value of selected indicator in the specified day. Second useful time function is "Aggregated value at time", which allows to specify the interval of displayed indicator values. More information about formulas can be found in the chapter about formulas.

**Add table formula** [X]

Name:

Formula: `value('2010-01-01','2010-02-28','M_PROFIT')`

Insert to formula: **Time function** [v]

- Previous value
- Previous value with level aggregation
- Next value
- Next value with level aggregation
- Value at custom time
- Value at custom time with level aggregation
- Aggregated value at custom time**

### More Time Intervals in Single Table

According to introduction of this chapter, the table accepts more time intervals. Every interval can be of different type (absolute, relative, custom), eventually values of selected period could be merged into single column. The order of displaying these intervals can be modified in time settings popup using arrows icon.

**Intervals in view** [X]

Granularity

- January to May
- Sum (one column)

Month [v] [X]

Day [v] [X]

**Add interval** [i] [Back]


Thanks to options described above, it is possible to reach sophisticated depiction of the table. Therefore the table might contain an absolute interval displaying indicators' values for each periods (months, years etc.) as well as the last column which shows the same interval but aggregated to single value for each indicator. Configuring the table this way enables user to analyze continuous values in connection with total (or average, minimal, maximal) value in single table for the same period.

		January to May						Sum (one column)
		2010						
		I			II			
		January	February	March	April	May		
Total [USD]	Income	316,497.93	0.69	45,236.69	534,362.00	34,704.00	982,801.31	
	Outcome	20,593.00	31,471.00	87,889.00	260,000.00	27,984.00	537,937.00	

### Member Sorting According to Indicator Values

Members of displayed attributes are sorted alphabetically at default. When the user places the indicator on the bottom part of the horizontal axis, it is possible to change members' order by clicking on the small arrow icon nearby indicator's name. This function activates ascending or

descending sorting of indicator values in the selected column. On the basis of the actual sorting the analyst acquires fast overview about indicators' values difference among individual attribute members within particular time unit. The members that contain no values within selected time unit, are always displayed above or beneath the sorted values.

	II
	May
	Total [USD] 
+ Production	10,000.00
+ Invoice	16,412.00
+ Receipt Card	17,160.00
+ Requisition	573,548.00
+ Issued Goods	

### Alphabetical Member Sorting

Alphabetical sorting is complement function to indicator values sorting. It allows user to sort members ascending (which is also default sorting for tables) or descending by the member names. If you want to do this, click on the arrow in square in upper left corner of the table. This small arrow shows the direction of sorting. This symbol is displayed only in case that table contains at least one attribute on vertical axis.

Swap 	
	Albania
	Angola
	Anguilla
	Antigua and Barbuda
	Argentina
	Armenia
	Aruba
	Australia
	Austria
	Azerbaijan
	Bahamas



### Custom Members

Table usually displays only members of attributes, which are contained in according data. In addition you can create your own members, which could be displayed in table under regular members. You can create your own member using the "custom member" link in the table settings window.



Table settings 200 ✕

**X axis**

- Time and formulas
- Indicators

Swap

**Y axis**

- Perrson
- Add drill down path
- Table appearance
- Custom members**

**Close**

Now, you have to specify the name of new member and attribute, in which will be new member displayed. Then it's necessary to define, what should new member consist of. You can choose members from all available attributes and freely combine them (like in the case of member filtering). It's possible to choose more than one member from each attribute. Values of those members will be summarized in the new member (you have to separate particular members of one attribute by semicolons). In our example we want to summarize values for "TV" and "Recorder" of "John Newman" and display this new member in "Person" attribute.

### Select associated attribute

Select attribute you want to add custom member for

Perrson ▼

### Add custom member

Name

**Criteria**

Attribute	Semicolon-separated attribute values
Perrson	<input type="text" value="John Newman"/>
Product	<input type="text" value="TV;Recorder"/>
Warehouse	<input type="text"/>

Choosing "John Newman" in person attribute causes displaying of values, which are connected only to this person (for case, that "recorder" and "TV" would be connected also with another person). The final table with custom member should look like this:

Swap		2010
		IV
		December
		Price
John Newman		200
Warehouse A		168
Camera		50
Monitor		8
TV		110
Warehouse B		32
DVD		12
Recorder		20
TV+recorder		130

Note: custom members are always displayed as the last members in the table. They can't be sorted by value with other regular members. If you want to change a sequence of custom members, use blue arrows in the custom members settings.

### Blank Members

It sometimes happens that some attribute contains a few blank members in source data. But we still need to know the value of indicators in the rows containing these blank members. In our example we have some blank members in attribute "Affiliate".

Affiliate	Employee	Identifier	Income
		10SP00003	4990.0
Prague		10SP00003	4000.0
		10SP00001	900.0
Pilsen		10SP00001	1000.0
		10SP00003	6400.0
Berlin	Robert Glaser	10SP00002	150.0
Prague	Robert Glaser	10SP00002	2670.0
Paris	Saskia Roth	10SP00001	160.0
Prague	Robert Glaser	10SP00003	2500.0
Berlin	Robert Glaser	10SP00003	2500.0
Prague	Robert Glaser	10SP00002	350.0
Paris	Robert Glaser	10SP00001	85.0
Berlin	Robert Glaser	10SP00003	1540.0
Berlin	Robert Glaser	10SP00002	1540.0

blank members of attribute "Affiliate"

Look at the same situation in report table. Table view for this data contains additional member "(blank)" which aggregates and displays values of all the blank members of Affiliate.

Swap		2010
		I
		January
		01
Income	+ Berlin	7,370
	+ Paris	1,445
	+ Pilsen	3,500
	+ Prague	10,510
	+ {blank}	12,290

If we want to remove this member from the table, we just choose condition "has value" in filter of "Affiliate" attribute.

**Attributes**

Affiliate	has value	▼
Customer		▼
Employee		▼
Identifier		▼
Product Code		▼

**Reverse time interval**

Sometimes we need to display table values in reverse orientation of time axis, especially in case that we are interested in the most actual values of indicators (for example during using of periodical imports from external data sources). All you need is just to easily click on selected time interval, open its settings and click on "reverse time interval". In case that table contains more intervals, only the selected time interval will be reverted. Reverse time interval is possible to display in any usual time unit.

	2010		
	I		
	March	February	January
Income	25,338.42	39,857.77	93,356

**Copy Table**

If you want to insert a new table with the same settings from some existing table in the same report, you don't have to create it from the very beginning. It's enough to click on the copy icon over the particular table and then will be immediately created a clone of this table (including filter settings, time, drill-down paths and other dimensions).



**Shifting of Aggregation Level**

In table, which contains more attributes in drill-down path, is possible to shift drill-down level of displayed indicator values. Indicator values of attribute, which is inferior in drill-down path, can display aggregated values of superior attributes in drill-down path. This function is useful in cases, where we want to compare individual parts of the complement. Look at table below - it shows regular indicator and than the same indicator with one shifted level of aggregation.

Swap	2010	
	IV	
	December	
	Price	Price -1 level
John Newman	200	200
Warehouse A	168	200
TV	110	168
Camera	50	168
Monitor	8	168
Warehouse B	32	200
Recorder	20	32
DVD	12	32

If you want to create indicator with shifted level of aggregation, use the formula interface. Choose "Indicators" as the formula category and select the appropriate indicator with "aggregated value" in brackets. The number in formula determines the level of aggregation shifting (default = 1). This function is also at disposal in a lot of time functions.

Indicators settings

Name **Price -1 level**

Formula `value('M_PRICE',1)`

Insert to formula

Indicator

- Price (Sum)
- Price (Min)
- Price (Max)
- Price (Avg)
- Price (DistinctCount)
- Price (Aggregated value)

### More Advanced Table Settings

All options regarding indicators are described in chapter 2.3.4 Indicator settings.

Description of all options referred to drill-down path can be found in chapter 2.3.7 Drill-down path.

### Ad-hoc charts

If you need a chart view of the tabular data, just click on attribute member a choose the "quick chart" function. In the following example we are analyzing visitors of a web page drilled-down by continents. Using the ad-hoc chart we are able to display this data for example for the Europe:

Počet přístupů podle zemí

	2011							Rok 2011 2011
	II		III		IV			
	červen	červenec	srpen	září	říjen	listopad		
Africa	7	3	1		5	3	33	
Americas	96	51	67	93	81	70	830	
Asia	49	30	31	42	27	39	367	
Europe			203	486	762	1 463	6 001	
Oceania			3	10	11	1	67	
ostatní			1	2	5	3	22	

Možnosti větvení

Překlad a vzhled větvení

Rychlý graf

Ad-hoc chart is displayed in a movable popup window. You are also able to dynamically change the chart type (the default chart is a bar chart).



## Video tutorials

# Video-tutorials

Table view settings

---

Conditional formatting

---

Custom members in table

---

Drill-down path in report

---

## 2.3.3 Chart settings

- Suitable Usage of Different Chart Types
- Charts Settings and Modification
- Appearance and color themes
- More Indicators in Single Chart
- Negative and Empty Values Interpretation

- Copy Chart
- Grouping in pie charts
- HTML5 and SVG charts (beta)
- Adjustable scale of line charts
- Video tutorials

## Suitable Usage of Different Chart Types

In BellaDati every chart type allows the analyst to display data observed in many different ways. Therefore we recommend the user to become more familiar with logical options that particular chart types can display.

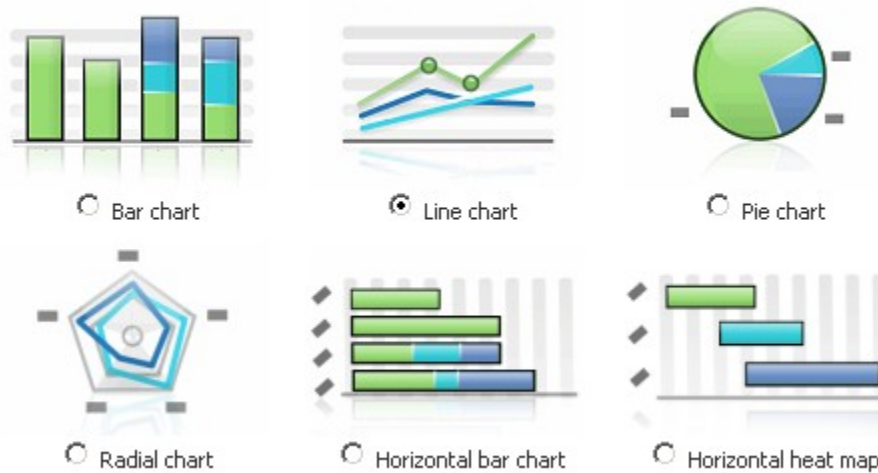


Chart name	Depiction without drill-down path	Depiction with selected drill-down path
Bar chart	total amount of the indicator in every time unit (displayed vertically)	relative distribution of the value of the indicator among individual members of selected drill-down path in every time unit (displayed vertically)
Line chart	(joined by a curve) the total amount progression of the indicator in time	(joined by a curve) the progression of the indicator for each drill-down member in time
Pie chart	indicator distribution among the selected time units (according the size of the pie wedge)	indicator distribution among the individual drill-down members for the whole time interval (according the size of the pie wedge)
Radial chart	indicator distribution among the selected time units (according the ray length)	indicator distribution among the individual drill-down members for the whole time interval (according the ray length)
Horizontal bar chart	total amount of the indicator in every time unit (displayed horizontally)	the amount of the indicator for each drill-down member for the whole time interval (displayed horizontally)
Horizontal heat map	total amount of the indicator progression in time (using color tone)	the indicator amount progression for each drill-down member in time (using color tone)

Charts, that allow to observe the progression of indicator value in time, is possible to use in analyses, where the user is interested in continuous value changes. On the other hand charts summarizing the whole time interval in single depiction are suitable to get an overview about total indicators' value. Radial and pie chart demonstrate the same logical options (indicator value distribution) however using another type of the depiction. The radial chart might be more well-arranged in case of a lot of drill-down members present. The pie chart is more suitable for comparison of negative indicator values. Horizontal heat map allows analyst to discover negative values by displaying the lightest color tone for each drill-down member (in contrast of discolored missing values).

## Charts Settings and Modification

Similarly as in the pivot tables alot of parametres can be configured in each chart. User is able to modify drill-down path depiction, indicators

observed, the time interval and chart appearance. The rules for the time interval settings are the same as for the pivot tables (described in chapter 2.3.2). However the option "show in one column" is missing for charts. The process of indicators' configuration is almost the same too (see chapter 2.3.4). On the other hand the depiction of particular indicators and conditional formatting is not possible to change in charts.



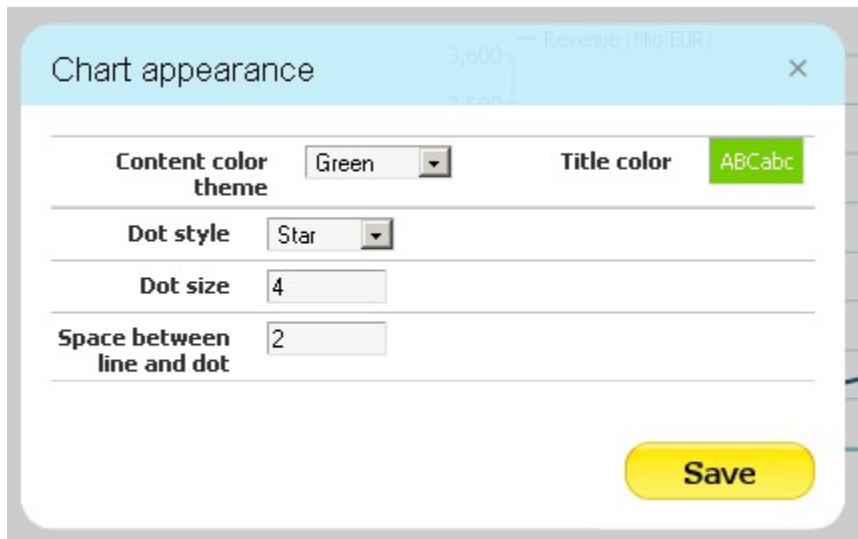
Settings regarding drill-down are significantly simpler than by pivot tables since charts do not display the whole drill-down path (drill-down function is not used here), but depict the total indicators level only (regardless of the drill-down path), or the members of single - currently selected drill-down attribute are displayed. This attribute can be chosen directly from the list available by clicking on "Drill-down path" link in the Charts Settings popup window (see above).

### Appearance and color themes

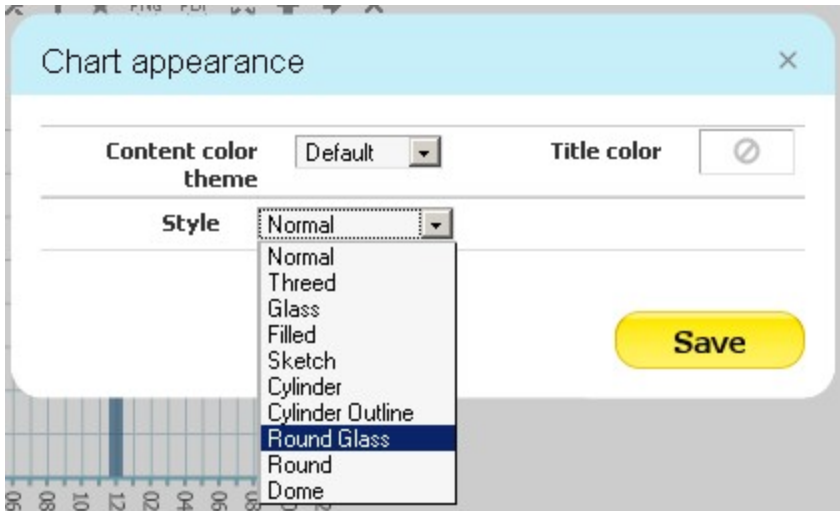
Graphic depiction of the charts can be modified using "Chart appearance" link.



There you are able to choose one of seven prearranged color themes that contain fair and lucid color combinations of sufficient contrast. It is also possible to modify the header color which will then propagate into dashlet moved to the workspace. Both color settings are possible to configure by pivot tables too. In addition line and radar chart allows the user to select the type of the joint point (dot, circle, triangle, no point etc.), modify their size and change the space width between the point and the line. The user can achieve individual appearance of the final charts exactly according to his idea using these settings.

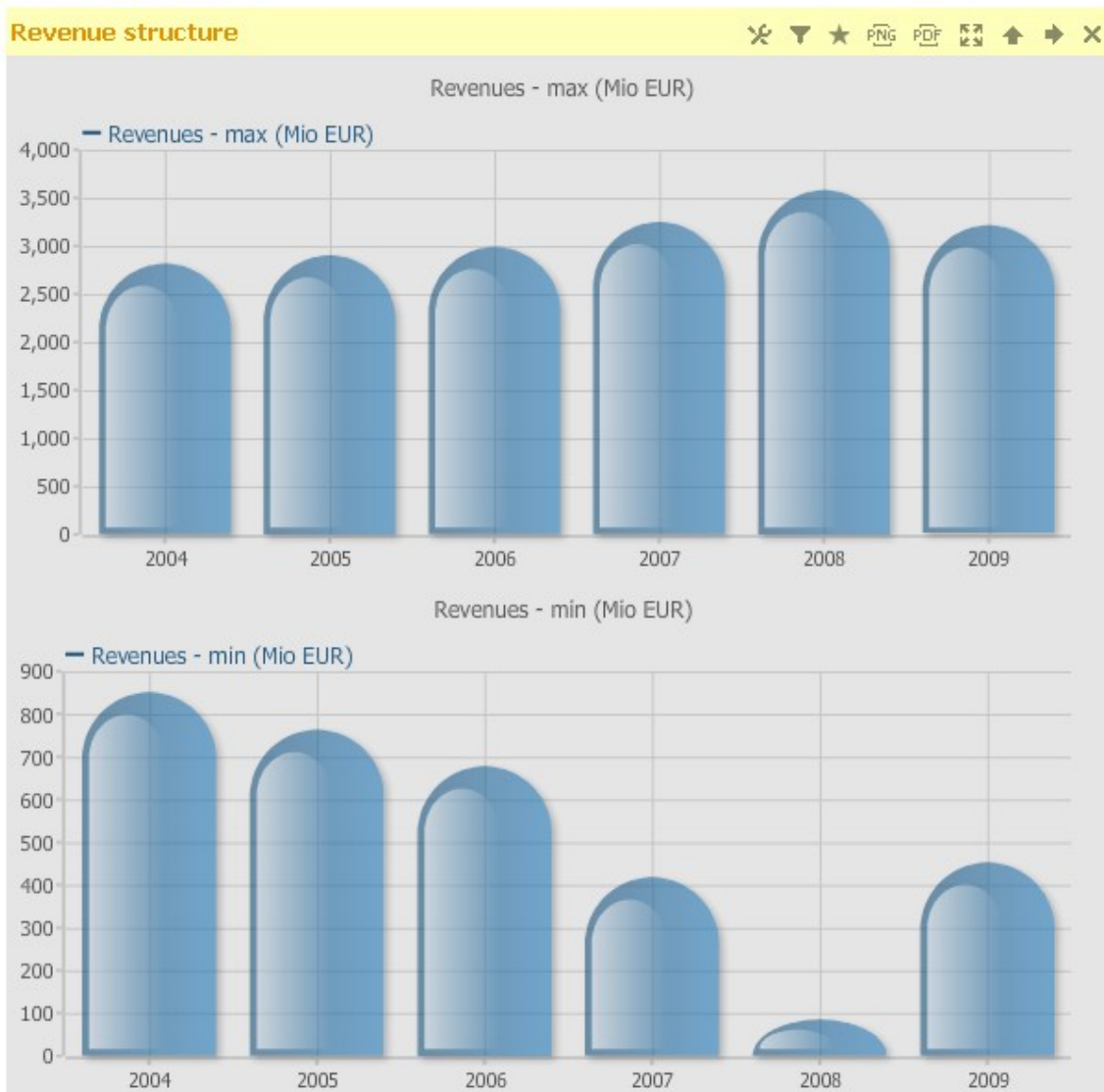


In the appearance point of view the (vertical) bar chart is a special case. The user can choose specific graphic style of columns displayed there. There are many different styles available - starting with 3D and oval columns, ending with less formal charts appearance. It is important that these alternative chart appearance can be selected only if there is no drill-down path selected.







### More Indicators in Single Chart

By default more indicators in single view are depicted in separate charts one below another. The particular indicator's name is displayed in the header of every chart. The final appearance of the bar chart with alternative columns' styles and more indicators can be displayed as follows:






When the user wants to display more indicators in a single chart, it is necessary to set "Display in single chart" checkbox in the Chart Settings main popup. This option is available only if the view contains more than a single indicator.

 <a href="#">Drill down path</a>	Not set.
 <a href="#">Time interval</a>	2004 - 2009
 <a href="#">Indicators</a>	<input type="checkbox"/> Display in single chart
 <a href="#">Chart appearance</a>	

After updating the view more indicators will appear in the single chart. These indicators are then differentiated by color (see fig.). More indicators' depiction in a single chart is solved specifically in every chart type.

 TIP: We recommend user to consider whether to display more indicators in a single chart in cases when attribute containing many drill-down members is selected. Such charts might not be well-arranged enough for the user and splitting the indicators into more charts and/or remove selected drill-down attribute could be better solution.



Two indicators in single bar chart

## Negative and Empty Values Interpretation

It's usually important to interpret negative (or empty) values in different ways. Sometimes it necessary to display negative values, sometimes we want to see only absolute values of the indicator and sometimes we need to even skip all the negative values. Therefore you can choose this interpretation in the indicator settings (only in charts). Selected rule will be used for all the values lower than zero:

- don't transform - remain all the negative values (default)
- absolute value - all the negative values are multiplied by (-1)
- skip - won't show negative values in chart at all
- to specified value - change all the negative values with one specified constant

Indicators settings
✕

---

Name

---

Color

---

Unit

---

Format

[Show help](#)

---

Rounding

Classic (half up) ▾

---

Members aggregation

Sum ▾

[Show help](#)

---

Time aggregation

Sum ▾

[Show help](#)

---

Transform negative values

Absolute value ▾

---

Transform also empty values

Value:

---

In the same way it's possible to specify your own constant for empty values. This constant will be displayed in each time unit, in which selected indicator doesn't show any values. This function is available even if no transformation of negative values is selected.

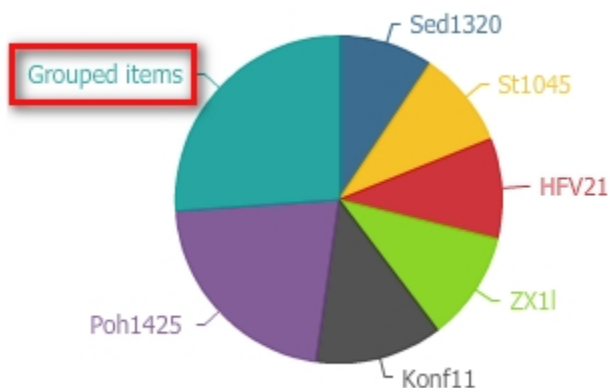
## Copy Chart

If you want to insert a new chart with the same settings from some existing chart in the same report, you don't have to create it from the very beginning. It's enough to click on the copy icon over the particular chart and then will be immediately created a clone of this chart (including filter settings, time and other adjustments).



## Grouping in pie charts

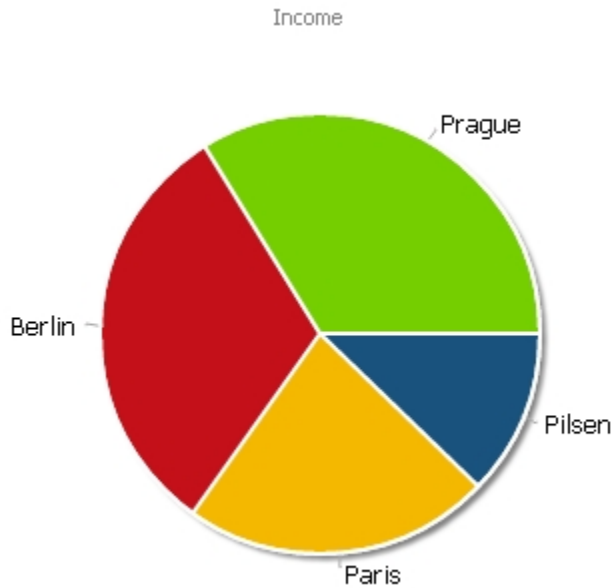
Pie charts has an ability to group low values parts of pie chart into one aggregated segment. It improves pie chart's ability to compare the most significant members/time units, because the less significant members/time units are hidden in standalone section of chart. All the members/time units with indicator values lower than 5 percent are grouped. Of course you can manually increase or decrease this limit according to your needs in pie chart appearance settings.



## HTML5 and SVG charts (beta)

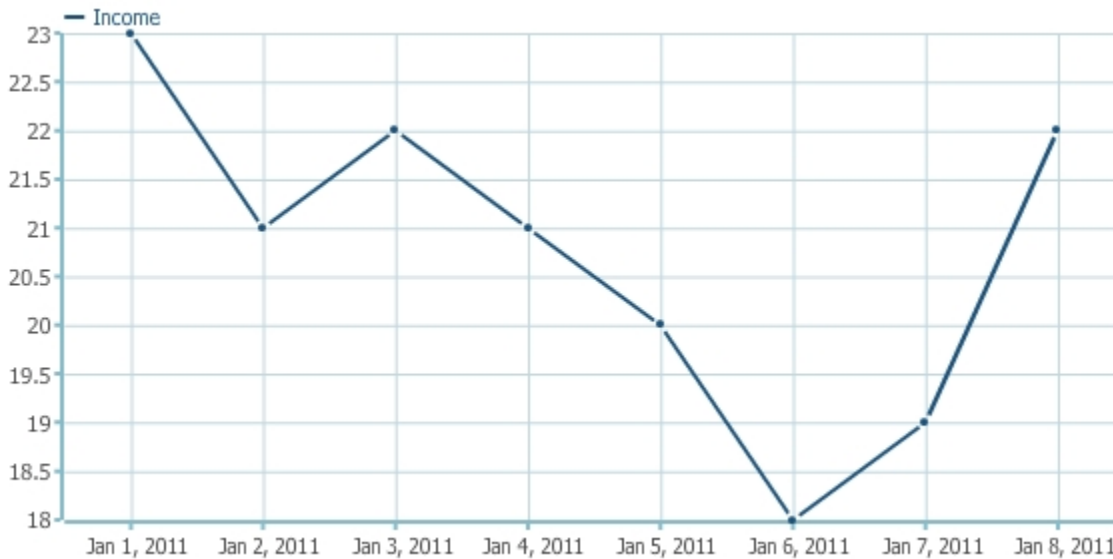
In a default settings all the charts in BellaDati are displayed by the Flash technology. This technology is very good but there are some limitations - it is not possible to display Flash charts on mobile devices like iPad or iPhone, that don't support the Flash technology. Therefore it's possible to

switch the chart technology to HTML5 or SVG in the user profile. These charts are released as a beta version now, because they are not fully supported by some browsers (e.g. Internet Explorer 8 and others) and charts (e.g. heat map). Some of charts may look different compared to basic Flash charts.

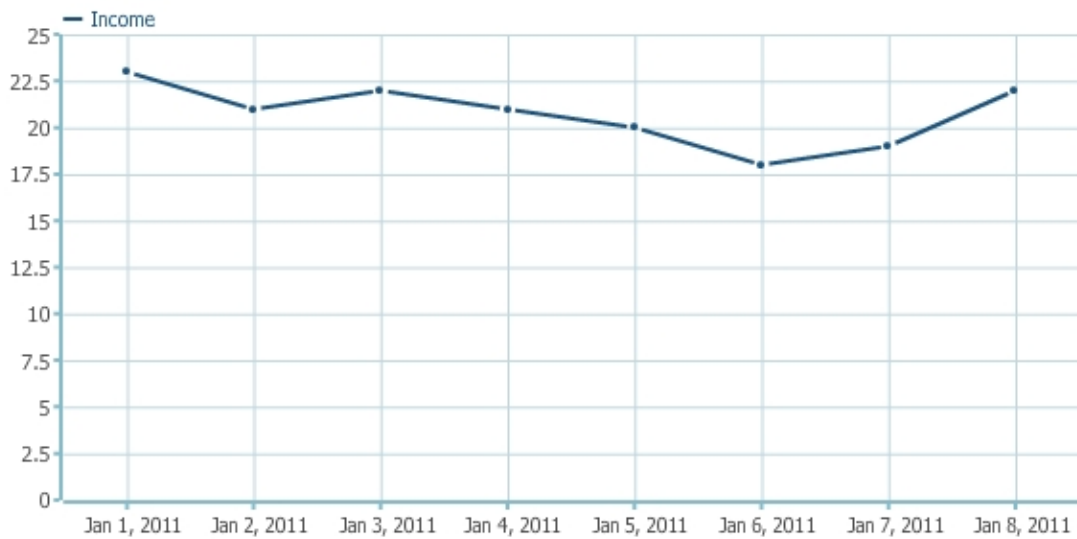


### Adjustable scale of line charts

Value axis in the line chart is automatically generated by the actual values from source data. This is a useful feature but sometimes user needs to display data in chart in his own scale - for example to display axis with the zero origin. Therefore there is a feature that allows to specify minimal and maximal value of the axis in case of line chart. This feature is available in the appearance settings. For example: displayed values of indicator are between 18 and 23. But our aim is to display values on the scale with zero origin and 25 as a maximal value, because we want to compare more charts in the same scale. So we just open line chart appearance settings and enter 0 as minimal value and 25 as a maximal value.



Line chart with default settings



Line chart after change of the scale

## Video tutorials

# Video-tutorials

### Chart appearance

---

### 2.3.4 Indicator settings

- Displaying and editing indicators
- Appearance and Conditional Formatting
- Indicators Duplication
- Indicator Groups
- Indicator's Color Change in Charts
- Count and Unique Count
- Cumulative Indicators

- [Video tutorials](#)

## Displaying and editing indicators

A lot of parameters can be set regarding indicators used in the reports. There is possible to edit following settings by indicators which were created during the import:

- name
- decimal format
- rounding
- unit
- members aggregation
- time aggregation

By indicators created using formulas can the user change its appearance, however it is not possible to modify time and members aggregation. Formulas are described separately in chapter 2.2.4, decimal format options description can be found in chapter 2.2.3.

The screenshot shows the 'Indicators' dialog box with three main sections:

- Available indicators:** Lists 'Costs' and 'ROI'.
- Indicators displayed in the view:** Lists 'New indicator', 'New indicator group', 'Group 1', 'Income [EUR]', 'Sub-group 2', and 'Constant'. A red arrow points from 'Income [EUR]' to the 'Selected indicator details' section.
- Selected indicator details:** A table showing settings for the 'Income' indicator. A red box highlights the 'Indicator setting...', 'Appearance settings...', and 'Duplicate' options.

Selected indicator details	
Name	Income
Unit	EUR
Format	###, #
Rounding	Always up
Members aggregation	Sum
Time aggregation	Sum

[Indicator setting...](#)  
[Appearance settings...](#)  
[Duplicate](#)

**OK**

Time aggregation determines, how the values of every time unit will be displayed to the user in case of these time units contain more data in lower time granularity. Eg. when the data of some indicator is stored by days and values of this indicator are displayed by months in the table, then the current time aggregation determines, how we call for displaying the monthly amount of values of every days of selected indicator - these can be counted using following rules:

- sum
- average
- minimum
- maximum
- distinct values count

The default time and members aggregation setting is sum.

Members aggregation influences, how the amount of the indicator of every attribute is calculated in each time unit. If any member of drill-down path contain more different records for the same time unit, these values will be aggregated according selected member aggregation method. These options are same as in case of time aggregation methods.

	2010		
	I		
	January	February	March
	Total [USD]	Total [USD]	Total [USD]
[-] Income	316,497.93	0.69	45,
+ DROB	9,250.00		
+ EB	150,000.00		
+ EURO	1,375.00		
+ HP	36,140.00		
[-] KB	119,732.93	0.69	45,
000/0001	56,310.18		
001/0003	12,154.00		
001/0004	11,492.00		
001/0005	39,758.00		
001/0007	18.75		
002/0005		0.69	
003/0002			45,
003/0006			
004/0001			
005/0003			
006/0002			
006/0004			
007/0002			
+ Outcome	20,593.00	31,471.00	87,

## Appearance and Conditional Formatting

Except detailed indicators settings in tables, it is also possible to modify its depiction using "Appearance settings" link in the bottom right part of indicator settings popup. This function allow the user to:

- edit basic color and font of every indicator
- enable predefined conditions (red and black numbers, increase and decrease, etc.)
- create own condition to activate specific indicator appearance

Basic indicator appearance options include particular color choice, which will be used to depict the indicator values in all table cells in the selected view (to the contrary of the default black-white combination). In addition to color selection, bold display of indicator's values can be activated here.

### Basic appearance

Format for no condition: AB  Bold

Red and black numbers is a function of the standard conditional formatting which depicts every pivot-table cell values greater than threshold in black and values lower than threshold in red. The threshold mentioned usually equals to zero, however it can be modified in the appropriate popup. Thus this function helps the user to simply identify important or critical values in more records such as lossy items, injuries etc.

	Value
Saint Kitts and Nevis	-1 189 873 500
Saint Lucia	-1 381 873 600
Saint Vincent and the Grenadines	-1 058 824 390
Samoa	-1 320 902 900
Sao Tome and Principe	-255 738 560
Saudi Arabia	443 272 170 000
Senegal	-11 675 438 600
Seychelles	-1 233 158 660
Sierra Leone	-2 936 385 020
Singapore	225 134 902 500
Slovakia	-9 821 713 000

Another option of conditional formatting usage is to monitor increases and decreases of indicator values. This allows the user to clearly differentiate indicator values in comparison to previous time periods using colors and arrows. This setting is accessible via the same popup as red and black numbers function.

	2010							
	I			II			III	
	January	February	March	April	May	June	July	August
Invoice					16,412		67,334	
Issued Goods		95	3,900					
Production		47,675	10,800		10,000			
Receipt Card		16,572			17,160		32,063	
Requisition		59,494	7,573		573,548			
Transmission	42,682							

"Create condition..." link serves as possibility to define custom formatting conditions meeting user's specific needs how to display indicators. There can be selected arbitrary color out of displayed palette and eventually a symbol for a lot of custom conditions. These conditions can be made up of plenty of options offered - it is possible to compare absolute and relative growth or cut-down by the arbitrary value. In addition indicator's value can be compared to its previous value or constant entered.

"Add" button allows the user to insert a new condition to the list of already existing conditional formattings. Any condition can be removed from this list by clicking on cross icon on the right side of the page. A lump of custom conditions combinations can be created during this process. The conditional formatting may coexist with basic indicator depiction settings at the same time. Using particular conditions serve to compare value of selected indicator with:

- specified constant
- previous value of the same indicator
- value of another indicator in the same time unit

The screenshot shows a configuration window for 'Total - Appearance' with the following settings:

- Basic appearance:** Format for no condition:  (selected),  Bold.
- Conditional formatting:** Conditional formatting presets: [dropdown menu] **Apply**
- Condition 1:** Apply format **AB** when indicator's value is lower than previous value. A red arrow points to this condition.
- Create condition...:** Apply format **AB** and symbol **Exclamation mark** when indicator's value is **greater than** value: **10000**. **Add** button.
- Save** button.

Except indicators it's possible to change appearance of particular time intervals in the table. You can find this adjustment directly in the time intervals's settings. There is also possible to forbid application of conditional formatting in this specific interval (if you don't want to combine two different kinds of appearance).

✕

---

Name

---

Appearance  Bold  
 Avoid conditional formatting  
ABCabc  Emphasise only

---

Time Unit

You may find another formatting settings if you want to display sum cells for first drill down level, when you select drill-down attribute for the table.

✕

---

Drill down by

---

Sum  Display sum cells for first drill down level  
ABCabc  Bold

---

Back
Save

### Indicators Duplication

Every indicator used in the view can be duplicated. The duplicated indicator inherits all properties of the original indicator (unit, decimal format, aggregation and formula eventually), however this indicator carries "(Copy)" in its label by default. Duplication makes using indicators more effective even further. Often we like to observe the same indicator in single view however with different settings - eg. calculated by another aggregation method. So it is possible to depict minimum, maximum, average and sum of all the values at the same time. The only thing user needs to is duplicate the original indicator more times and set different aggregation method for every duplicated indicator.



### Indicator Groups

Indicator groups simplify using a lot of indicators with similar characteristics (eg. cost indicators). The groups of indicator creation and assigning indicators to these groups in data sets is described in chapter 2.2.2. On the view level it is possible to insert arbitrary indicators from the group (their name begins "Indicator in group") and/or the whole group of indicators. The whole indicator groups can be added only to pivot-tables. The detailed settings allows the user to define if (and eventually how) to aggregate indicators on the group level (sum, average). The indicator group is then displayed in the pivot-table as individual indicators with an additional row containing the value of the selected aggregation.



Group		Indicators		I			II	
				January	February	March	April	May
				COSTS [USD]	Variable [USD]	356	60,441	11,473
Fix [USD]	42,682	123,836	22,273			617,120		
Σ	43038.00	184277.00	33746.00			917100.00		

It is also possible to create indicator groups ad-hoc in indicator settings by clicking on "create indicator group". Then you will be able to create hierarchy of indicators and indicator groups simply by moving them using drag and drop function.

### Indicators displayed in the view

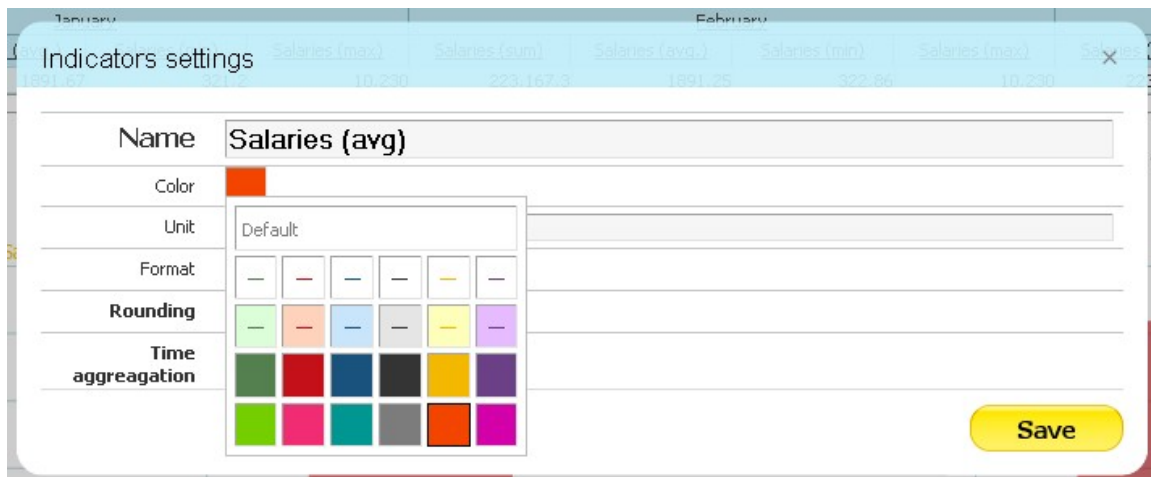


### Indicator's Color Change in Charts

Currently, custom color selection of indicators in charts is accessible only for the following chart types:

- line chart
- bar chart
- radial chart

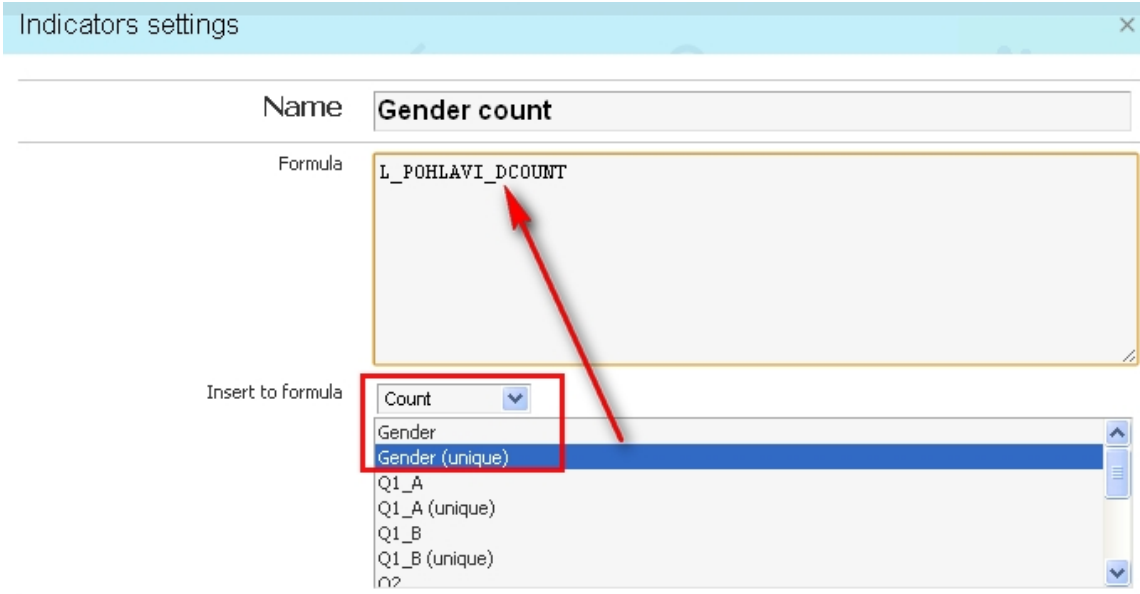
The user should take in account that the indicators' custom color selection has effect only in the case of no drill-down path selected in bar and line charts. On the other side it is possible to show more different indicators with custom colors in the single chart. Indicator's color can be set by using a simple color palette which can be found in the popup of detailed indicators settings within enumerated chart types. The palette offers generic colors and also colors which highlight the area below the curve (line chart) or the area between the points and the middle (radial chart).



### Count and Unique Count

Count indicator is a specific kind of formula indicator. It's a useful tool if you want to know, how often any member is repeated in one time unit. For example: you can analyze a survey data, which contain in each record information about gender of particular respondent. You want to know, how many persons are males and how many persons are females. The solution is to import "gender" as an attribute, M/F will be its members and create new own formula indicator, consisting of count function for this attribute. If you want to use count in formula, you have to choose "count" from formula categories and select related attribute from the list below - gender. This new indicator will show you an absolute number of all gender

records (summarized males and females). But if you select gender as the drill-down path attribute, you will see this count indicator differentiated for males and females separately.



An alternative to count function is "Unique count" function. Unique count indicator doesn't display the total number of members of selected attribute. It shows the number of different members in data. Repeating members are not counted. It means that standard count indicator may show you for example "20" as a count of gender members (10 males, 10 females), but unique count indicator shows you only "2", because there are only two possible gender members, which are repeated.

## Cumulative Indicators

We usually want to know not only absolute values of some indicator but also the cumulative value of that indicator. Cumulative value means gradual summarization of value since specified moment at time. Cumulative indicators may be created by formula and inserted in the same view next to original regular indicator (so we are able to see actual and cumulated value of some indicator next to each other).

Cumulative indicator can be created as new formula indicator. Just choose "time function" from formula category and select "cumulative value" from the list below. In this formula you have to specify start time of cumulating (in our example "1.1.2010") and the name of regular indicator, which will be cumulated ("Value"):

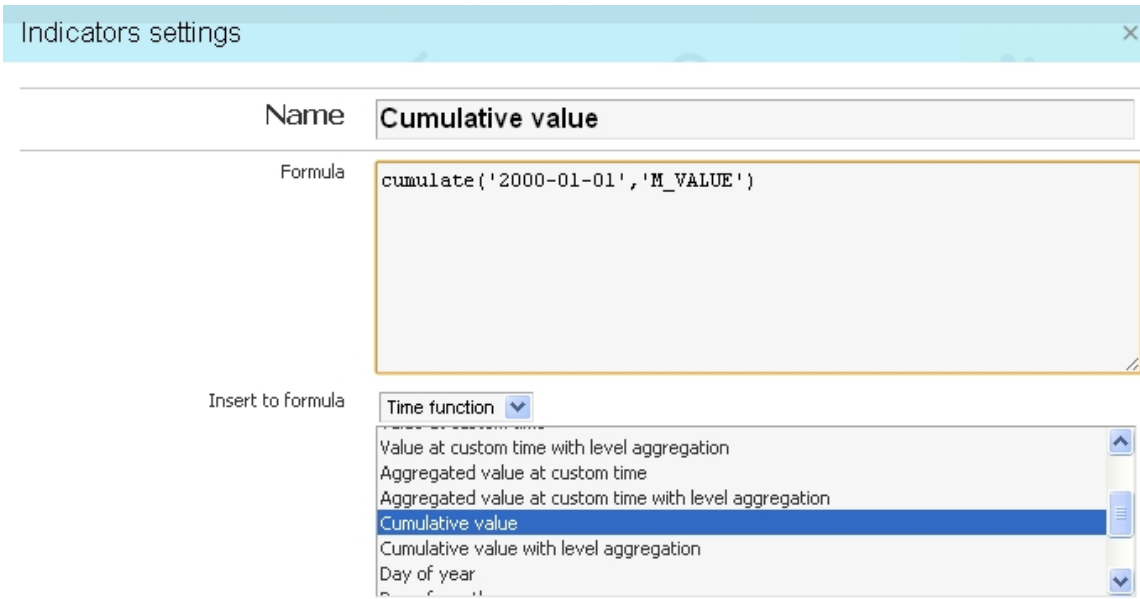


Table below shows the regular indicator (value) and formula indicator (cumulative value), created on the picture above.

Swap		2000	2001	2002	2003	2004	2005	2006
Value	Japan	69,090,500,000	26,481,000,000	51,601,100,000	72,488,100,000	94,230,500,000	69,912,000,000	63,045,700,000
Cumulative value	Japan	69,090,500,000	95,571,500,000	147,172,600,000	219,660,700,000	313,891,200,000	383,803,200,000	446,848,900,000

Note, that if you want to display cumulative indicator in week time unit, you have to enter (into formula) start date, which is Monday in calendar. In other cases no values will be displayed.

## Video tutorials

# Video-tutorials

[Conditional formating](#)

---

[Members aggregation](#)

---

[Time aggregation](#)

---

[Count indicators](#)

---

[Cumulative indicators](#)

---

## 2.3.5 Time settings

Within table time settings it is necessary to set time unit (days, weeks, months, quarters, years) and also particular interval to be observed. There exist three different types of intervals:

- absolute - (common setting) displays time from the first to the last day of the selected time unit (granularity)
- relative - indicates the range of the time interval from actual day according to selected time unit (granularity)
- custom - allows user to choose the interval on daily basis and keep arbitrary time granularity

### Relative interval

Relative interval can be defined by both positive and negative numbers. Actual time unit (eg. day or month) is represented by zero. For example relative interval "From -1 To 3" in conjunction with time granularity months means, period starting one month earlier than actual month and ends three months later will be displayed. Imagine it is April 2010. Therefore period from March to July 2010 will be displayed. Similar rules are valid also for another time granularities.

The screenshot shows the 'Interval settings' dialog box. The 'Name' field contains 'Relative'. The 'Time Unit' is set to 'Month'. Under 'Time interval', the 'Relative' radio button is selected. The 'From - To' field is set to '-3 - 4'. A red box highlights the '-3 - 4' text, and a red arrow points from a text box to it. The text box contains the text 'According to current date (month)'. There are 'Back' and 'Save' buttons at the bottom.

### Custom interval

Second specific case is represented by custom interval. This allows user to define accurate beginnings and end of intervals on the lowest time granularity available (days) and simultaneously display arbitrary time granularity. That is the main difference from absolute interval, in which only time units corresponding to time granularity set are allowed (month granularity allows choice between months 1 to 12, week granularity allows choice 1 to 52 etc.). In case of custom interval it is possible to select eg. years as time granularity and simultaneously accurately define, from which part of the year indicators in table will be displayed. If the user choose month time granularity and selected interval from 15th January 2010 to 15. February 2010, he would see two months in the table (January, February 2010), however indicator values will correspond to selected period from half of January to half of February.

The screenshot shows the 'Interval settings' dialog box. The 'Name' field contains 'Custom Interval'. The 'Time Unit' is set to 'Month'. Under 'Time interval', the 'Custom' radio button is selected. The 'From - To' field is set to '15.1.2000 - 15.2.2000'. A red box highlights the '15.1.2000 - 15.2.2000' text, and a red arrow points from the 'Month' dropdown to it. There are 'Back' and 'Save' buttons at the bottom.

It is possible to use universal variables, in custom interval fields. These variables will be still actual according to actual date or actual data

- now - represents actual date (today)
- availableFrom - represents the first day in source data (refreshed after each particular import)
- availableTo - represents the last day in source data (refreshed after each particular import)

### Time interval in one column

The last useful function regarding time in the tables is the option to merge the whole observed interval into the single column, using checkbox in Time settings popup. The interval type selected has no influence here. In all cases the whole period will be aggregated into the single column. Values in this column usually represent the sum of values from the observed period (no matter of time granularity chosen). Time aggregation of

each indicator can be set differently however (eg. minimum, maximum or average - see chapter 2.3.4 for more information).

Normally new imports to the data sets occur. Therefore time interval in the reports is needed to be refreshed during analysis. When the user want to modify the time interval according to the whole range of data imported, it is sufficient to click on "Set according to data availability" link. So actual data available in the data set is not necessary to look for neither change displayed interval From-To in the view.

### Entering time values

- you can enter time (date) absolutely in two different formats: dd.MM.yyyy (e.g. 1.12.2010), or yyyy-MM-dd (e.g. 2010-12-01)
- it's also possible to enter date relatively:
  - now - represents actual date
  - actualyear - represents the first day of actual year (1.1.20XX). For example actualyear selected on 21.9.2010 represents date 1.1.2010
  - actualquarter - represents the first day of actual quarter (1.1.20XX, 1.4.20XX, 1.7.20XX, 1.10.20XX). For example actualquarter selected on 21.9.2010 represents date 1.7.2010
  - actualmonth - represents the first day of actual month (1.1.20XX, 1.2.20XX, ...). For example actualmonth selected in 21.9.2010 represents date 1.9.2010
  - actualweek - represents first day of actual week (Monday). For example actualweek selected on 21.9.2010 represents date 20.9.2010 (Monday of this week in calendar)
  - relative and absolute enterig of date can be adjusted by operators using this syntax: date +/- n[d|BellaDati25en:w|m|q|y], where n is integer, d represents day, w represents week, m represents month q represents quartal and y represents year. We can for example define time in this way: actualyear + 2m -4d. Today is 21.9.2010, so this value represents 1.1.2010 + 2 months - 4

days, which means date 25.2.2010.

## 2.3.6 Analytical filters

- Purpose of Analytical Filters
- Filter Creation and Editing
- Wildcards
- Filters in Another Parts of BellaDati

### Purpose of Analytical Filters

Analytical filters have the key meaning in all types of view in the report. Besides this usage they server as a worthwhile tool on another locations of BellaDati (see the last section of this chapter). Filters enables creating sharper data slices which allows user to observe values in more detailed and efficient way according user's interest. In the filter particular members can be selected to show for each drill-down attribute in the view. There is a set of conditions in every filter which allow to define which particular members will be or will not be displayed in the view. It is important filters allow to select data also in these drill-down attributes which are not directly displayed in the selected view (eg. they are not covered by used drill-down path by pivot-tables or they do not correspond to selected attribute in chart).

### Filter Creation and Editing

The filters can be defined for every view using a funnel icon in the view's control panel.



This icon becomes highlighted after the filter is activated so the user is visually informed about the view is filtered by some condition.



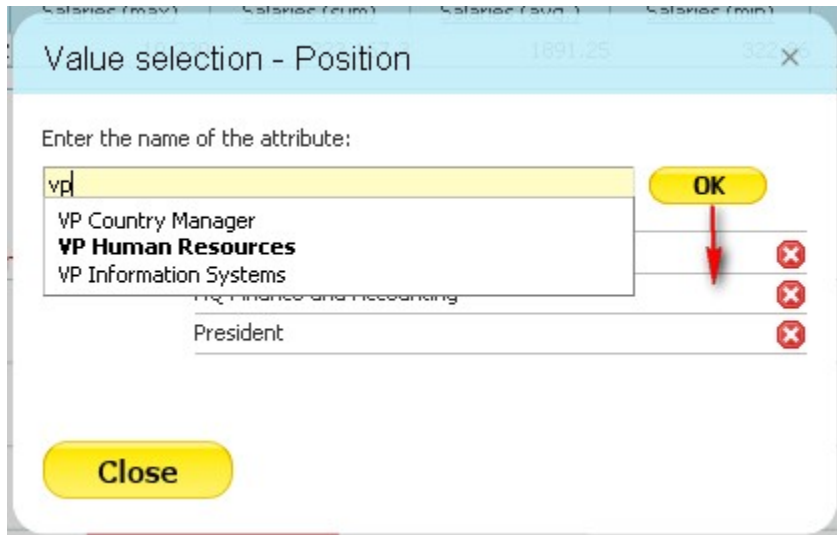
#### Active filter

Every drill-down attribute from source data set is displayed on each row in the filter settings. For each of these drill-down attributes it is possible to choose one of the five conditions determining which members will be displayed in the selected view. These conditions are following:

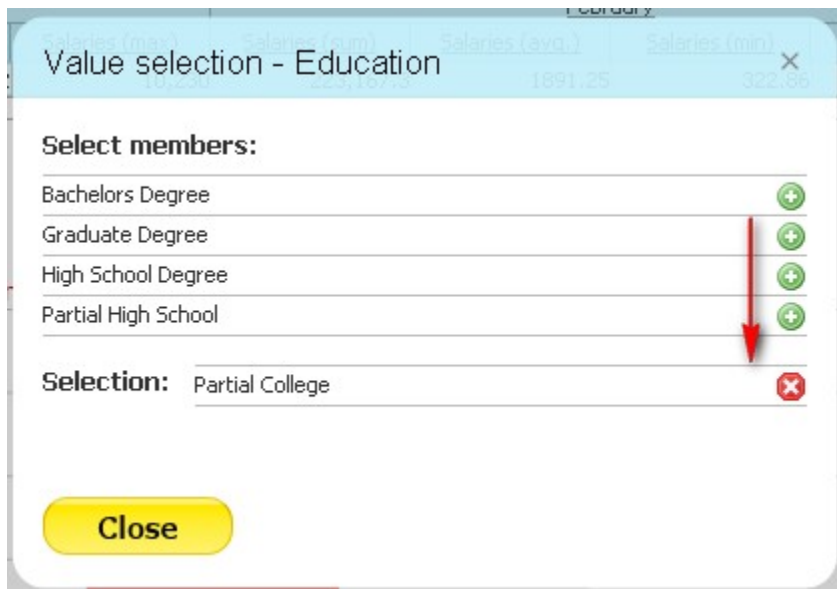
- equals to - selects just one member to display
- not equal to - selects just one member not to be displayed (selected member will be ommited)
- contains - allows to select more members, which will be displayed in according attribute
- don't contain - allows to select more members, which will not be displayed in according attribute
- has value - displays only members with any values
- empty - displays only members with empty values

Filter settings	
County	
Lastname	
Department	has value
Position	contains
Education	equals
State	empty
Firstname	contains
	not equal to
	has value

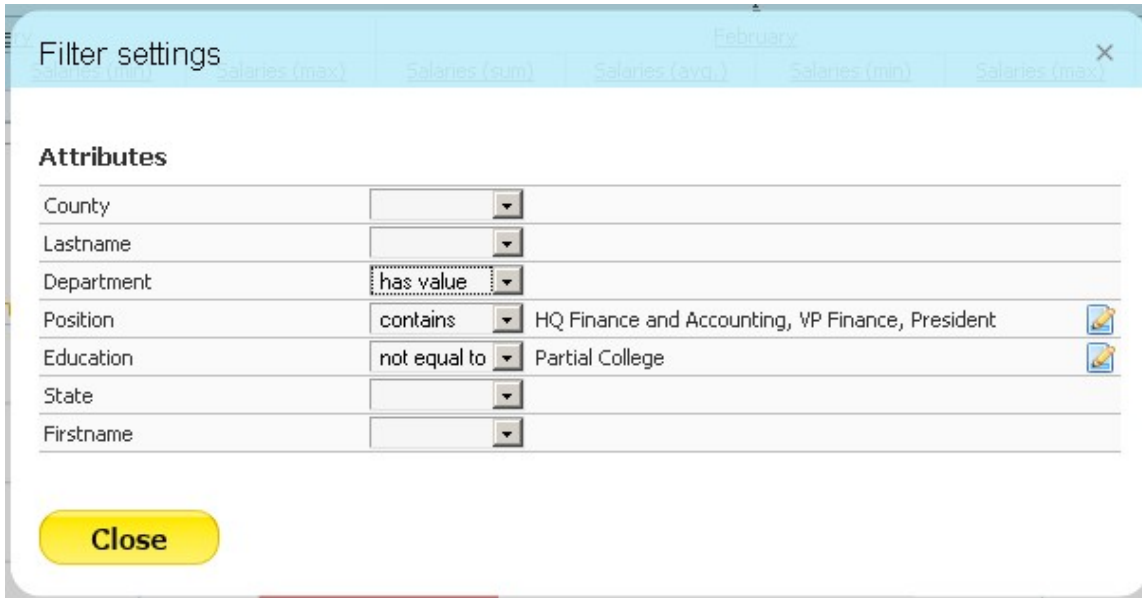
For the first three conditions it is necessary to choose particular members that will apply for selected condition by clicking on icon symbolizing a pencil on a paper. These members can be chosen by clicking on "+" symbol nearby offered members. After this action is completed selected members will move down to the bottom of the popup.



In case there a lot of members in that particular drill-down attribute, an row with full-text search function among all members will automatically appear instead their list. The user is then able to add desired members to the list by clicking "Add" button.



After all required members have been selected the choice must be confirmed and we recommend to verify filter settings. If every condition is configured properly, the user is able to close filter settings popup and right after that all conditions become active in the desired view.



## Wildcards

It is possible to use symbol "%" as a wildcard. It stand for any other combination of characters. It can be used in following conditions:

- equals to
- not equal to
- contains
- don't contain

Example: we have a table with a "product" attribute. Its members are iPod, iPhone and Nokia N8. We use filter condition "contains" and type "ip%". After confirming we can see only iPod and iPhone in the table.

## Filters in Another Parts of BellaDati

Filters of drill-down members are well aplicable outside reports too. They are a suitable tool for users with "data manager" role since they allows filtering:

- during browsing records in the data sets (see chapter 2.2.2)
- by creation and editing of alarms (see chapter 2.2.7)

At first the filter makes browsing actual data more well-arranged in selected area of the data warehouse. Secondly filters specify particular area of data for alarms to be triggered.

## 2.3.7 Drill-down path

- [Predefined vs. ad-hoc drill-down path](#)
- [How to create ad-hoc drill down path](#)
- [Combination of more drill-down path in one pivot table](#)

### Predefined vs. ad-hoc drill-down path

It was mentioned in the Quickstar guide, that predefined drill/down path in the data set is not the only way, how to select the the drill-path. It's because you can choose your drill/down path in a particular pivot table directly ad-hoc (without preparation in data set) in the report. This section is focused on this ad-hoc creation of drill down path in reports and the ways how to combine both possibilities. Suppose that you have got a report with a prepared drill-down path which looks like this one:



## Add drill down path

03/2010



Drill down by **Affiliate**

Calculate sum for each level of drill down path

### Predefined drill down path:



Back

Save

## How to create ad-hoc drill down path

Even if you don't have a predefined drill-down path, you can still choose any available attribute (in the table settings), which could be the first level of the drill-down path. In this case won't be displayed any available tree of drill-down path .

## Add drill down path

03/2010



Drill down by **Product set**

Calculate sum for each level of drill down path

Back

Save

If you confirm this choice, you will see a pivot table with only one-level drill-down path ("product set" in our example). If you click on the "+" drill-down symbol, you have to choose how should the drill-down path continue. Therefore choose the next attribute of your new drill-down-path (it could be part of any predefined drill-down path as well).

	+ 101100002	
	+ 101100006	

Select the next attribute in drill down path

Select the following attribute in drill down path:

- Affiliate
- Customer
- Employee
- Product Code

If you really choose an attribute, which is a part of predefined drill-down path and click on the drill-down symbol again, you will be able to choose between rest of the predefined drill-down path OR any other available attribute. If you want to continue the path, click on the according part of the displayed tree.

[-]		
Income [EUR]	+ Berlin	
	+ Paris	
	+ Pilsen	
	[-] Prague	
	+ ABX Transport	
	+ Data Sins	
	+ Gant Petrol	
	+ GD Electronics	
	+ Genuie SW	
	+ Synrgia COMM	

Select the next attribute in drill down path ✕

Customer — Product Code

▼ Advanced settings...

If you want to continue with any other attribute, click on the "advanced settings" and then choose, which attribute you wish to use. There is no limitation in your choice.

[-]		
Income [EUR]	+ Berlin	
	+ Paris	
	+ Pilsen	
	[-] Prague	
	+ ABX Transport	
	+ Data Sins	
	+ Gant Petrol	
	+ GD Electronics	
	+ Genuie SW	
	+ Synrgia COMM	

Select the next attribute in drill down path ✕

▲ Simple setup...

**Select the following attribute in drill down path:**

📁 Employee

📁 Product set

📁 Product Code

Important thing is, that you can save your chosen drill-down path or cancel it. If you want to save your selected path, click on the floppy-disc symbol. If you want to undo it, click on the symbol of floppy-disc with a back arrow. After this, the same (saved) drill-down path will be displayed after repeating drill-down/roll-up operations. If you don't save your drill-down path, you will have to choose it again, when you next time open this report.

Overview

		Jan 1, 2010 - Aug 18, 2010	
Income [EUR]	Berlin		89,429.07
	Paris		64,442.34
	Pilsen		34,757.58
	Prague		96,240.61
	ABX Transport		3,400
	Data Sins		9,330.46
	Gant Petrol		35,975.01
	GD Electronics		25,183.6
	Genuie SW		20,173.54
	Syngia COMM		2,178
Z100		2,178	

## Combination of more drill-down path in one pivot table

It's possible to insert more than one drill-down path in only one table. It's sometimes useful to display different attributes or even whole different drill-down paths on different axis of the table. If you click again on "add drill-down path" (table settings) in case that you have chosen one drill down path, you will be able to add another drill-down path and place it on the second axis of the table.

Table settings

**X axis**

- Time
- Customer

**Y axis**

- Indicators
- Affiliate

[Add drill down path](#)
[Table appearance](#)
[Custom members](#)

Close

This will cause a more detailed differentiation of the table. You can freely examine relationships between individual members.

		Jan 1, 2010 - Aug 18, 2010						
		ABX Transport	Data Sins	Gant Petrol	GD Electronics	Genuie SW	Syngia COMM	Z100
Income [EUR]	Berlin	21,558	22,280	3,508	8,979.58	10,493.49	95	22,610
	Paris	4,972.22	19,990	20,621	17,008	85	1,751.12	
	Pilsen	9,251.44	183.3	2,660	1,353.6	8,526.4	11,782.84	
	Prague	3,400	9,330.46	35,975.01	25,183.6	20,173.54	2,178	

## 2.3.8 Cooperation and sharing

BellaDati is a service focused on adopted for cooperation of more users in one single report. There is a set of features, which allow users to fluently share actual analytical knowledge from data.

- Description and comments
- Table comments
- iFrame
- iGoogle
- Report copy
- Sharing reports with other users in current domain

### Description and comments

The first and the most easy way, how to share basic information about the report, is through the report description. You can change the name and description of the report by clicking directly into its text field. Then just edit or add new text and confirm your changes by the "save" button.

The second possibility is to share information about report via comments. Comments are short blocks of text, which can be optionally displayed on

the right part of the report screen. If you want to display all existing comments, click on the "Comments" button on the right vertical side of the report screen.

Comments



You can enter your text into the text field below in this window. Maximal length of one comment is 800 characters. There is also displayed name and picture of the author of the particular report. You can also delete your own comments by the red symbol next to each of your comments. If there is a large number of comments, it's possible to switch pages of comments by "next" and "previous" buttons. It's possible to hide or redisplay comments by the "hide" link in the upper right corner of the comment window or by the comments button.

**Add comment** [» Hide](#)

I want watch this conversation


**Submit**

**Comments**

 [George Analyst](#), Sep 13, 2010 12:37 PM 


OK, it will be ready during this week. Afterwards you will be ready to present analytical outputs to CEO.

---

 [Marty Manager](#), Sep 13, 2010 12:34 PM

George, please prepare analysis and forecastiing fo upcoming business period.

---

 [Marty Manager](#), Sep 13, 2010 12:34 PM

There were no transactions during 6/2010 - it's because of BPE processing.

If you want to be informed about new comments of other users by email notification, click on the "watch this conversation" checkbox of the comments window. Received email contain text of new comments, its author and hypertext link to the source report.

## Table comments

Another feature, which is focused on cooperation and sharing of reports are table comments. These comments may be inserted into any cell of the pivot table. This cell is marked by the red corner. All the readers may display content of the comment by moving of mouse cursor over a particular cell. It always shows author and content of the comment. Table comments are (unlike report comments) displayed also in the workspace, on public dashboards, in iFrames and iGoogle dashlets. Table cells may be inserted by the owner of the report and by all the sharing users (with right to edit report). Owner may delete any comments, sharing users may delete only their own table comments.

Income	Planned values
01/2010 - 08/2010	01/2010 - 08/2010
78,361.32	
103,567.52	
125,528.02	
105,109.56	

Denis Neumann: This low value is caused by the decrease in production

## iFrame

Another way, how to share your analytical outputs, is through the iFrame. This feature allows users to share their particular views from reports on public web sites. The view on the external website is displayed with a link connected to the source report. iFrame view remains on web site even after you log out from BellaDati. If anybody else click on the link to the source report, he will be moved to the login screen. If you lose the permission for this view (if anybody cancels sharing of this report with you, or in the case when your user account will be deleted from the domain), the iFrame won't be displayed on external website anymore. Domain administrator is also able to forbid all the iFrame/iGoogle/public dashboard publishing for all the users of his domain.

If you want to create iFrame, click on the "Sharing - Web" button at the report toolbar. Then choose a particular view, which you want to publish on the web, from the list. After this you should choose size of your iFrame - it's recommended to use maximal size of the iFrame, because this causes flexible and automatic adjusting of the size to the target website. When you select the size of the iFrame view, you can copy CTRL+C the code below and paste it CTRL+W directly into target HTML code.

**Place on the web** ✕


---

**View** Business overview 2010


---

**Publish on** Web


---



Small (250x180)



Medium (500x300)



Wide (fits window width)

You can add the selected item to any external website. Please copy the HTML code from the bottom field of this dialog box.

```
<iframe style="border: 1px solid silver;width:100%;height:600px;"
src="https://service.belladati.com:443/en/bi/igadget
/view/dashletgadget?dashletType=VIEW_REPORT&h=600&
&uid=t9cPyQsyN69iWQ%2Fle26ZLw%3D%3D&
&uidc=m32mOz9%2BM3lg8bVG%2BR7fpw%3D%3D&viewReportId=931">
</iframe>
```

Close

 The important thing is, that it's not allowed to use drill down operation in iFrame/iGoogle views on external web sites.

## iGoogle

iGoogle is very similar to iFrame feature. It allows users to place their view directly on their iGoogle dashboard. It's not necessary to copy any HTML code in this case. You just need to choose "iGoogle" in the web sharing window instead of "web".

Place on the web ×

---


**View** Incomes vs. Costs

---

**Publish on** iGoogle

---

Click on the following link to add the selected view onto your iGoogle desktop.

 [Google](#)

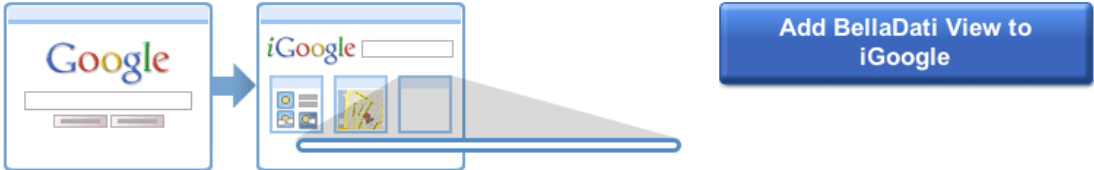
---

**Close**

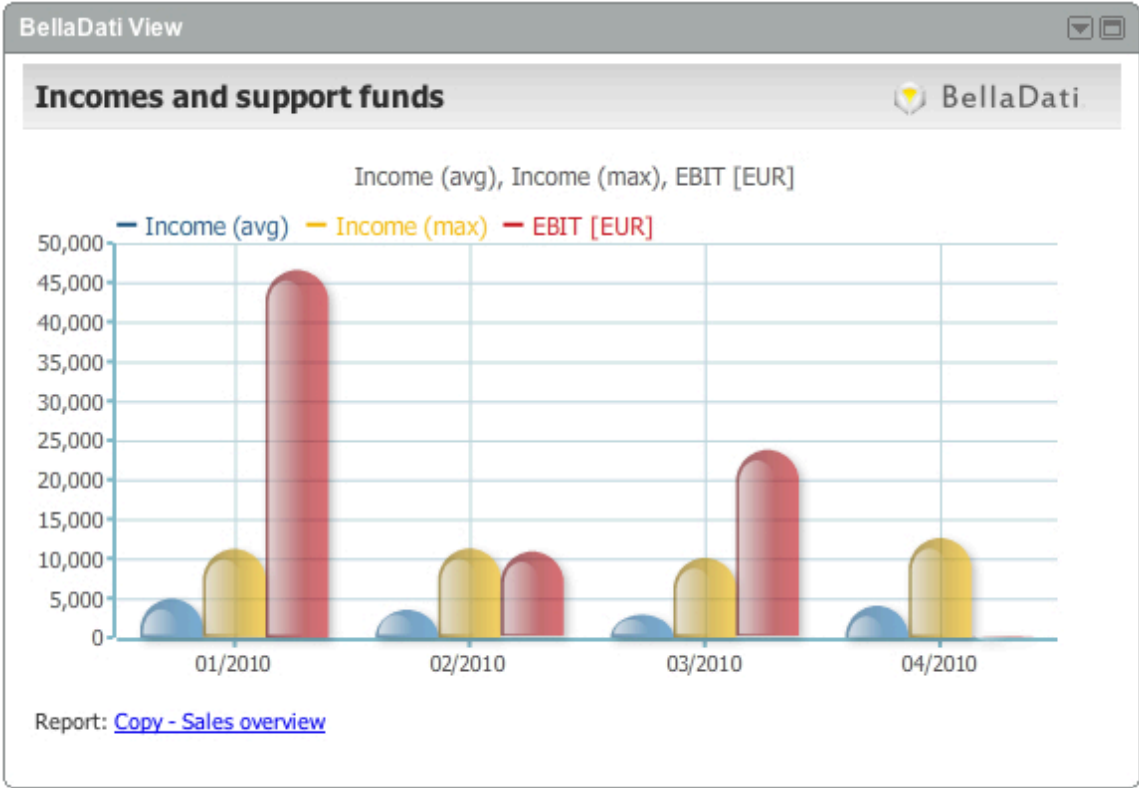
Then just click on the iGoogle button and confirm your choice on the next screen.

**iGoogle**

Get BellaDati View on your Google homepage

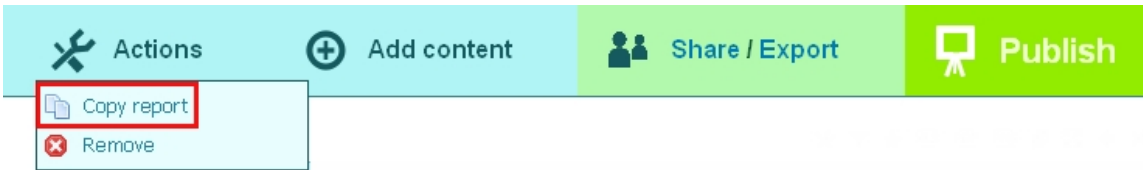


You will see a brand new dashlet containing your selected view on your iGoogle dashboard. You can easily change your iGoogle dashboard to the full analytical overview with your favorite views from different reports. The security rules are the same as in the case of iFrame.



## Report copy


Report copy is very useful feature when you want to share your whole report with other users. If you create a report, you can make unlimited number of its copies. Each clone is labeled with additional text "Copy" in its name. This copy contains all the views, active filters and other settings of the original report. Only exceptions are comments, which are not copied.

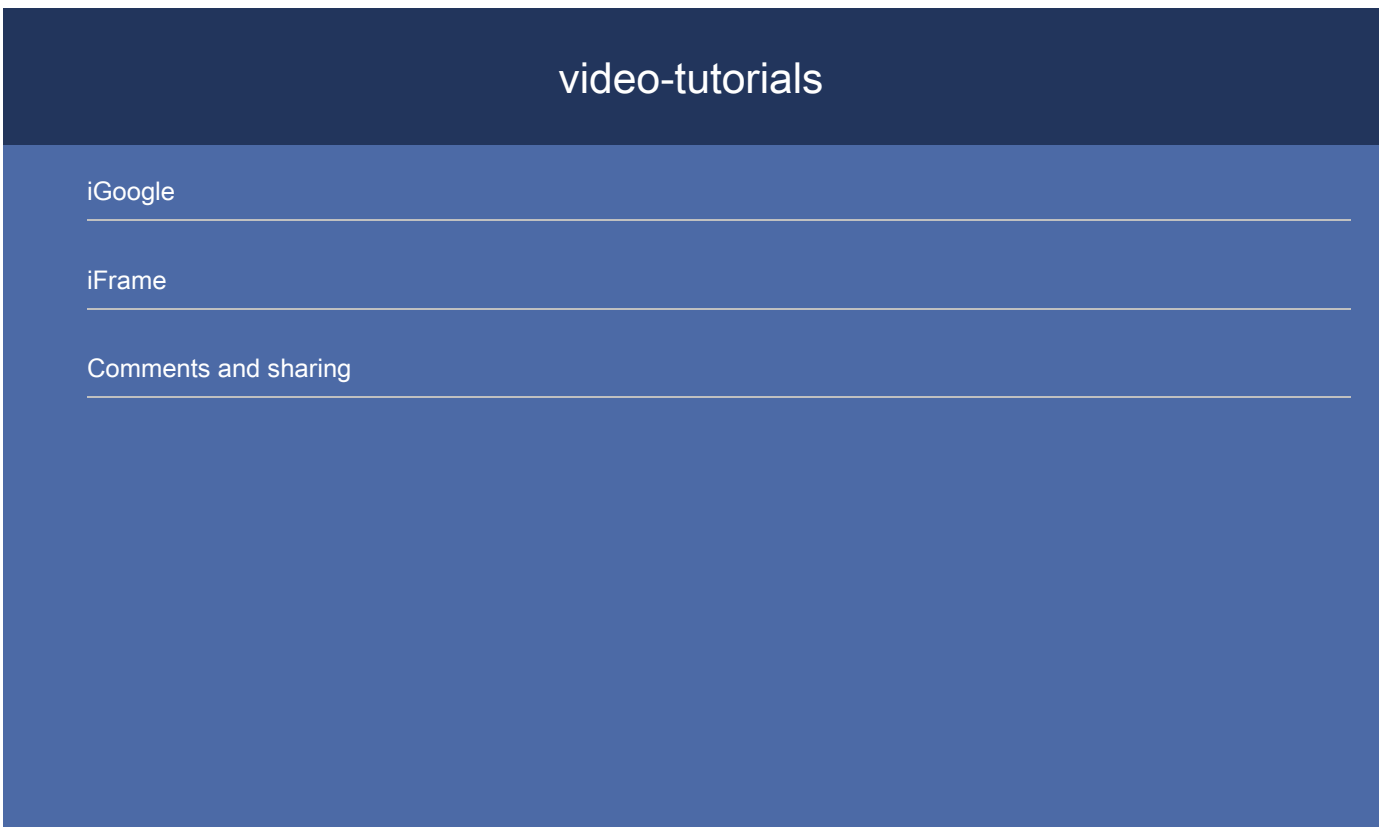


If you want to create a copy of your opened report, click on the "Actions - Copy report" button at the report toolbar. You will be automatically moved to the new copy.

## Sharing reports with other users in current domain

You can of course share your whole reports with other users in your domain. Purpose and way how to share your report is described in chapter 2.5.3.

 TIP It's useful to combine report sharing with a report copying. You can create a copy of the report first and share this copy with other users or whole user groups. In this case you are protected from damaging of your original report.



## 2.3.9 Custom content and attachments

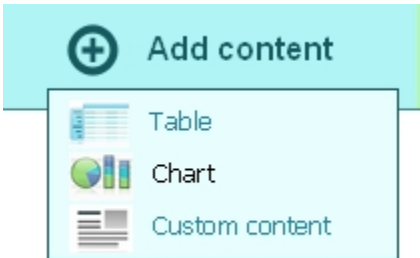
- Purpose of custom content
- How to create custom content
- HTML in custom content
- Report attachments
- Video-tutorial for this chapter

## Purpose of custom content

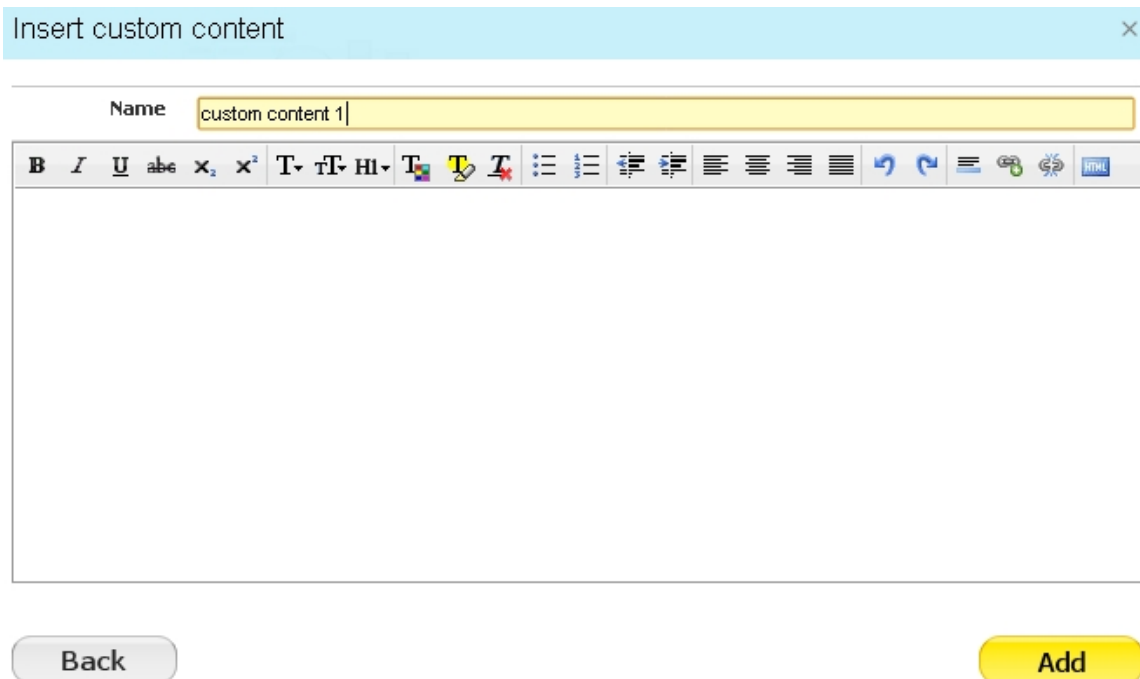
Charts and tables is not the only entity, which can be insert into report. It's possible to insert one or more custom contents directly into your report. This content can be used as a complement to existing views. It can contain any formatted text, pictures, hypertext links or even embed videos. Therefore you can use your own custom content to enrich your report by medial information.


## How to create custom content

Custom content can be inserted into report in the same way like views - click on the "add view" button and select "custom content" from the list.



You will open the window, where you will be able to insert your own text and adjust it with a standard WYSIWYG editor in the line above. You can change font, alignment, color of your text, insert hypertext links or edit HTML code. You can also choose the appearance of the whole content and its title.



 TIP: It's possible to create custom content dashlet directly in the the workspace in the same way. It is located in the list of "other" dashlets.

## HTML in custom content

It's possible to insert a lot of different objects via the HTML editor in custom content - pictures, iframes, embed videos etc. If you don't want to edit the code, it's enough to just choose the selected area on external web page by mouse, press CTRL+C, click into your custom content and press CTRL+V. All the selected content will be copied into your report. It's possible to easily copy pictures, formatted texts, videos or hypertext links in this way as well.



### History of home values 2010

The video player displays a news segment titled "Home Prices, Historical Data Chart - Glenn Beck explains". The chart, titled "HISTORY OF HOME VALUES", plots "THOUSANDS OF DOLLARS" on the y-axis (ranging from 60 to 200) against years on the x-axis (1890, 1910, 1930, 1950, 1960, 1970, 2000, 2010). A red line shows fluctuations, with a sharp peak around 2000 and a subsequent decline. A red circle highlights the year 2010, labeled "NOW". The video includes a "FOX NEWS" logo, a "4:05 CT" timestamp, and a "G3" logo. A "YouTube" watermark is visible in the bottom right corner of the video frame. The video player interface shows a progress bar at 3:33 / 5:51.

### Report attachments

You can attach your own files, containing additional information connected to your report. If you want to insert new attachments, just click on the "Attachments" tab on the right side of the report screen.

Comments

Attachments

It's allowed to attach one or more files from your hard drive. Owner of this report and creators of the particular attachment is also able to delete this attachment. Those attachments can be downloaded by all users, who share this report. Note, that space used for your attachments is calculated into the data limit for your domain.

### Video-tutorial for this chapter

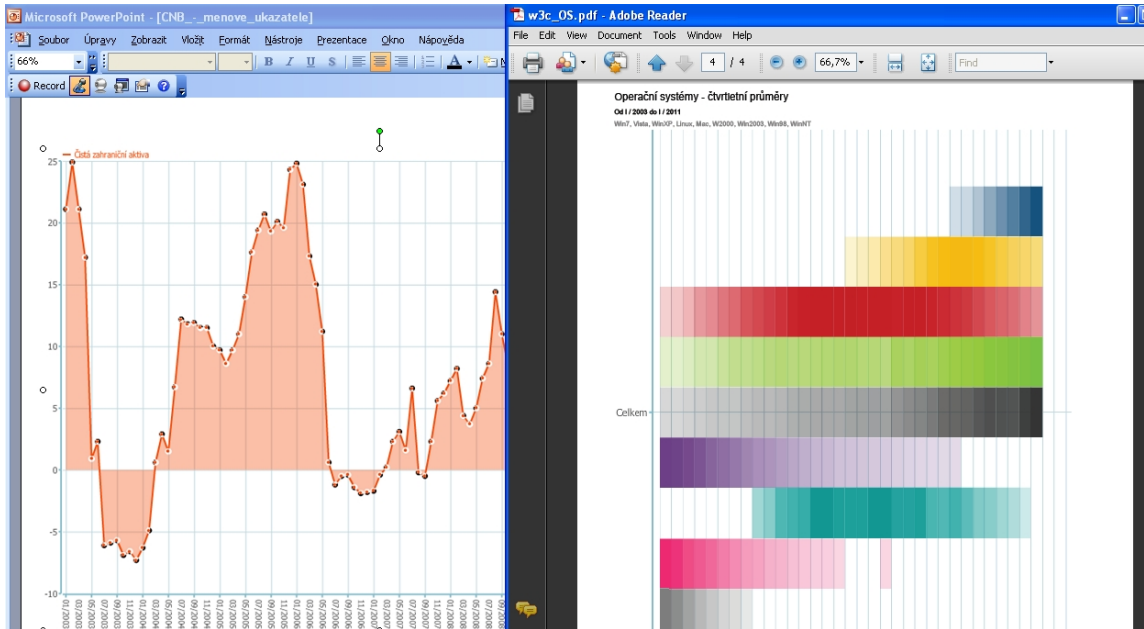
## 2.3.10 Exports

- Whole Report Export
- Individual View Export

### Whole Report Export



Whole reports can be exported into the external formats PDF (Adobe Portable Document Format) and PPT (Microsoft PowerPoint) files for presentation purposes. All you need is to click on the export button and choose the file type and export format in the window. In case of PDF you may choose size (A1..A4) and orientation of the output document. Then just click on the download button to get your output.



In case of PPT export every chart and table is exported to a separate page. Views containing more indicators in more charts are splitted to the individual pages (slides) of the presentation too. Too large tables are splitted into more small tables.

### Individual View Export

Every individual view can be exported separately too. In addition to PDF document export it is possible to export charts into PNG graphic format. Moreover it is possible to export pivot-tables directly to XLS (Microsoft Excel Spreadsheet). If you want to export a standalone chart or table just click on the blank list symbol above to open the export wizard window.



## 2.3.11 Formulas

- General formula rules
- Indicator defined by formula
  - Indicators and codes
  - Mathematical functions
  - Time functions
  - Time variables (and operators)
  - Count
  - Regression

- Context change formulas
- members()
- membersSum()
- Time defined by formula
  - Unsupported functions
  - Function indicator()
- Other formula features
  - Connection of strings by formula
  - Getting of member values
  - Setting and getting of parameters in formula defined indicator

In this chapter you can find different ways where and how to use formulas in BellaDati. You can use formulas:

- as a data manager during preparing of new indicators in the data set
- as a report editor during creating of new indicator for particular chart or table
- as a report editor during creating of new time interval of the pivot table

First two cases differ only in the user roles. If data manager prepare new indicator by formula, this indicator will be available for all the report editors, who will be able to use this data set to create reports. It will be accessible in all the charts and tables and marked by a calculator machine icon. It means that its main advantage is reusability.

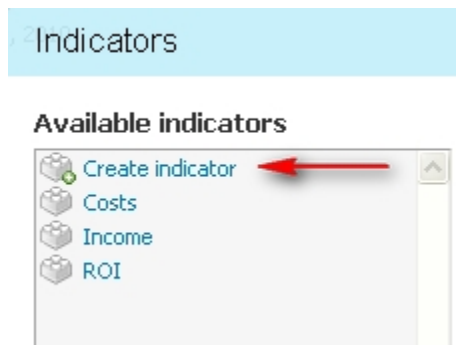
### Indicators

	Action	Code / Formula	
Costs		M_COSTS	<a href="#">Remove</a>
Income		M_INCOME	<a href="#">Remove</a>
ROI		M_ROI	<a href="#">Remove</a>

[Create indicator](#)

[Create a group of indicators](#)

On the other hand indicator created during report analysis by report editor in one chart or table won't be accessible in other views. You can change the formula of this indicator anytime you want without a necessity to change the source data set.



Formulas may be also used to define time intervals in table. In the time settings is possible to click on the link "Insert formula". You can insert more time intervals in one table. If you define some of them by formulas, you can use some functions and variables like in the indicator formulas. Another functions and variables are specific only for time formulas. You can find the detailed description of the particular functions and variables in the text below.



### General formula rules

You don't have to enter formulas manually during their creating. There is a field, where you can directly type the formula, but the easiest way is to use predefined functions and variables in the second part of the formula window. There you just choose appropriate category and then directly select necessary function or variable from the list below. It is immediately displayed in the formula field at the actual position of a mouse cursor. You can freely combine different functions, variables and their categories.

There are some basic rules which are in common for all the formulas:

- indicator codes, time and variables must be inserted between apostrophes, for example: 'M\_INCOME'
- it's possible to enter time absolutely in two different formats: dd.MM.yyyy OR yyyy-MM-dd or relatively by time variables and operators
- it is necessary to enter constants (and eventually apostrophes, commas and brackets in case that they are not included in the function template) manually
- if you want to get value of indicator in the actual time column, it's enough to simply enter its name without apostrophes

Indicators settings
✕

---

**Name**

Formula

Formula

`cumulate('actualYear - 1m', 'M_INCOME')`

formula field

**category**

Time function ▾

Previous value

Previous value with level aggregation

Next value

Next value with level aggregation

Value at custom time

Value at custom time with level aggregation

Aggregated value at custom time

functions or variables in current category

---

Unit

---

Format

[Show help](#)

---

**Rounding**

Classic (half up) ▾

---

Save

## Indicator defined by formula

### Indicators and codes

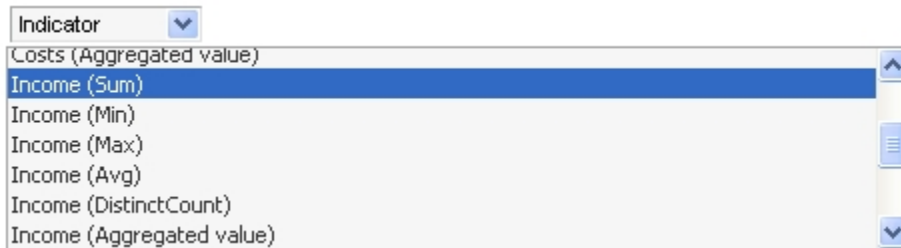
In formulas it's possible to refer to another (regular) indicator by its code. This code consists of "M\_" prefix and the name of the regular indicator without diacritical marks. You can't refer in formula to another indicator defined by formula. You can enter the code using category "Indicators". You just click on the selected indicator in the list below. The shortcut in the brackets appended at the end of the indicator determines its member aggregation (usually "sum" - for more information about aggregations please visit chapter 3.3.4). Another shortcuts means:

- (min) - minimal value
- (max) - maximal value
- (avg) - average value
- (distinctCount) - distinct count of values
- (aggregated value) - value of indicator in the higher level of drill-down path. The number of parameters determines number of higher levels. For example - in the table below you can see three indicators created by formulas: `value('2010-01-01', 'M_INCOME', 1)`, `value('2010-01-01', 'M_INCOME', 2)`, `value('2010-01-01', 'M_INCOME', 3)`:

This function is useful for comparing of the partial value with the total value - e.g. `M_INCOME / value('2010-01-01', 'M_INCOME', 1)`

2010				
I				
January				
Q1				
	Income	value(2010-01-01,'M_INCOME,I)	M_INCOME / value(2010-01-01,'M_INCOME,I)	
Berlin	18,760	61,730	0.3039	
Paris	19,410	61,730	0.31443	
Pilsen	3,750	61,730	0.06075	
Prague	19,810	61,730	0.32091	

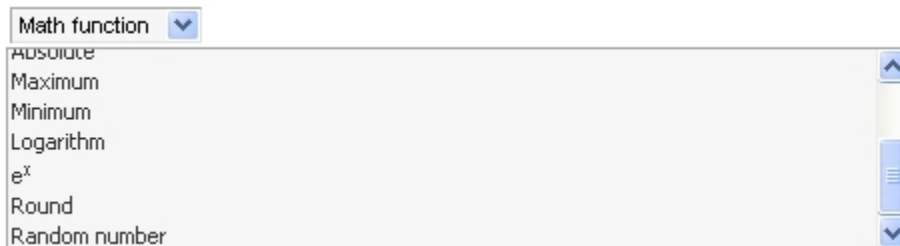
If you want to specify aggregation of the source indicator manually, you can just type the shortcut of aggregation after "at" symbol, e.g. M\_INCOME@AVG or M\_INCOME@DC. If you need to modify time aggregation of the selected indicator, you can insert it in the same way - you just need to append "T" at the end of the shortcut. (e.g. M\_INCOME@AVGT). Both aggregations may be combined together - for example M\_INCOME(at)MIN(at)MINT refers to indicator Income, with members aggregation equal to minimum and also time aggregation equal to minimum.



### Mathematical functions

Naturally it's possible to use a whole variety of mathematical operations in the formulas. Besides basic operations (summation, subtraction, dividing, multiplication etc.) are available some more complex functions:

- absolute value - returns positive equivalent of the number
- maximum - returns bigger of two values
- minimum - return smaller of two values
- logarithm - natural logarithm
- e(x) - exponential function
- round - rounds value to integral number
- random number - returns random number between 0 and 1



### Time functions

Time functions offer possibility to refer to specific time points or intervals. Most of them are associated with the time unit, in which is the formula indicator displayed.

Meanings of particular time functions:.

- Previous value - returns previous value of selected indicator (you need to enter the indicator's code as a parameter of the function). Previous means previous value in the displayed time unit of the chart or table. For example: we have a table, that shows values of income in three months (January, February, March). Indicator created by this function returns January income in February column and February income in the March column.
- Previous value with level aggregation - works in the same way as "previous value". Moreover it contains the second parameter (integral number, default = 1), which allows to display aggregated value of the indicator in the higher level of drill-down path (for more information see chapter "Indicators and codes - (aggregated value)" above).
- Next value - returns next value of selected indicator. Next means next value in the displayed time unit of the chart or table. For example: we have a table, that shows values of income in three months (January, February, March). Indicator created by this function returns February income in January column and March income in February column.

	2010		
	I		
	January	February	March
prev('M_INCOME')		97,376	39,857.77
Income	97,376	39,857.77	25,338.42
next('M_INCOME')	39,857.77	25,338.42	24,298.64

- Next value with level aggregation - works in the same way as "next value". Moreover it contains the second parameter (integral number, default = 1), which allows to display aggregated value of the indicator in the higher level of drill-down path (for more information see chapter "Indicators and codes - (aggregated value)" above).
- Value at custom time - contains two parameters: time in basic format and indicator code. It returns value of selected indicator in selected time point. The value is aggregated according to time unit of the source chart or table. It means that if we type 2010-01-15 and our table is displayed in months, this function returns value for whole January 2010 (not only value for the selected day).

	2010		
	I		
	January	February	March
Income	97,376	39,857.77	25,338.42
value('2010-01-15','M_INCOME')	97,376	97,376	97,376

- Value at custom time with level aggregation - works in the same way as "value at custom time". Moreover it contains another parameter (integral number, default = 1), which allows to display aggregated value of the indicator in the higher level of drill-down path (for more information see chapter "Indicators and codes - (aggregated value)" above).
- Aggregated value at custom time - contains two time parameters and indicator code. It returns values for whole time interval between entered times. Time unit of the source chart or table is not reflected. For example formula value('2010-01-01','2010-02-28','M\_ZISK') returns in each cell a sum of income values between the start of January and the end of February 2010.

	2010		
	I		
	January	February	March
Income	97,376	39,857.77	25,338.42
value('2010-01-15','2010-02-28','M_INCOME')	137,233.77	137,233.77	137,233.77

- Aggregated value at custom time with level aggregation - works in the same way as "aggregated value at custom time". Moreover it contains another parameter (integral number, default = 1), which allows to display aggregated value of the indicator in the higher level of drill-down path (for more information see chapter "Indicators and codes - (aggregated value)" above).
- Cumulated value - contains two parameters: time and indicator. It returns cumulated (summarized) value since the selected time point to the contextual time in the chart or table. Table below shows indicator Income and also cumulated version of this indicator (cumulated since 2010-01-01).

	2010		
	I		
	January	February	March
Income	97,376	39,857.77	25,338.42
cumulate('2010-01-01','M_INCOME')	97,376	137,233.77	162,572.19

- Cumulated value at custom time with level aggregation - works in the same way as "cumulated value". Moreover it contains another parameter (integral number, default = 1), which allows to display aggregated value of the indicator in the higher level of drill-down path (for more information see chapter "Indicators and codes - (aggregated value)" above).

## Time variables (and operators)

### Ordinal variables (return ordinal numbers of the displayed time unit)

- dateDayOfMonth() - returns ordinal number of the day in month
- dateDayOfWeek() - returns ordinal number of the day in week
- dateDayOfYear() - returns ordinal number of the day in year
- dateMonth() - returns ordinal number of the month in the year
- dateYear() - returns year

	2011			
	I			II
	January	February	March	April
Income	290,169.93	21,723.69	133,137.69	260,000
dateDayOfMonth()	1	1	1	1
dateDayOfWeek()	6	2	2	5
dateDayOfYear()	1	32	60	91
dateMonth()	1	2	3	4
dateYear()	2,011	2,011	2,011	2,011

Contextual variables (return date in yyyy-MM-dd format)

- contextDay() - returns date of the first day in context of displayed time unit
- contextWeek() - returns date of the first day in week (Monday) in context of displayed time unit
- contextMonth() - returns date of the first day in month in context of displayed time unit
- contextQuarter() - returns date of the first day in quarter in context of displayed time unit
- contextYear() - returns date of the first day in year in context of displayed time unit

	2011			
	I			II
	January	February	March	April
Income	290,169.93	21,723.69	133,137.69	260,000
contextDay()	01.01.2011	01.02.2011	01.03.2011	01.04.2011
contextWeek()	27.12.2010	31.01.2011	28.02.2011	28.03.2011
contextMonth()	01.01.2011	01.02.2011	01.03.2011	01.04.2011
contextQuarter()	01.01.2011	01.01.2011	01.01.2011	01.04.2011
contextYear()	01.01.2011	01.01.2011	01.01.2011	01.01.2011

Actual variables (return date in yyyy-MM-dd format)

- actualDay() - returns actual day (today)
- actualWeek() - returns date of the first day of actual week (Monday)
- actualMonth() - returns date of the first day of actual month
- actualQuarter() - returns date of the first day of actual quarter
- actualYear() - returns date of the first date of actual year

	2011			
	I			II
	January	February	March	April
Income	290,169.93	21,723.69	133,137.69	260,000
actualDay()	08.03.2011	08.03.2011	08.03.2011	08.03.2011
actualWeek()	07.03.2011	07.03.2011	07.03.2011	07.03.2011
actualMonth()	01.03.2011	01.03.2011	01.03.2011	01.03.2011
actualQuarter()	01.01.2011	01.01.2011	01.01.2011	01.01.2011
actualYear()	01.01.2011	01.01.2011	01.01.2011	01.01.2011

Formulas were displayed on 8 Mar, 2011

### Operators

It is possible to easily modify dates entered absolutely (dd.MM.yyyy or yyyy-MM-dd) or relatively (time variables) by operators. Operators allow to adjust date according to your needs - it's a useful complement, especially for time variables like contextYear() or actualMonth that usually returns the first date of the time unit. Syntax of operators:

date +/- n[d|w|m|q|y]

where n is an integral number, d is day, w is week, m is month, q is quarter, y year. Date (including operators) must be\* located between apostrophes and time variables must be entered without brackets (e.g. actualYear()).

Operator examples:

- '2011-01-01 + 1q - 2m + 1d' means Jan 1, 2011 + 1 quartal - two months + 1 day, that represents Feb 2, 2011
- 'actualYear + 1m + 1d', (used on 8 Mar, 2011) means 1 Feb, 2011 + 1 month + 1 day, that represents 2 Feb, 2011 as well

You can see an application of both operators in "value at time" function in the table below:

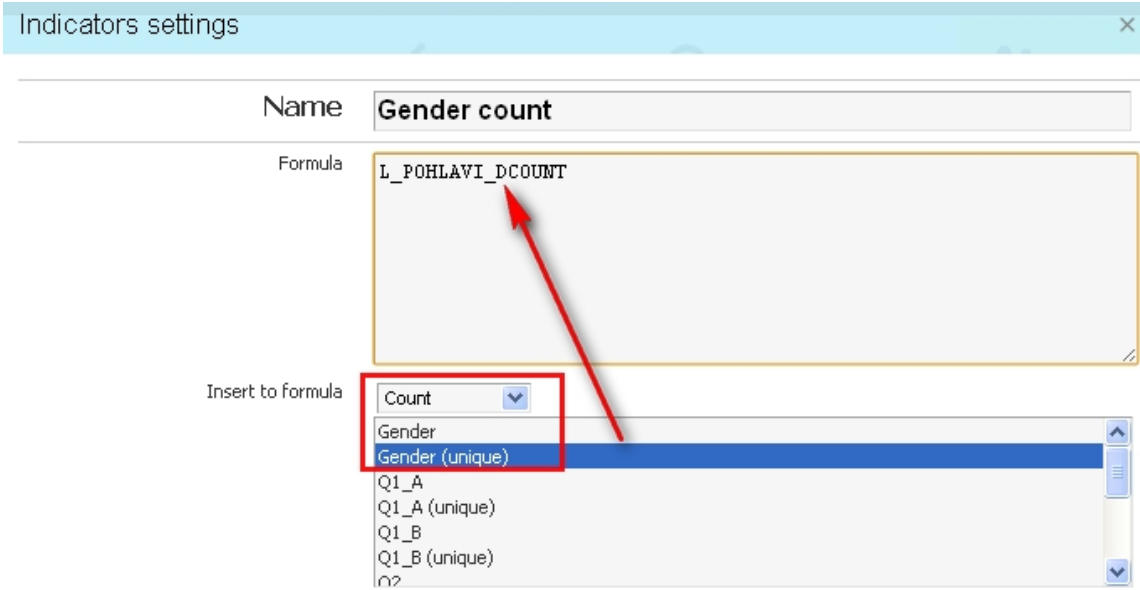
	2011	
	I	II
	February	Q2
Income	150	150

	value('actualYear + 1m + 1d', 'M INCOME')	value('2011-01-01 + 1q - 2m + 1d', 'M INCOME')
Income	150	150

## Count

Count indicator is a specific kind of formula indicator. It's a useful tool if you want to know, how often any member is repeated in one time unit. For example: you can analyze a survey data, which contain in each record information about gender of particular respondent. You want to know, how many persons are males and how many persons are females. The solution is to import "gender" as an attribute, M/F will be its members and create new own formula indicator, consisting of count function for this attribute. If you want to use count in formula, you have to choose "count" from formula categories and select related attribute from the list below - gender. This new indicator will show you an absolute number of all gender records (summarized males and females). But if you select gender as the drill-down path attribute, you will see this count indicator differentiated for males and females separately.



Indicators settings

Name **Gender count**

Formula **L\_POHLAVI\_DCOUNT**

Insert to formula

- Count
- Gender
- Gender (unique)
- Q1\_A
- Q1\_A (unique)
- Q1\_B
- Q1\_B (unique)
- n?

An alternative to count function is "Unique count" function. Unique count indicator doesn't display the total number of members of selected attribute. It shows the number of different members in data. Repeating members are not counted. It means that standard count indicator may show you for example "20" as a count of gender members (10 males, 10 females), but unique count indicator shows you only "2", because there are only two possible gender members, which are repeated.

## Regression

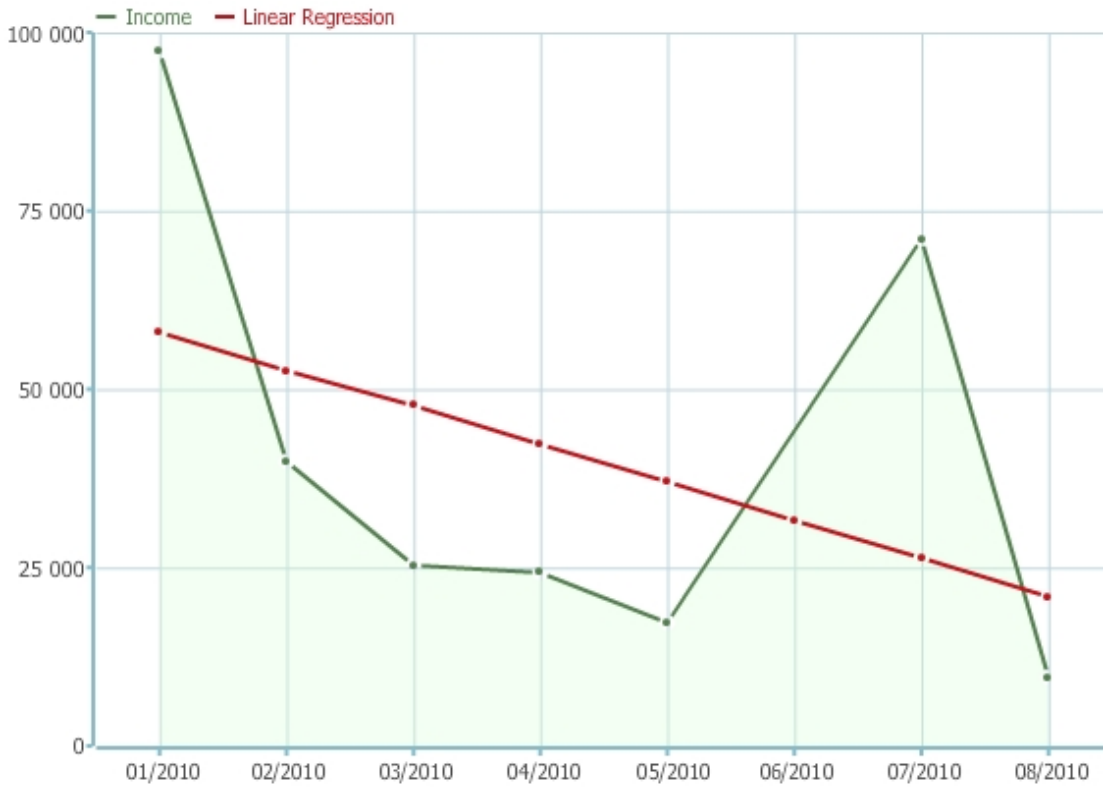
test

Sometimes we have just some sections of indicator values in time, but we want know (or approximate) missing values (interpolation), future values of indicator (extrapolation) or we just simply want to explore the trend of indicator values in time. For this purpose we have prepared a set of regression functions that allows to approximate indicators values according to its existing values. There are three types of regression to use:

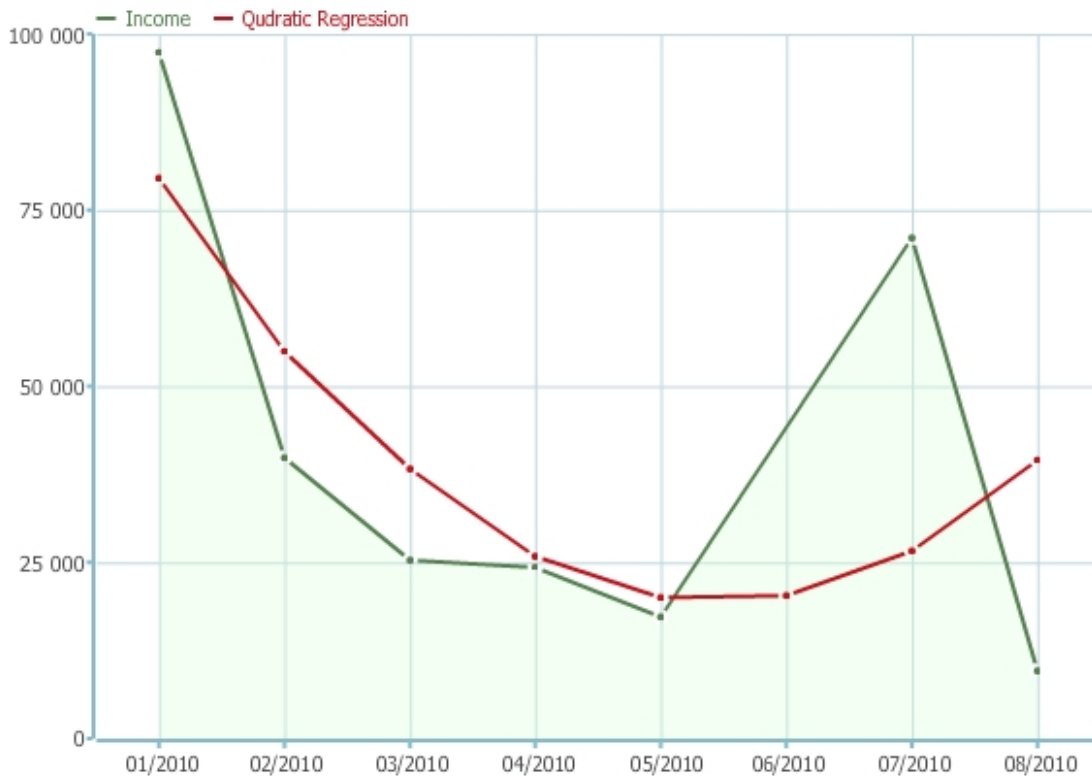
- linear (syntax `linreg('M_INDICATOR')`)
- quadratic (syntax `polyreg(2,'M_INDICATOR')`)
- polynomial (syntax `polyreg(X,'M_INDICATOR')`, where X is the level of regression from interval `<3..10>`)

All regression types can be found in formula indicator settings - category Regression. You need to fill-in the code of source (approximated) indicator into all three types of formula. In case of polynomial regression you may also change its level. Regression indicators may be used in tables and charts. In case of charts we recommend to use line charts to get clear visual interpretation of regression. Linear regression will be displayed as a line, that shows increasing, decreasing or stagnation trend of indicator values:





In case of quadratic or polynomial regression the regression indicator is more flexible in term of curve shape. It's possible to say that higher level of polynomial regression causes more precise approximation of real indicator values.




### Context change formulas

Context change formulas differ from other formulas. They just change the time or drill-down level context of other formulas that may be encapsulated into braces after context change formulas. All these formulas contain parameters (time unit or drill down level) that must be filled

between brackets and field for another functions/formulas between braces. All the functions and formulas between braces will be interpreted in changed context. Particular context change formulas and their meanings are following:

- prev(YEAR){} - change of time context to one year earlier
- prev(QUARTER){} - change of time context to one quarter earlier
- prev(MONTH){} - change of time context to one month earlier
- prev(WEEK){} - change of time context to one week earlier
- prev(DAY){} - change of time context to one day earlier
  
- next(YEAR){} - change of time context to one year later
- next(QUARTER){} - change of time context to one quarter later
- next(MONTH){} - change of time context to one month later
- next(WEEK){} - change of time context to one week later
- next(DAY){} - change of time context to one day later
  
- aggregatePrevLevel(1){} - change of drill-down level context to upper level (numeric parametr represents number of levels)

 **ADVICE:** You may simply enter only the first letter of time unit (in case of time context changing formulas) instead of their full names (Y,Q,M,W,D), for example prev(Y){}

### members()

members() function enables computing the formula for desired drill-down member values.

#### Syntax

```
members( '[L_LEVEL1_CODE=VALUE][L_LEVEL2_CODE="VALUE1,VALUE2"]', { expression } )
```

Formula defined as an expression will be evaluated for defined drill-down member values only.

### membersSum()

membersSum() function allows to realize calculation of formula indicator for each particular member alone (instead of their aggregation).

Example: We have two indicators: "Count" (values 1, 2, 3) and "Price" (values 10, 5, 20). We explore indicators for three different members (Prague, Pilsen, Brno) and also aggregated for whole Czech republic (CZ). Our aim is to create indicator, that multiplies indicator "count" by indicator "price". If we would type a simple formula M\_COUNT\*M\_PRICE a problem will occur - there would be wrong values on aggregated levels (CZ) and in "Total" row. It's caused by the multiplication of aggregated levels. Sum of "count" (6) has been multiplied by sum of "price" (35), which displays value 210 on aggregated levels. But we want to display real value according to base members (1 \* 10 + 2 \* 10 + 3 \* 20 = 80).

Therefore we use membersSum() function, which allow us to calculate this properly. We just enter the multiplication formula into the function between braces. You can compare simple multiplication and membersSum() function values in the table below:

Count		Price		Count * Price		membersSum(M_COUNT*M_PRICE)	
	2011		2011		2011		2011
	II		II		II		II
	April		April		April		April
	05		05		05		05
CZ	6		35		210		80
Brno	3		20		60		60
Pilsen	2		5		10		10
Prague	1		10		10		10
Total	6		35		210		80

Remember that you may alternate this function for some other aggregations, like:

- membersAvg
- membersMin
- membersMax

#### Syntax

```
membersSum( { expression } )
membersSum( 'L_LEVEL_CODE', { expression } )
```

### Time defined by formula

We have mentioned that it's possible to define own time intervals in table by formulas. This settings is available in table settings - time - insert formula. There is possible to use the same principles as in the case of indicators creating. However there are some limitations and specific functions, because we specify time interval and not indicator.

### Unsupported functions

- prev(), next()
- cumulate()
- dateDayOfMonth(), dateDayOfWeek(), dateDayOfYear(), dateMonth(), dateYear()
- contextDay(), contextWeek(), contextMonth(), contextQuarter(), contextYear()

### Function indicator()

Function indicator() returns indicator from the context of the row of current table. This function returns also values of formula defined indicators (defined in report).

Example:

`value(actualYear(),'actualMonth - 1m - 1d',indicator())`

This formula returns value of indicators located in table since the first day of this year to the first day of actual month. If we would use this formula on 8 Mar, 2011 it would return values of displayed indicator from 1 Feb, 2011 to 1 Mar, 2011.

	2011	<code>value(actualYear(),'actualMonth - 1m - 1d',indicator())</code>
	I	
	January	
Income	290,169.93	290,169.93

### Other formula features

#### Connection of strings by formula

In formulas it's possible to use whole string of text as a variable. This could be for example useful for combination of indicator codes:

Example - our aim is to connect two different text strings, which create code of income indicator, and display this indicator. Therefore we prepare a simple formula:

`def a = 'M_'`

`def b = 'INCOME'`  
`return value(a + b)`

#### Getting of member values

If you want to get a name of actual member's name, you can create formula defined time interval in table containing memberValue() function. This function returns the value (name) of actual member in row (or column in case that attributes are located on horizontal axis).

	2010		memberValue()
	I		
	January		
	01		
Income	+ Berlin	7,370	Berlin
	+ Paris	1,445	Paris
	+ Pilsen	3,500	Pilsen
	- Prague	10,510	Prague
	+ Matthias Daumann	4,000	Matthias Daumann
	+ Robert Glaser	6,510	Robert Glaser

#### Setting and getting of parameters in formula defined indicator

Function set('parameter', value) allows to save value of parameter into the time interval defined by formula. If this time interval is crossed by the

indicator defined by formula, it's possible use get('parameter') function to get and use value of saved parameter. Example:

Time interval formula:

Indicator formula:

Result:

		2010		
		I		
		January	February	January, February SET
Income	Berlin	17,968	8,088.07	26,056.07
	Paris	36,778	3,080	39,858
	Pilsen	3,750	18,809.24	22,559.24
	Prague	26,590	9,880.46	36,470.46
GET	Berlin	not set	not set	26,056.07
	Paris	not set	not set	39,858
	Pilsen	not set	not set	22,559.24
	Prague	not set	not set	36,470.46

We used set() and get() functions to connect strings of indicator "Income" code (M\_INCOME). We have set string 'INCOME' into parameter of time column and get it in the formula indicator. Then we have just connect this string with "M\_" prefix and display value of this indicator in selected time interval.

## Report Variables

Values displayed in reports can be modified by report variables. These variables are valid only for currently opened report. Each report variable can be set as text or date type. Text can also contain numbers to be further calculated within indicator formulas.

Only report author or report editor can create report variables and their default values. Every other user that has access to the report, can change report variable value also, however this ad-hoc value does not overwrite default value specified by author or editor and is valid only for his session (until the user logs off).

\_Note: Report variables could also serve as a basic forecasting tool. An analyst then changes report variables values and instantly see recalculated values displayed in the whole report and therefore the rate of influence according parameter change.\_

CashFlow
Actions
Add content
Share / Export
Publish

More...

To: 31.3.2010 CZK\_USD\_rate: 20.00 Save Hide

Copy report  
Remove  
Variables

CF1

		Income		Outcome	
Period	2010	I			
		January	Total [USD]	316,497.93	20,593.00
			Total [CZK]	6,329,958.60	411,860.00
		February	Total [USD]	0.69	31,471.00
			Total [CZK]	13.80	629,420.00
		March	Total [USD]	45,236.69	87,889.00
			Total [CZK]	904,733.80	1,757,780.00
Sum			Total [USD]	361,735.31	139,953.00
			Total [CZK]	7,234,706.20	2,799,060.00

To: 31.3.2010 From: 1.1.2010 CZK\_USD\_rate: 20.00 Save Hide

CF1

Period	2010	I	January	Total [USD]	Income	316,497.93	Outcome	20,593.00
				Total [CZK]		6,329,958.60		411,860.00
			February	Total [USD]		0.69		31,471.00
				Total [CZK]		13.80		629,420.00
			March	Total [USD]		45,236.69		87,889.00
				Total [CZK]		904,733.80		1,757,780.00
Sum				Total [USD]		361,735.31		139,953.00
				Total [CZK]		7,234,706.20		2,799,060.00

**Variables**

Name	Type	Default value	Value
To	Date	1.1.2010	31.3.2010
From	Date	1.1.2010	
CZK_USD_rate	Text	20.00	

Insert

Show variables settings panel to report users

Save

### Selected indicator details

Name	Total [CZK]
Color	Default
Formula	<u>@CZK_USD_rate</u> * M_TOTAL_USD
Unit	
Format	#,##0.00
Rounding	Classic (half up)

## Dashboards

This chapter is focused on workspace functions. Every user has own workspace. It's possible to manage more workspaces by one user or also public one of those workspaces. Features, described in this chapter, are available via "Workspaces" button in upper blue application menu.

Workspaces are designed for many different purposes. It's possible to

- insert and collect views from different reports on one workspace
- use specialized dashlets for navigation among reports
- examine news dashlet containing information about users' actions
- manage multiple workspaces
- public one or more of your workspaces on the web
- adjust appearance of the workspace



### 2.4.1 Insertion of views

### 2.4.2 Dashlet adjustment

### 2.4.3 Other dashlets

### 2.4.4 Desktop management

### 2.4.5 Publishing dashboards on Mercato

## 2.4.1 Insertion of views

- Views on desktop
- Placement of a view from report on dashboard

- Using of view dashlets

## Views on desktop

Desktop is a flexible tool, which allows to create contain depending on users preferences. It's possible to change the content of your desktop or adjust its appearance as well. The important thing is that you can insert more views from different reports (created from different data sets) into one single dashboard (desktop consists of one or more dashboards - see chapter 2.4.4). Each dashboard contains of two types of dashlets:

- view dashlets (containing views and custom contents from existing reports)
- other dashlet (with a specialized function such as navigation among reports - described in chapter 2.4.3)

This chapter is focused on the first type of dashlets. This use case is typical for users with "report editor" roles. It's possible compare views from different reports next to each other or just display most actual or even your long-term favourite views. Of course it's possible to combine this use cases. It's only up to you, how you will use this tool.


## Placement of a view from report on dashboard

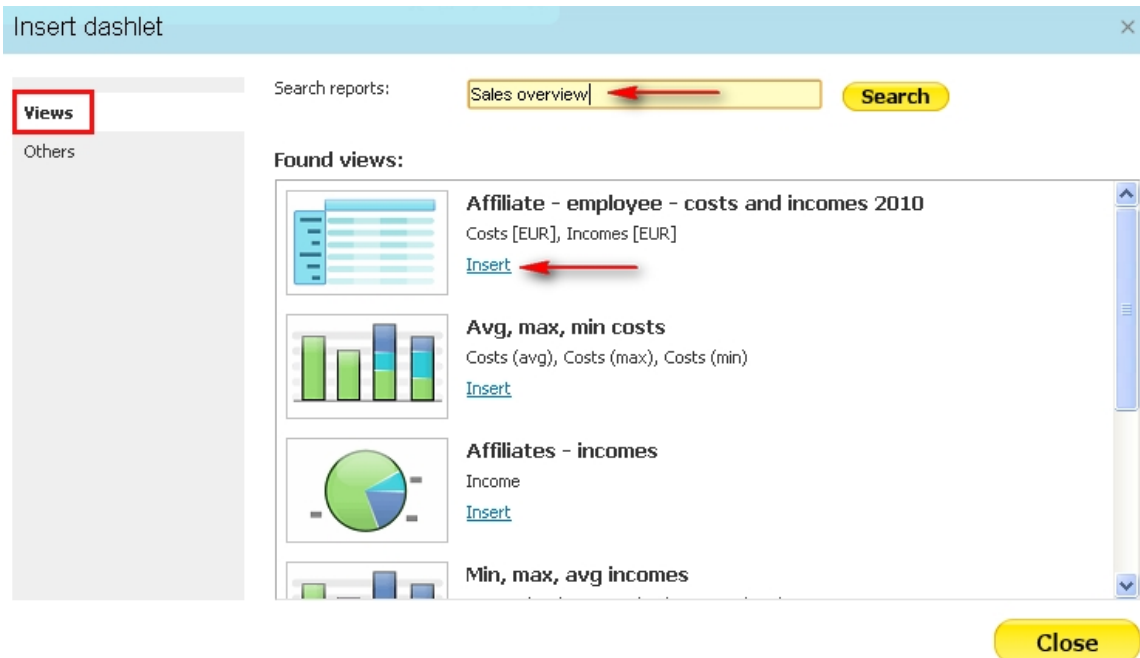
There are two ways, how to transfer a view from report to your dashboard.

- directly from the report screen through the star symbol above particular view or by "Publish" button in the upper action menu (there you can move more than one view on the dashboard in a single step)
- from the desktop screen through the "Insert dashlet" link (in the upper right corner)



In the first case you have to specify target dashboard (because one user can have more than one dashboard). In the second situation you should just choose a "view" from the list in the left side of the displayed window. Than find your report by the search line and choose (by the clicking on the "Insert" link) a particular view or custom content from the detailed list (which shows types of views and contained indicators) below.

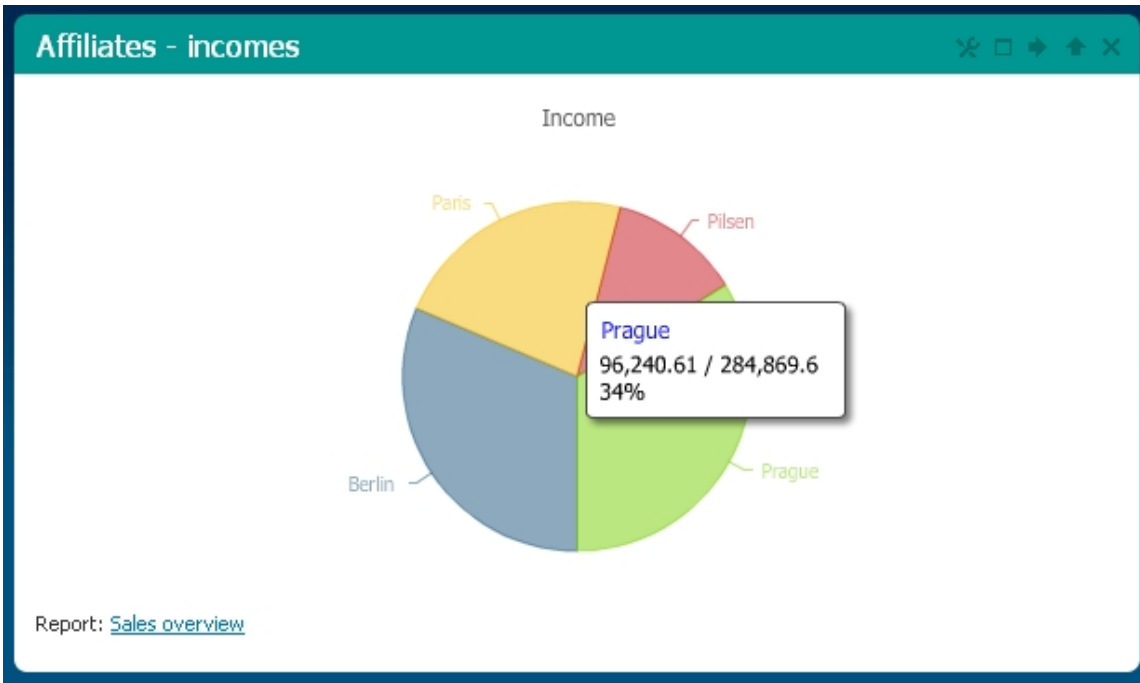
 Note, that all the views are not copied or moved. They are accessible on dashboards, but their source is still in the according reports. If you remove source report, you will also lost contained views on your desktop.



## Using of view dashlets

Desktop is not a tool designed for detailed analysis (in comparison with reports). It's purpose is mainly a presentation of analyzed data. However, there are a few view operations, which can be used during using of desktop:

- using of drill-downs in tables (by the "+" symbol next to the name of a particular member)
- using of tool tips in charts (during move of mouse cursor over the particular part of a chart)



It's allowed to use a drill-down in pivot tables, but there is a limitation - you can use this operation only when the drill-down path is predefined or created ad-hoc and saved through the floppy disc symbol. If you want to change the drill-down path (or change some other characteristic of the table), you can open the source report by the link located in the left bottom corner of particular dashlet.

A table titled "Affiliate - employee - costs and incomes 2010" showing costs and incomes for various employees. A red arrow points to the Prague row. The table is part of a dashboard window titled "Affiliate - employee - costs and incomes 2010".

	2010	
	Costs [EUR]	Incomes [EUR]
+		
+ Berlin	-116,051.34	268,287.21
+ Paris	-93,131.26	193,327.02
+ Pilsen	-38,983.48	104,272.74
- Prague	-97,313.14	288,721.83
+ Matthias Daumann	-40,333.62	125,119.62
+ Peter Grundmann	-7,374.09	38,723.34
+ Robert Glaser	-40,451.55	83,798.07
+ Saskia Roth	-9,153.88	41,080.8

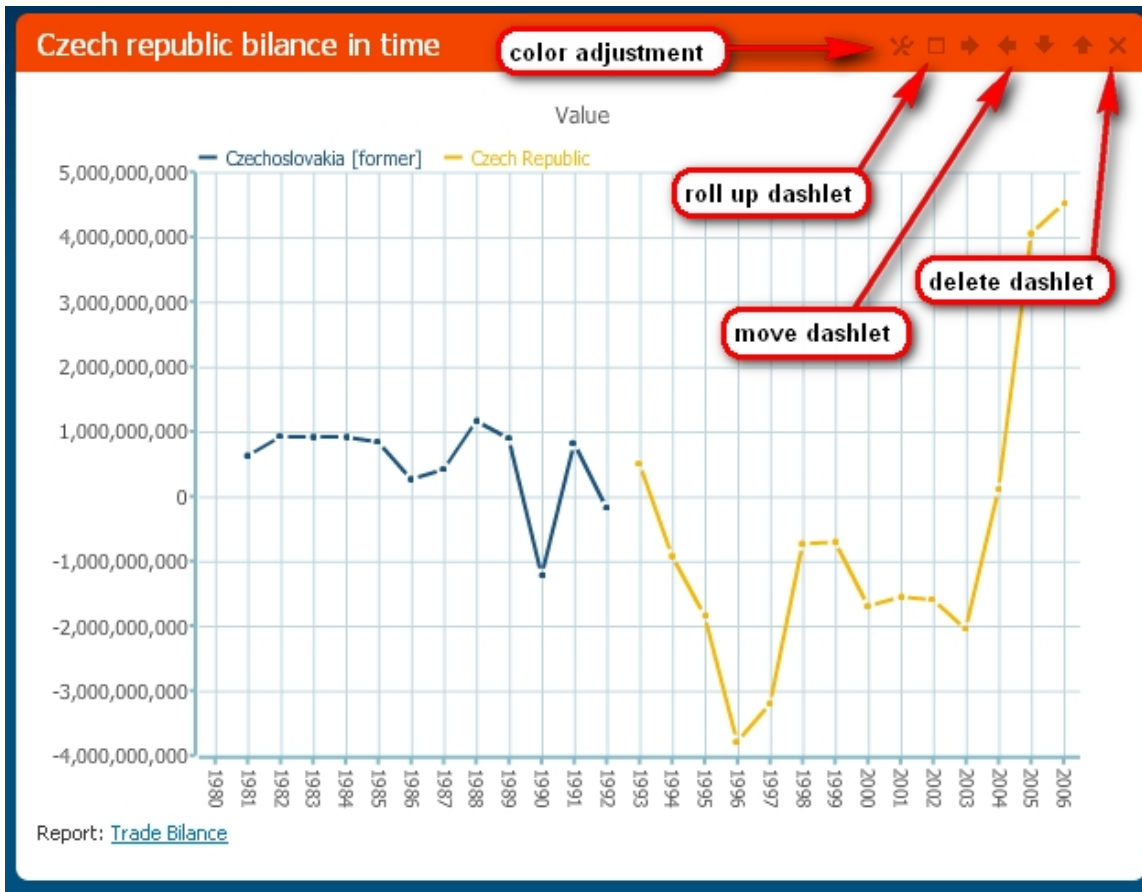
## 2.4.2 Dashlet adjustment

There is a set of ways, how you can adjust every dashlet according to your own individual needs. It's possible to:

- move dashlets across the whole workspace
- change color of particular dashlet
- roll up/recover particular dashlet
- remove (delete) dashlet from the workspace

All these adjustments can be done through the icons in the right upper corner of particular dashlet (color area). Dashlets, which are not useful at the moment, can be only rolled up into one small line or even removed from the workspace.





### 2.4.3 Other dashlets

- News dashlets
- Navigation dashlets
- Custom content dashlet
- RSS dashlet
- Help (welcome) dashlet
- Filter dashlet

#### News dashlets

News dashlet is a tool designed for watching the actual users' actions across the application. There are displayed:

- alarm activations and its reasons
- new comments in reports
- imports into data sets
- changes in reports
- changes in sharing (data sets, reports, dashboards)

Note that there is displayed information about data sets/reports related to the actual user (owned/shared) in the news.



If you want to add news dashlet to your workspace, click on the "Insert dashlet" link in the upper right corner of the workspace screen. Then choose "Other" from the left list in displayed window and then click on the "Insert" link below the "News". All other dashlets (that mean all dashlets that don't contain a view or custom content) can be added this way.

## Navigation dashlets

Navigation dashlets allow users to find their important or actual reports. If you click on the name of particular report in those dashlets, you will be immediately moved to this report. There are three types of navigation dashlets:

my reports - display reports in which is the actual user owner.

shared reports - display reports, which have been shared with actual user (with either view or edit permissions)

most visited - shows the cloud of the most visited reports. The size of letters indicates the popularity (relative number of visits) of the particular report.



## Custom content dashlet

There is one more special dashlet available in the list of "other" dashlets - a custom content dashlet. Like a custom content in the report, it's possible to create your own content (formatted text, hypertext links, pictures, video etc.) directly in the workspace without necessity to prepare it before during report analysis. It's a useful tool, because you can immediately enrich your workspace by a text or medial content and then publish your whole workspace on the web (see chapter 2.4.5).



## Custom content dashlet

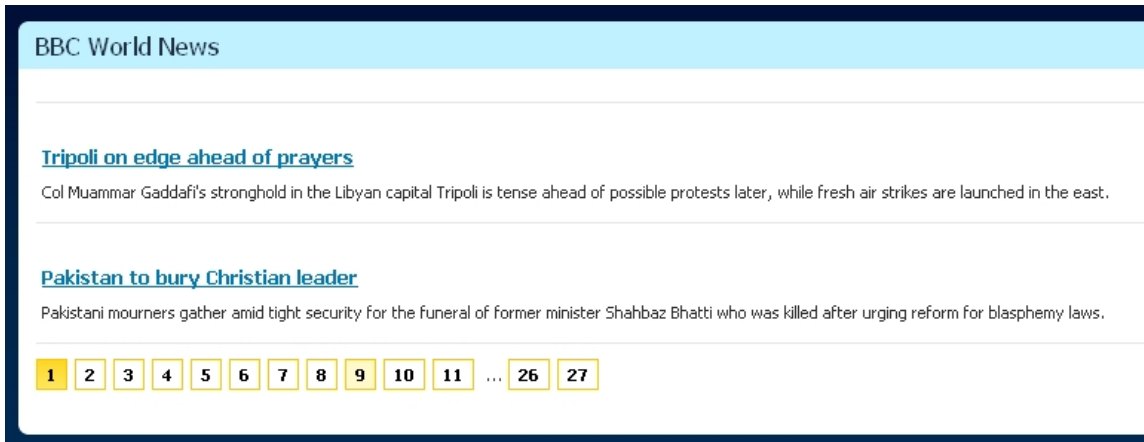
Displays custom content - styled text, videos, pictures or HTML code.

[Insert](#)

The way, how to use the custom content dashlet, is the same as the using of custom content in reports.

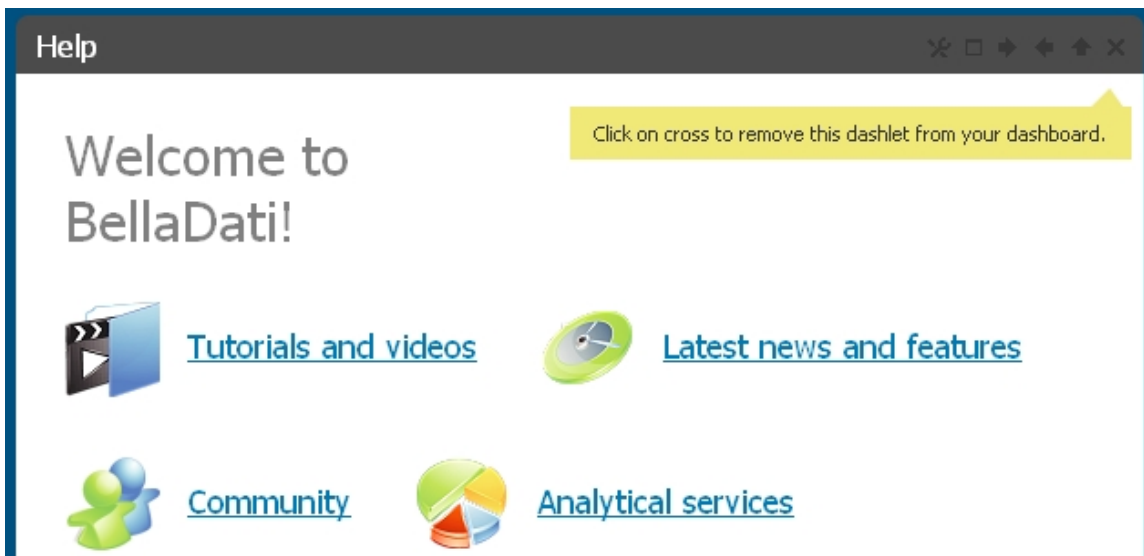
## RSS dashlet

RSS feed dashlet represents another level of multimedia content on your desktop. User friendly RSS dashlet allows you to insert any public channel directly among your charts and tables. All you need is to specify URL of target RSS source and select the number of feeds, which will be displayed in the dashlet. This dashlet will display the most actual information from selected source.



## Help (welcome) dashlet

Help dashlet is shown as a first dashlet on desktop after the first signing into your BellaDati account. It contains links to documentation, video-tutorials, company and product profiles on social networks etc.



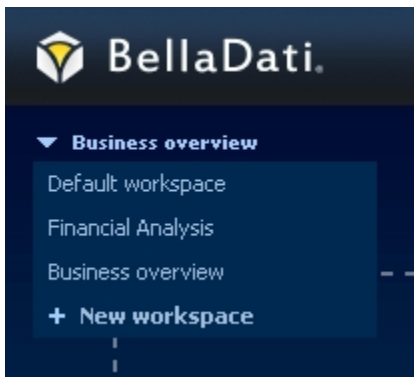
## Filter dashlet

Filters make possible to save your searching results among reports, data sets, indicators and attributes. Moreover it's possible to display any of saved filters as a standalone dashlets on your dashboard. Therefore you can use filter dashlets as a useful complement to other navigational dashlets (e.g. My Reports, Shared Reports, Most Visited etc.). You can for example create dashlet containing all the reports, which you share to other users etc.



## 2.4.4 Desktop management

One individual user is able to manage one more dashboards. The size of a dashboard is not limited, but the multiple dashboard management is a useful tool for differentiating your contents of your desktop. It's also possible to publish one of your dashboards (see chapter 2.4.5) while the other ones remains private. If you want to switch between dashboards or create a new (empty) dashboard, click on the name of your current dashboard in the upper left corner. You should see a list of existing dashboards. If one of the contained dashboards is public, it has person symbol next to its name.



In the same line you can find other useful buttons for dashboard management. It's possible to:

- edit dashboard details (name, description),
- clear all the content (delete every dashlet in actual dashboard)
- delete the whole dashboard
- attach your own file from your hard drive
- share your dashboard with other users in your domain
- publish your dashboard on public URL (Mercato)



For more information about public sharing of particular dashboards see chapter 2.4.5.

## 2.4.5 Publishing dashboards on Mercato

Mercato is a virtual marketplace with data analyses created in BellaDati service. Mercato is available on address [mercato.belladati.com](http://mercato.belladati.com). You may allow public access to your visually clear analytical outputs (charts and pivot tables) of some of your Belladati dashboards via this marketplace. All the analyses on Mercato can be easily search through dashboard names, keywords or categories.

Dashboards found: 6



### Macroeconomic indicators of European Central Bank

Key metrics aggregated for the whole European area. Key indicators include EURO exchange rate, total debt, GDP, unemployment rate, labour productivity, inflation rate and M3 monetary aggregate. Data are periodically actualized.



### USA - number of failed banks

Number of banks failed since 2000 in USA, differentiated by state, city and bank name. Comparison of time period before financial crisis and during financial crisis.

### Categories

**Economy**

[Finance](#)

[Society](#)

[Technology](#)

[Governance](#)

[Entertainment](#)

[Environment](#)

How to create your own public analysis on Mercato from your BellaDati dashboard? It's pretty easy - you just need to find and open selected dashboard to publish and click on "Publish on Mercato" link in the upper right corner (remember that this link is available unless you domain administrator denied the possibility to publish contain of your domain).

Then you just need to fill in some basic information about your public analysis into opened window (mandatory fields are bold):

- analysis title
- analysis URL (last part of address)
- description of analysis content
- key words for better searching results
- small picture that will be displayed with your analysis
- big picture that may be chosen by Mercato admin and displayed on the main screen
- categories allowing easier finding of your analysis by topic

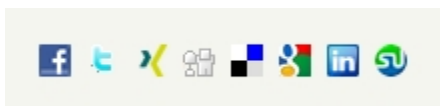
After confirming your public dashboard on Mercato will be created. You immediately visit your analysis by the new link "Visit" in the upper right corner.

Remember that your analysis will be available to public via its URL address. You may send it as a link to anybody. Note, that all the dashlets in the public analysis are displayed in full size. Navigational and other dashlets are not displayed at all. Only following types of dashlets are moved to the public analysis:

- charts
- pivot tables
- custom contents
- RSS feed dashlets

Visitors of your public analysis may explore its content but they are unable to change it in any way. They can for example use the drill-down function in tables, but only to predefined level. On the other all the changes on your source dashboard in BellaDati will be immediately propagated to the Mercato.


If you want to notify your friends, associates or other readers on Facebook, Twitter, LinkedIn etc., you just need to click to on the symbol of according social network in the corner of your public analysis (and eventually sign into the network).



You may see a preview of message that will be displayed on the network including analysis title and its description. After confirming a new information about your analysis will be displayed on your own account of according social network.


## Administration

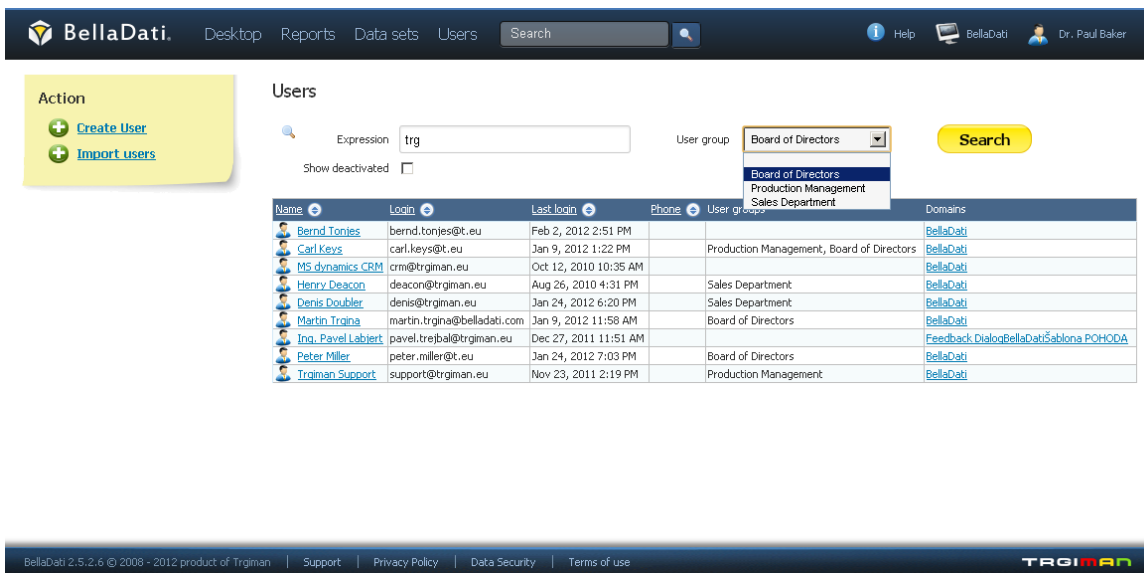
This section describes the basic BellaDati administration which can be performed via application GUI by an user with necessary [permissions and roles](#).

 If you are looking for detailed guide for system administration, visit the [Developer network](#) instead.

The following chapters are covered here:

- [Administering Users](#)
- [Administering User Groups](#)
- [Managing User Profile](#)
- [Domain Overview and Administration](#)

 You must have the Domain Administrator role to administer your domain.



**BellaDati** Desktop Reports Data sets Users Search Help BellaDati Dr. Paul Baker

**Action**

- + Create User
- + Import users

**Users**

Expression:  User group: Board of Directors Search

Show deactivated:

Name	Login	Last login	Phone	User groups	Domains
Bernd Tonjes	bernd.tonjes@bt.eu	Feb 2, 2012 2:51 PM			BellaDati
Carl Keys	carl.keys@bt.eu	Jan 9, 2012 1:22 PM		Production Management, Board of Directors	BellaDati
MS dynamics CRM	crm@trgiman.eu	Oct 12, 2010 10:35 AM			BellaDati
Henry Deacon	deacon@trgiman.eu	Aug 26, 2010 4:31 PM		Sales Department	BellaDati
Denis Doubler	denis@trgiman.eu	Jan 24, 2012 6:20 PM		Sales Department	BellaDati
Martin Trojina	martin.trojina@belladati.com	Jan 9, 2012 11:58 AM		Board of Directors	BellaDati
Ina Pavel Labjert	pavel.trejbal@trgiman.eu	Dec 27, 2011 11:51 AM			Feedback, DialonBellaDatiSabloná POHODA
Peter Miller	peter.miller@bt.eu	Jan 24, 2012 7:03 PM		Board of Directors	BellaDati
Trojina Support	support@trgiman.eu	Nov 23, 2011 2:19 PM		Production Management	BellaDati

BellaDati 2.5.2.6 © 2008 - 2012 product of Trgiman | Support | Privacy Policy | Data Security | Terms of use **TRGIMAN**

## Administering Users

Click "Users" in the main menu to display the list of users in the current domain.

This table shows:

- User name (surname, title)
- Login name (usually e-mail)
- Last login date and time
- Phone
- User group each user belongs to
- Domain (more domains are relevant for global BellaDati administrator only)

Actions available:

- Sort existing users by name, login, last login date and phone.
- Filter users by: expression (match within name or login), user group, show deactivated users
- Create new user
- Bulk user import using the CSV file

On this page:

- [Creating user](#)
- [Importing users](#)

## Creating user

# Video-tutorials

[How to manage users](#)

---

[Bulk import of users](#)

---

1. Click "Create User" in the left submenu. The popup appears.
2. Enter the information and set the options below.
3. Click "Create". New user has been created now in the actual domain.

Enter the following information in the popup (bold are mandatory):

- Degree before (title)
- Name: 2 characters minimum
- Surname: 2 characters minimum
- Degree after
- E-mail
- Phone
- Mobile phone

You can immediately assign these basic [roles](#) to the user:

- Report editor
- Data manager



Check "Send notification" option to let the new user know about his new account in BellaDati via automatic e-mail.

## Importing users

The screenshot shows the BellaDati 'Users' management page. On the left, there are buttons for 'Create User' and 'Import Users'. The main area displays a table of users with columns for Name, Login, Last login, Phone, User groups, and Domains. An 'Import users' modal window is open, showing a file upload field with 'pkc.xlsx' selected, an encoding dropdown set to 'WIN-1250', and checkboxes for 'Create as report editor', 'Create as data manager', and 'Send notification'. Below the modal, a sample CSV structure is provided: 'Name;Surname;Title before;Title after;Group;Email;'.

BellaDati offers bulk import from text (CSV) file. This is a fast way to create new users when you are migrating from an old BellaDati domain or another application and you already have the list of users.

1. Click "Import users" in the left submenu. The popup appears.
2. Select the text file to upload.
3. Set encoding: UTF-8, Win-1250, Win-1252, ISO-8859-1 or Auto
4. Optionally set **roles**: Report editor, Data manager
5. Check the "Send notification" option will let the new users know about their new account in BellaDati via automatic e-mail.

The file structure:

```
Name;Surname;Title before;Title after;Group;Email;
```

1. Name, surname and e-mail are mandatory.
2. The e-mail will serve also as login name. The password will be generated randomly.
3. When requested user group does not exist, it will be created during the import.
4. The selected roles will be assigned to all users in the list. When no role has been selected, all users will have only the common BellaDati user role.

### Next Steps

- Take me back to [administration overview](#)
- Administering [user groups](#)
- [Managing user profile](#)

## Administering User Groups

User groups serve for easier and more transparent sharing [data sets](#), [Reports](#), and [Dashboards](#) within domain.





The main advantage of using user groups is the simplification of controlling the user access to underlying BellaDati objects (reports, dashboards, data sets and data).

The screenshot displays the BellaDati user management interface. At the top, there is a navigation bar with 'BellaDati', 'Desktop', 'Reports', 'Data sets', 'Users', and a search bar. The 'Users' menu is active. On the left, an 'Action' sidebar contains a '+ Create Group' button. The main area is titled 'User groups' and contains a table with columns 'Name' and 'Description'. The table lists three groups: 'Board of Directors (3)', 'Production Management (2)', and 'Sales Department (3)'. Each group has icons for 'Roles', 'Edit user group', and 'Delete user group'. A modal window titled 'Edit group members' is open, showing a list of users: 'Henry Deacon', 'Dr. Paul Baker', and 'Denis Doubler', each with a 'Remove' button. Below the list is a search field labeled 'Name, user login:' and 'Back' and 'Add' buttons. The footer contains copyright information and the 'TRIGMAN' logo.

Click "User groups" in the main menu "Users". The list of all groups in the current domain will be displayed. You can sort the list by group name and description.

The following actions are available:

- Add or remove group users (members): Click on the group name. The popup appears. Use autocomplete for adding the new users.
- Edit group name and description
- Create new group: Enter the group name and optional description and click "Save".
- Set roles: report editor, data manager
- Delete user group: You must confirm the delete in the popup.

## Next Steps

- Take me back to [administration overview](#)
- Administering [users](#)
- [Managing user profile](#)

## Managing User Profile

Click your name in the top left menu to display your profile.

Each BellaDati user has the following information associated (bold are mandatory):

- Name: 2 characters minimum
- Surname: 2 characters minimum
- Degree before, degree (title)
- Photo
- Work details: office, position
- Phone, mobile
- E-mail

- Address: street, number, city, district, region, zip code, state



Uploading the photo is recommended since you can easily identify who has created or modified an report, data set, dashlet or imported the new data via tooltips.

**BellaDati** Desktop Reports Data sets Users Search Help BellaDati Dr. Paul Baker


### Edit User Profile

**Name**

**Surname**

**Degree before**

**Degree**

**Photo**   Remove

Update photo:  Soubor nevybrán

**Office**

**Position**

**Phone**

**Mobile**

**Email**

**Address**

**Street and address**   /

**City**

**District**

**Region**

**Zip code**

**State**

Configuration

## Preferred language

You can select locale, which defines, how BellaDati is displaying labels.

- Use browser default settings: BellaDati selects the actual language automatically according your browser's locale. Since BellaDati does not support all locales (see below), English will be the default language in most cases.
- Particular language (represented by flag): BellaDati will be always displayed in the selected language regardless the domain or browser locale.



1. Currently supported languages in BellaDati are: English, German and Czech  
BellaDati platform is however very flexible to add new languages - please contact support to get actual status of language options.
2. Language detection: German for Germany and Austria, Czech for Czech republic and Slovakia, English otherwise.

## Charts display technology

BellaDati supports two basic technologies to render charts:

- HTML5
- Flash
- Auto detection: HTML5 is preferred all time when supported browser is detected.



1. We strongly recommend to use [HTML5 technology](#) for charts rendering. Flash technology is considered obsolete and has worse performance than HTML5 technology. Therefore flash is usually necessary only for backward compatibility on those corporate PCs with only Internet Explorer 8 or older installed.  
Required browsers with HTML5 support: Microsoft Internet Explorer 9, Mozilla Firefox 5+, Google Chrome
2. Browser's compatibility with chart rendering technologies is always checked during login. A warning message is displayed when a problem has been found.
3. Changing chart render technology may require logout and login to BellaDati again to take effect.

## Actions

- User group edit: Add groups via auto-complete, remove user from groups.
- Change password: Change user's password to specified here.
- User roles: You can assign or remove the roles here.
- User groups: You can assign or remove the user from user groups.
- Password reset: Generate new password and send it to user here.
- Deactivate / Activate: Temporarily suspends or activates the user account. That user cannot login to BellaDati however all
- Delete user: Removes the user from the domain.



When an user is deleted, all his data sets will be assigned to domain administrator (he becomes its owner). All dashboards and reports owned by the deleted user will be deleted too!

## Next Steps

- Take me back to [administration overview](#)
- Administering [users](#)
- Administering [user groups](#)
- [Managing user profile](#)

## Domain Overview and Administration

Click on the domain name at top right corner of the main menu.

The screenshot shows the BellaDati administration interface. At the top, there is a navigation bar with the BellaDati logo and menu items: Desktop, Reports, Data sets, Users, Domains, Settings, and a search bar. On the right of the navigation bar, there are links for Help, the user's name (Sebastian Lozano), and a profile icon (Ing. Mf Urbankl).

Below the navigation bar, the user's name "Sebastian Lozano" is displayed, along with "Remaining days to domain expiration: 4." and a "Buy BellaDati" button. A progress bar shows "26/30" days remaining.

The main content area is divided into two columns. The left column, titled "Number of users", shows a progress bar for "1/6" users. Below this, there is a table of domain settings:

Active	<input checked="" type="checkbox"/> <a href="#">Deactivate</a>
Domain expiration date	Feb 6, 2012
Active from	Jan 7, 2012
Valid to	
Max disk usage	500 MB
Maximum users count	6
Max user dashboards per users	4
Max reports count per user	6
Max data sets count	6
Min import period	Week

Below the table, there are several action buttons: Edit, Copy template from domain..., Delete domain, Delete content, Disable web sharing, and Mark as a demo data template.

The right column, titled "Business Intelligence module statistics", shows a table of usage statistics:

Number of data sets	0/6
Number of reports	0/6
Number of dashboards	1/4
Disk space used	0.0 out of 500 MB
	0%

At the bottom of the page, there is a "Configuration" section with a dropdown menu for "Preferred language" set to "English".

The footer of the page contains the following text: "BellaDati 2.5.2.6 © 2009 - 2012 product of Trigram | Support | Privacy Policy | Data Security | Terms of use" and the Trigram logo.

The following information are displayed for the domain:

- Domain description

- Number of users: Total users count in the domain.
- Active from: When
- Max disk usage
- Maximum users count
- Max user dashboards per users
- Max reports count per user
- Max data sets counts
- Min import period

Business Intelligence module statistics:

- Number of data sets: Total data sets count in the domain.
- Number of reports: Total reports count in the domain.
- Number of dashboards
- Disk space used

Available actions:

- Buy BellaDati: If you have registered trial tariff, you will be transferred to [my.belladati.com](https://my.belladati.com) where you can purchase full version.
- Edit: Timezone change - affects the date and time displayed in BellaDati, however data itself are not affected!
- Delete content: All data sets, reports and dashboards will be deleted after confirming this action in the popup.
- Disable web sharing: Reports and dashboards will not be shareable via public links.
- Preferred language:
  - Use browser default settings: BellaDati selects the actual language automatically according your browser's locale. Since BellaDati does not support all locales (see below), English will be the default language in most cases.
  - Particular language (represented by flag): BellaDati will be always displayed in the selected language regardless the domain or browser locale.



1. Currently supported languages in BellaDati are: English, German and Czech  
BellaDati platform is however very flexible to add new languages - please contact support to get actual status of language options.
2. Language detection: German for Germany and Austria, Czech for Czech republic and Slovakia, English otherwise.
3. Language selected in the [user's profile](#) has priority over this setting.



Please note, the screenshot above is only illustrative. Domain admin and data manager has access only to fewer information about the domain.



Information about occupied space may not be exact and actual. These information are updated asynchronously.

## System administrator



This role is available in on-premise version only.

BellaDati system administrator have access to all domains in BellaDati and can perform the following additional actions:

- Deactivate / Activate: Deactivate the domain
- Switch to another domain
- Change domain limits: disk space, users count, maximum dashboards per user, maximum reports per user, maximum data sets count, minimum import period (day, week, month, quarter, year), expiry date
- Change domain description and time zone
- Delete domain: The whole domain will be deleted including all users, groups, data sets, data, reports and dashboard after confirming this action in the popup.
- Copy template from domain: You can copy the content of demo domains to this domain.
- Mark as a demo data template: The content of the current domain could be copied to another domains.



Domain limits are set automatically according registered tariff or On-Premise license.

## Application Settings and Configuration



These settings are available in BellaDati On-Premise only.

Go to menu "Settings" - "Configuration". You have following parameters:

- Maximum upload file size (in bytes): default 20MB (21000000)
- Max failed logins count: default 3

### Next Steps

- [Administrating users](#)
- [Administrating user groups](#)