User Manual

instantOLAP

Version 2.2.7
16.01.2007

Copyright

Contents

Typographical Conventions

Before you start using this guide, it is important to understand the terms and typographical conventions used in the documentation. The following kinds of formatting in the text identify special information.

Formatting convention	Type of Information
Special Bold	Items you must select, such as menu options, command buttons, or items in a list.
Brackets (<>)	Used for variable expressions such as parameters.
`Monospace font`	Marks code examples or an expression-syntax.
Emphasis	Use to emphasize the importance of a point.
CAPITALS	Names of keys on the keyboard. for example, SHIFT, CTRL, or ALT.
KEY+KEY	Key combinations for which the user must press and hold down one key and then press another, for example, CTRL+P, or ALT+F4.

CHAPTER 1:

Using the Web-Frontend

Contents of this chapter:

Starting the Web-Frontend

Opening the Web-Frontend in your Browser

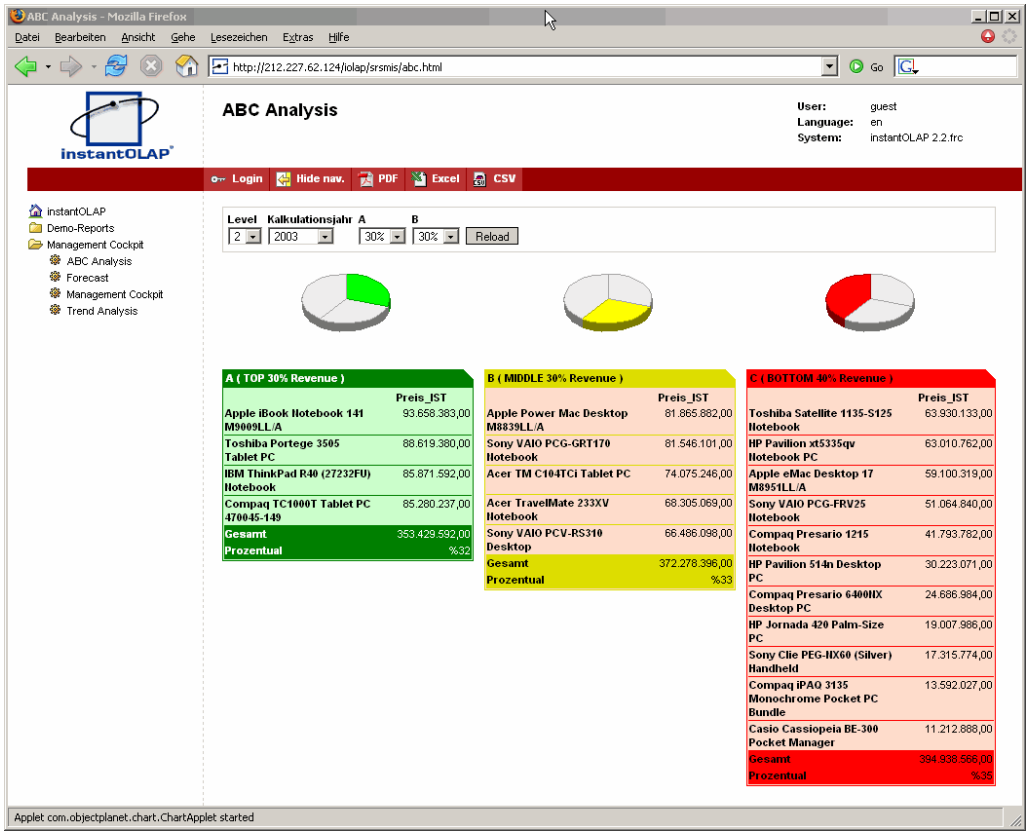
Open a browser and enter the location (URL) of your instantOLAPserver.

The URL of an instantOLAP-Server is "`http://<severname>[:<port>]/iolap`" or "`https://<severname>[:<port>]/iolap`", where <servername> must be replaced by the name or ip-address of the server and <port> must be replaced by the port of the server. If the port is 80, you can leave it out (this is the standard-port for web-applications). If your server is available only under a secure https-connection, you must use the protocol "https" at the beginning of the URL.

E.g. an URL "`http://localhost:8080/iolap`" connects to a local instantOLAP-server running on your machine on port 8080. An URL "`https://10.0.0.10/iolap`" connects to a server running on server with the IP-address "10.0.0.10".

Depending on your server's installation you will be allowed to use the application as a guest or not (the default setting of instantOLAP allows the guest-mode). If working as a guest is allowed, you will immediately see the OLAP-frontend with it's menu, navigation-bar and content-area. Otherwise, you'll be directed to the login-dialog and have to log in first.

The instantOLAP Web-Frontend



Logging in

Using the system as guest

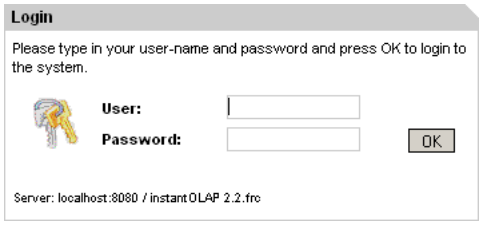
Depending on the server configuration, you may be allowed to use the OLAP-server as a guest. Guests can browse and execute the public queries. If no guests are allowed, you will immediately be asked to log in with your account. Otherwise you can use the Login button to change from the guest-account to your own account at any time.

Logging in

The login button

To log into the server with your own account, use the Login button in the upper right left of the Browser. After pressing the Login button, the login dialog will appear. Enter your account and password and press the button "OK" (or just confirm your input with the "Enter" key).

The login-dialog



If your input was correct you will be logged in. Now you can see all the folders, queries and snapshots in the navigation-sidebar to which you have to access to. If an error was found (e.g. when you mistyped your password), the error message will be displayed and you have to retype your login-information. The following errors may occur when logging in:

- **Wrong username or password:** You mistyped your username or password and have to enter the data again.
- **Too many users:** The maximum number of concurrent users was reached. Wait at couple of minutes and then try to log in again.
- **The user xxx is already logged in:** The account you want to use is a named account and is already in use. After this message appears, the system will give you the possibility to stop the current session and to relogin into the web-frontend.

Logging out

The logout button

When you are logged in, you can log out at any time by pressing the "Logout" button at upper left corner. After logging out, you will be forwarded to the login-dialog again or can use the system as a guest (only if the guest-mode is enabled for your server, see above).

Unless you log out, your current session uses one license from the license-pool of the server. If you close the browser without logging out before, your session will be terminated automatically after approx. 15 minutes (depending on your server-configuration) and the license is in use until then. To reduce the license usage for your server, you always should log out before closing your browser after using instantOLAP!

Navigation

The navigator sidebar

The navigation sidebar at the left hand side of your browser shows all folders, queries and snapshots you have access to with your account. Folder have a folder icon ( ), queries a gearwheel-symbol ( ) and snapshots a small disk-symbol ( ) left to their titles.

The navigation sidebar

To view a query or snapshot, simply click onto its row (onto the title or onto the icon left to the title).

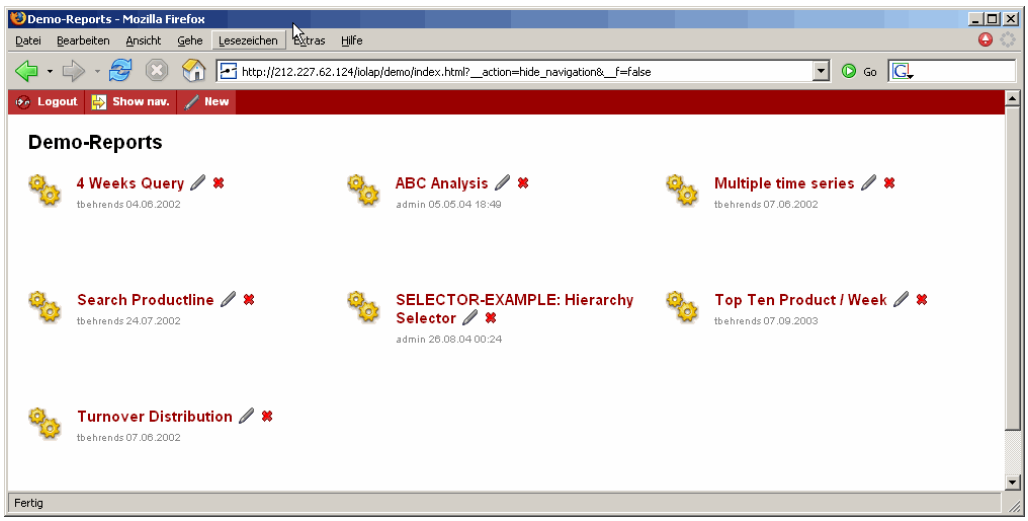
To open another folder click onto the folder-row. The currently shown folder will be closed and the new folder will be displayed.

Depending on the folder settings, there can be a default-subfolder or -query which opens automatically when you open a folder. Otherwise the folder's content will be displayed in a more detailed way inside the content are (this is called the "Index-View", see below).

The Index-View

When opening a folder, you will see it content with more detailed information inside the content area (the largest area in the Web- Frontend).

The Index-view



For each entry, there is a number of additional properties being displayed:

- **Name:** The name of the query, snapshot or folder
- **Author:** The author of a query or of a snapshot
- **Date:** The creation-date of the query or snapshot
- **Description:** The description of the folder, query or snapshot

Inside the index-view you can navigate like in the navigation-sidebar:

Just click onto a icon or title in the Index-View to open a query, snapshot or subfolder.

Hiding and showing the navigation

The navigation naturally needs some space to be displayed in the browser. For large reports you may want to hide the navigation bar in order to gain more space for the report.

The Hide-navigation button

In this case you can use the "Hide navigation" button to hide both the navigation-sidebar and the info-panel. After pressing the button, the query will be reloaded and displayed in a larger view. Only the menu bar will be visible now.

The show-navigation button

If you want to return back to the default-view with a visible navigation sidebar and info panel, use the "Show navigation" button.

The info-panel and menu

The info-panel

The info-panel displays all important information about your session and the query you are currently viewing. If no query is opened (e.g. because you opened a folder with the navigation-bar) you can only see the session-information.

The info-panel

The following information is displayed about your session:

- **User:** The account under which you are currently working. If you didn't login before, this field shows "guest" to signal you are using the guest-account.
- **Language:** The language your Browser is using System The instantOLAP version running on this server.

If you executed a query, the following additional information is displayed about the query-execution:

- **Query:** The name of the query you're viewing.
- **Model:** The model (datasource) where the query loaded it's data from.
- **Duration:** The execution-time (without HTML-transformation and data-transfer to the browser) of the query in seconds.

The menu

The menu is displayed directly under the the info-panel. Depending on your account and the query which you are currently viewing, different options are displayed. There are several groups of actions:

- **Login, Logout:** Logging in and out.
- **PDF-Export, Excel-Export, CSV-Export:** Exporting queries and snapshots to various export-formats (PDF, Excel and CSV).
- **Save Query, Snapshot:** Saving queries and snapshots.
- **Show Navigation, Hide Navigation:** Hiding and showing the navigation-sidebar.
- **Delete:** Deleting queries and snapshots.
- **Add Comment:** Add comments to a snapshots.
- **New, Edit:** Creating new or editing existing queries (for Power-users and administrators only).
- **Workbench:** Launch the instantOLAP administration tool (for administrators only).

Using queries

Opening queries

To open a query click onto it's icon or title in the navigation-sidebar at the left border.

If the query you want to open is located in a different directory, you have to open the other directory first and navigate until you can see the query in the navigation.

The navigation sidebar

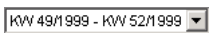
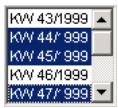
You can also open queries from the Index-View, which is visible after opening a folder.

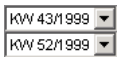
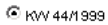
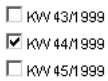
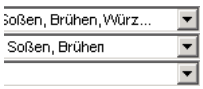
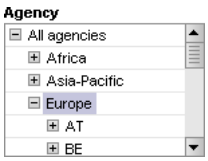
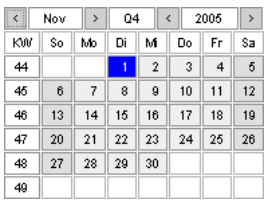
When clicking onto the query, the server starts loading and computing the data from the underlying databases. Depending on the complexity of the query, the database-speed and the amount of already cached data in the server, this may take a fraction of a seconds or a few seconds. If the query-execution takes longer than a few seconds, a progress-bar will appear and show you the progress of the execution.

When the query is finished, the progress-bar will disappear and the result of the query will become visible.

The selector toolbar

The selector toolbar is displayed on top of the report (below the title) and allows to influence the current selection of dimensions and variables. With the selection you can determine which data is being displayed in the report (e.g. for which day or product you want to see the data). The toolbar varies from report to report and may contain different numbers and types of selectors. There is a variety of possible types:

Single-Selector  	Allows the selection of a single element from one dimension. The Single-Selector is displayed as a simple combo-box with which the you can select a single element.
Multiple-Selector  	Allows the selection of one, more or no elements from one dimension. It is displayed as a listbox where you can select multiple elements from by holding the "Shift" key while selecting the desired elements.
Interval-Selector	The Interval-Selector lets your select a whole interval of elements by selecting the first (from)

	and the last (to) element of the desired range. This selector is mainly used for selecting time-intervals.
Radio-Selector 	The Radio-Selector works like the Single-Selector, but the elements are displayed as radio-buttons and are all visible at the same time.
Checkbox-Selector 	The Checkbox-Selector works like a Multiple-Selector, but the elements are displayed as checkboxes and are all visible at the same time.
Search-Selector 	With the Search-Selector you can search elements within a dimension by entering a search-pattern. The search-pattern can contain simple characters and digits and wildcards (the possible wildcards are ? and *). The wildcards ? stands for one single unknown character, the * for a number of them. E.g. the pattern "H*e" will search for all items starting with "H" and ending with "e". Note that the search-selector is not case sensitive.
Hierarchy-Selector 	The Hierarchy-Selector allows the search and selection of a single dimension-element by navigating through the dimension hierarchy.
Hierarchy-Selector Applet 	The Hierarchy-Selector applet is similarly to the previous Hierarchy-Selector but displayed as an interactive Java-Applet. This applet allows to navigate in the Dimension hierarchy without reloading and re-executing the report meanwhile. Use the + buttons to open elements in the hierarchy. By selecting an item it will be used when pressing the reload-button next time. By double-clicking an item you can select it and re-execute the report immediately.
Calendar 	The Calendar-Selector is used to select a day, week, month, quarter or year from the time dimension. The dimension is displayed in form of a classic calendar wherefrom you can select an element by clicking on it. Note that you only can select that items which the report is configured for. All other items will be deactivated.

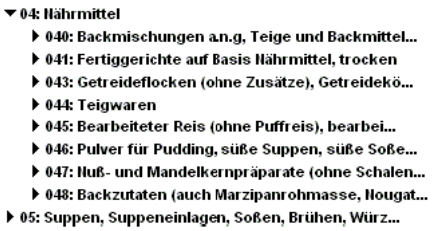
Button-Selector 	The Button-Selector changes the selection to a specific element which is defined by the query-creator (e.g. the previous day or the most selling product).
Reload button 	The Reload button reloads the query with all changes you made in the selector toolbar. Note that some selectors reload the query immediately after you changed their selection, in this case you don't need to use this button.

Drilldown and rollup

Inside tables you can drill down or roll up within the hierarchy of a dimension. Drilldown means to unfold the "children" of an element in order to view the details of aggregated values (e.g. to view the sales of single product belonging to one product group from which you drilled down).

Drilldown must be enabled for a query, otherwise you would be able to use this feature. Drilldown enabled queries show drilldown ( ) and rollup ( ) symbols at the left side of the elements, which you can use to unfold or fold the detailed view for one element by clicking onto them.

Drilldown and drilldown-icons inside table-headers



Use the arrow-symbols in the table headers to drill down or roll up inside a dimension.

Depending on the query, the child elements of a header will be displayed under or nested to it.

Links

Table cells (both header and data cells) of a query can be linked to other queries in order to allow a different or more detailed view of the data displayed by the regarding header or cell. When using a link, the target query will be opened with the dimension selection of this cell. E.g. a table cell showing the sales for the product 'A' linked to a trend-analysis will automatically open the trend-analysis for this product 'A'.

Link-icons in a table

Like in HTML, links can be texts or linked icons. In the case of a linked text, you'll have to click onto the cell-text to open the linked query. In the case of an icon, only the icon will be linked and you'll have to click onto the icon (displayed at the right border of the cell) to open the other query.

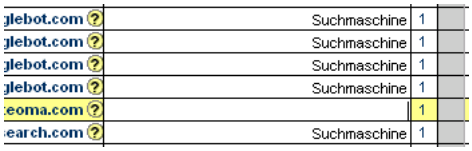
By clicking onto a linked text of a cell or a linked symbol inside a cell you can open the detail-report which is linked to this cell.

Depending on the link there can also be a popup-text which appears when moving the mouse over the link. This text can give some additional information about the link target.

Input fields

Depending on your access rights and of the current query, some fields of the query may accept data input. If you want to change the cell contents, simply click into the cell and the cursor will become visible inside the cell.

Inputfields in queries



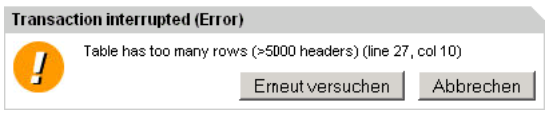
Change the content of the cells and then press the "Reload" button (see above) of the selector toolbar to upload the new data to the instantOLAP-server.

It is also possible to change multiple cells at once before sending their data back to the server.

Error messages

Whenever an error occurs (e.g. when a query is corrupted or you have no sufficient access rights to use a query), the error-dialog will open. Inside the error-dialog a detailed description of the error will be displayed, together with a "Retry" and a "Cancel" button.

The error-dialog



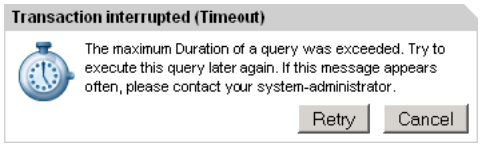
Use the button "Retry" to reload the query again. Depending on the error, the query may work after you waited some seconds or minutes, e.g. when the database-server was down and has now been restarted by the system administrator.

Otherwise use the "Cancel" button, which will return to the index of the current directory. From here you can open other queries or log out.

Timeouts

While executing a query, timeouts can interrupt the transaction. Timeouts can happen when the query is too complex (or too large) or when the server-load is too high. In this case you should wait some seconds or minutes and retry to execute the query by using the "Retry" button.

The timeout message



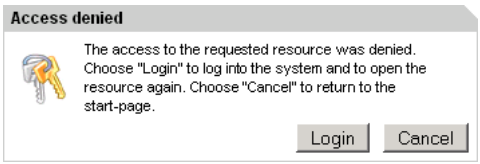
Use the button "Retry" to execute the query again. If the server-load has decreased yet, the query will be executed.

Otherwise use the "Cancel" button, which will return to the index of the current directory. From here you can open other queries or log out.

Access denied

When trying to access a report which you don't have access to or after trying to execute a query after your current session has timed out due to inactivity, a dialog saying "Access denied" will appear.

The access-denied message



Use the button "Login" to login to the system and to execute the query then. The system will remember the query-path and execute it after logging in automatically.

Otherwise use the "Cancel" button, which will return you to the root folder of the repository.

Exporting Results

How to export

The PDF button

To export the current query-result to a **PDF-file**, click on the "PDF" button in the menu-bar. After pressing the button, a new browser window will open and the PDF-file will be loaded and displayed. The Adobe PDF-viewer must be installed on your workstation in order to view PDF-files.

The Excel button

To export the current query-result to a **Excel-worksheet**, click on the "Excel" button in the menu-bar. After pressing the button, a new browser window will open and the Excel-file will be loaded in to the browser. You must have an installation of Microsoft Excel (version 97 or later) on your workstation to view the files (OpenOffice also works, but with restrictions and the result looks different).

The CSV button

To export the current query-result to a plain CSV-file (char separated values), click on the "CSV" button in the menu-bar. After pressing the button, a new browser window will open and the CSV-file will be loaded into the browser.

You can also use the CSV-format to export your report and import it into Excel without any formatting information.

Limitations

- **PDF:** Because of the page-limitation in PDF-files, the layout of queries may differ from the HTML-view. Also, some formats like the newsticker are not supported by the PDF-export and remain empty.
- **Excel**: Each block of the query will be exported as a single worksheet in Excel. Most style-attributes (colors, borders, fonts and so on) will be exported, but there is no guarantee the Excel export looks completely the same as the HTML-view of the query. Also, charts will be exported as data-tables, not as images or charts!
- **CSV:** The CSV-export will display all sections from your original query in one single CSV-file. Because CSV is a pure text-format, all layout information is dropped. For charts, the CSV-file will contain the source data of the chart but not the chart itself.

Saving exported files

You can save the exported files in two different ways:

- Open the export-view as described above and then use the menu item "Save as..." or the "Save" button in your PDF-viewer, Excel or browser.
- Use the right mouse-button in the query-view and choose the menu-item "Save target as..." to save the file directly without viewing it before.

Saving queries

Saved queries

Most queries contain selectors with which the user can select the displayed data. Some selectors have default-values and others are initially empty when the query is executed the first time. Before all selectors are configured, the user of a query usually won't see any data.

Using your own copies of queries can save some time, because the selections for all selectors without default values are stored together with the query. Whenever the user opens his own version of a query, all settings are automatically pre-configured with the new defaults.

With this feature you can create your own bookmark for queries which, by default, display the same selection as in the moment you set the bookmark - e.g. a special customer or product. You also may save multiple bookmarks for the same query, all with different settings.

Note that a bookmark will not work anymore after the original query has been deleted or moved to a different folder.

Saving queries

The save button

To save a query, use the "Save" button from the menu-bar when viewing the query. This will create a copy of the current query in your user-directory. All selectors (without default-values) of the new copy will use the current settings as new their default values.

Deleting queries

The delete button

To delete an own query, open it from the user folder. The query will show a "Delete" button in its menu bar. Use this button to delete the query (a confirm-dialog will pop up, which you have to confirm).

Snapshots

What snapshots are

All queries show the current data of their data-sources. Whenever a data-source changes, all queries using it will generate different results and show the changed data.

Snapshots are queries which are stored together with their data and are independent from their original data-sources. Snapshots never change after they have been created.

Because snapshots only store a small subset of the data, all navigation (drilldown and selection of dimension elements) is disabled in snapshots.

Snapshots can be created manually (see below) by any user or automatically by the server. Because snapshots don't consume any CPU after they have been created, they are often use to deliver reports to a large number of users.

Saving snapshots

The snapshot button

To save a snapshot, use the "Snapshot" button from the menu-bar when viewing the query.

All current navigation settings from your query will be used to take the snapshot - it will contain exactly the data you see with its selections and drilldown-states. After you saved the snapshot you will be automatically forwarded to it. All snapshots are stored in your user folder together with all other saved snapshots and queries.

Adding comments to snapshots

You may can comments to your snapshots by using the "Comment" button from the menu-bar.

The comment button

To add a comment, open the snapshot from the user-menu (it is automatically opened after taking it) and press the "Comment" button. A dialog will pop up and ask you for the new comment. Type in your comment and press "OK" to commit your input (or "Cancel" to cancel the operation).

The new comment will be appended at the end of the snapshot, together with the name of its author (your account) and the creation date.

A comment under a snapshot

Deleting comments

All comments added by the current viewer will also have a small delete-icon (a small red cross [.gif]) in front of it.

Use this icon to remove a comment from the snapshot.

You only can delete comments which have been added by yourself.

Deleting snapshots

The delete button

To delete a snapshot, open it from the user-menu and wait until is has been displayed completely. The snapshot will have a "Delete" button in its menu bar. Use this button to remove the snapshot from the user folder (a confirm-dialog will pop up, which you have to confirm).

CHAPTER 2:

Using the Workbench

Contents of this chapter:

System requirements

For the Windows installation of the Workbench you need a computer with the following equipment:

- Microsoft Windows NT, 2000 or XP
- Intel Pentium IV / 1 GHz processor or faster
- At least 100 MB of free disk space
- At least 512 MB of free memory

For the alternative Java Web-Start installation of the Workbench you need a computer with the following equipment:

- A Java Virtual Machine 1.5 or higher must be installed
- Intel Pentium IV / 1 GHz processor or better
- At least 50 MB of free disk space
- At least 512 MB of free memory

Workbench installation

Installation variants

The instantOLAP Workbench is the remote administration-tool for the instantOLAP server which is executed locally on your workstation. Therefore before you can use it, the software must be installed on your computer. There are two different ways to install the Workbench:

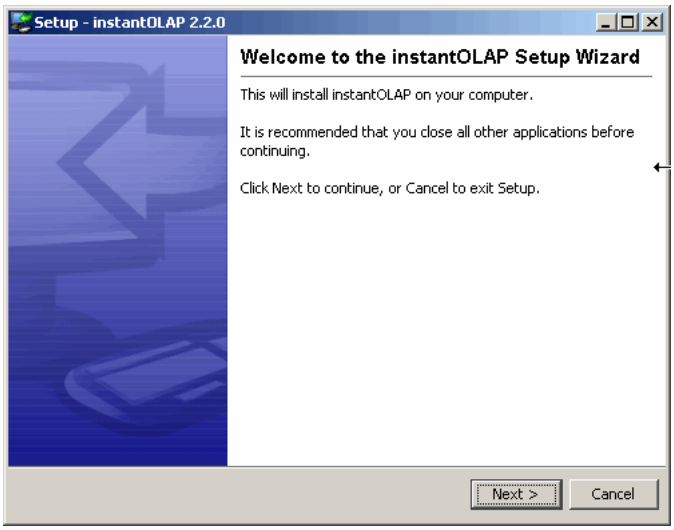
- Using the Windows-Installer
- Using Java Webstart

Installation using the Windows-Installer

This is the simplest way to install the Workbench: Download the newest version of the workbench from the instantOLAP website (http://www.instantolap.net/) and choose the "Tools only" Windows-installer.

The "Tools only" package contains only the Workbench and no server components of instantOLAP. It is delivered together with an integrated Java Virtual Machine and is a complete package, you don't need to install no further software on your computer.

The Windows installer



Start the executable installer and follow the dialog's instructions (you can use the directory and Start-Menu location). After the installer finished, the Workbench is ready to run. It can be started from the Windows-Start Menu (Programs/instantOLAP 2.x.x/Tools/Workbench).

Installation using Java Webstart

The Windows-Installer only works with Microsoft Windows. Also, the software will not be updated automatically every time if a new version is available. If you want to install the Workbench on another operation-system than Windows or use the automatic update-feature for you clients, you should use the Java-Webstart installation.

Java Webstart is an integrated installer for Java-programs which comes with the Java Virtual machine. It automatically downloads the software from the source-location and checks for new versions every time you start the program.

To use the Webstart installer, the Java Virtual Machine must be installed on your computer before you can start the Workbench download.

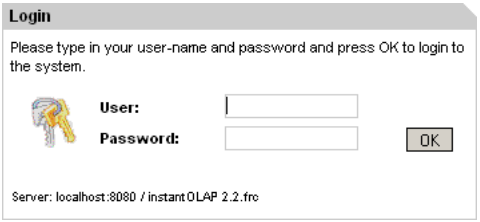
Install the Virtual Machine

To install the newest Virtual Machine, open the Java website http://www.java.com and navigate to the download of the software. The instantOLAP Workbench needs Java 1.5.0 or newer. Install the Virtual Machine before following the next steps.

Open the Web-Frontend and log in with your account

After you installed Java, start your browser and open the instantOLAP Web-Frontend. The URL of the frontend is "http://<server-name>:<port>/iolap" (the server-name and port depend on your instantOLAP installation). Then log in with your account.

The login dialog

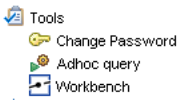


Remember that you need an account with administrator or power-user rights, otherwise you won't be able to follow the next steps.

Start the installation

After logging in, the Workbench item will appear under the "tools" menu in the navigation tree.

The Workbench item



After pressing the button, the installer will open and start downloading the necessary files from the instantOLAP-server.

When the download has finished, a dialog will open and ask you to grant access to the computer for the software (because it may need to access your locale file system in future). Choose "Yes" and accept the software.

Now the Workbench starts and the login-dialog will become visible.

Starting the Workbench again

To start the Workbench later, simply click onto the workbench button again.

This will force the Webstart installer to compare the source files with your local installation and to start the Workbench immediately (without any download) if no file

has changed. If a new version of the Workbench is available, the installer will automatically update your local copy of the software.

After you started the Workbench several time the installer will also ask you to create a desktop-icon for the software.

Logging in

After you started the Workbench (either from the Windows Start menu or using the Workbench menu in the Web-Frontend), its login dialog appears with which you can log into any server you have added to the server-list before.

The login dialog

dialog_workbenchlogin_en.gif

In the center of the dialog you can see the list of all instantOLAP servers which are already configured as targets for you Workbench. You can only log into servers which have been added to this list.

The only server, which is visible directly after the installation of the Workbench, is the server "LOCALHOST" - the instantOLAP server, which is running on your computer locally. If no server is running on you computer (e.g. because you only installed the the Workbench without server), or if you want to work on a remote server, then you must add a new server-configuration now.

Under the server-list you can choose the language for the Workbench from the drop-down list. For the version 2.2 of instantOLAP, English, German and Turkish are available as languages. By default, your Desktop language is selected (or English, if your language is not provided).

Editing the server list

With the help of the buttons in the toolbar right beside the server list you can edit the list of the servers, i.e. add new servers and edit or delete existing entries.

Adding a new server to the server list

Use the "Add server" button in order to add a new server to the list. Then a dialog opens in which you can make the configuration settings for the new server.

dialog_serverproperties_en.gif

The following entries are available for each server:

Property	Description
Name	The server name is the logical name, under which it will be displayed in the server list. You can assign any name for the server here and the name is mandatory.
Server	The address or the IP of the server, followed by its port number. E.g., after a standard installation instantOLAP runs on port 8080, but any other port is possible.

Proxy	You can use optionally define a proxy server to be used order to connect with the instantOLAP server. For this, type the proxy name or IP in the long field and the port of the proxy in the short field.
User	The user account, under which you would like to connect the server. You must use a valid administrator account here (a user with the role "iolapAdmin").
Password	Here you can type optionally the password of the user, so that this does not have to be queried with each login attempt. This is only for reasons of comfort. Note that, for security reasons, it can be better not to provide a password since each user with access to your computer can connect the instantOLAP server without knowing the password.

After you made all inputs, you can use the button of "Ok" in order to store the new settings. If you use the button "Cancel" instead, all changes are lost and no server is added to the list.

Editing existing server entries

Existing entries in the server list can also be edited.

Use the "Edit settings" button to edit the server which has selected in the list before. Then the same dialog as when creating a server opens, now with all fields being pre -filled with the current settings of the server.

Change the configuration of the server as desired and then press the button "Ok" in order to store the new settings. If you use the button "Cancel" instead, all changes are lost and the server remains unchanged.

Deleting a server from the list

Select the server you would like to delete in the server list and use you then the button "Delete server" from the toolbar. After a safety inquiry the server will be deleted from the list.

Logging in

In order to log into a server, you must select it first from the list of the servers with the mouse. Then use the button "Login" at the lower edge of the dialog. Alternatively you can login by double clicking an entry with the mouse.

If the server-configuration does not contain a pre-configured password, the password for the server will be requested. Otherwise this step is skipped you will see, if no errors occurs during the login, the desktop of the Workbench next.

However if an error occurs during the login, e.g. because the server is not reachable, an error message will be displayed in the login dialog and you can try to login again. e.g. after changing the server configuration or solving the network problem.

Installing a first license when logging in

If there is no license installed on the server, you will be able to install a first license when trying to log on with the Workbench. After installing a first license you will able to work with the server and install further licenses.

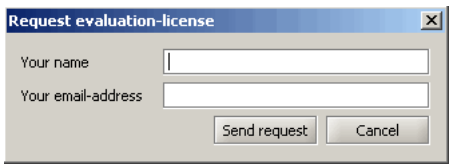
In a case of a server without a license a dialog which refers you to the absence of a license and asks you whether you would like to install a new license now. Confirm the dialog in order to arrive at the next step, entering the license number.

dialog_nolicense_en.gif

Then a second dialog appears, in which you are requested for the license number.

Request an evaluation license

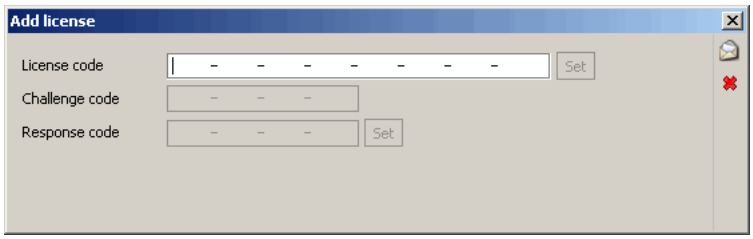
With the button "Request evaluation license" you can request an evaluation license from the instantOLAP server at the time of the license input. An evaluation license can be used for testing instantOLAP. It is valid for one month and covers 3 administrators.



After pressing the button a further dialog appears, in which you must type in your name and a valid email address. Then the dialog connects the instantOLAP webserver which will immediately send an email containing the new license to the provided email address. Open your mailbox now and transfer the license code into the license dialog.

Your computer must be online and able to connect the instantOLAP server to request an evaluation license. Alternatively you can also request a license manually from the instantOLAP website (http://www.instantOLAP.net).

Input of the license number



Enter the license code (which you received) in the field "License code" and press the button "Set" right from the input field thereafter.

If the license code was correct, the input field will deactivate and the dialog will ask you to unlock the license.

Unlock the license

The second step after the input of the license number is to unlock it. To unlock a license means to announce its usage to the the instantOLAP webserver.

For unlocking a license you receive a "Challenge code" from the license dialog, which contains the license number and the used IP of the server - this challenge code is displayed in the dialog after the input of the license code in the second line.

You can also unlock a license automatically in order to save the manual conversion of the challenge code into a response code. You will be asked to unlock the license automatically by the system after entering the license-code. For an automatic unlock your computer must be online and the Workbench must be able to connect the instantOLAP server (www.instantolap.net).

Some licenses, e.g. an evaluation license you received from the instantOLAP web-page, do not need to be unlocked. In this case the input of a license is finished directly after entering the license code.

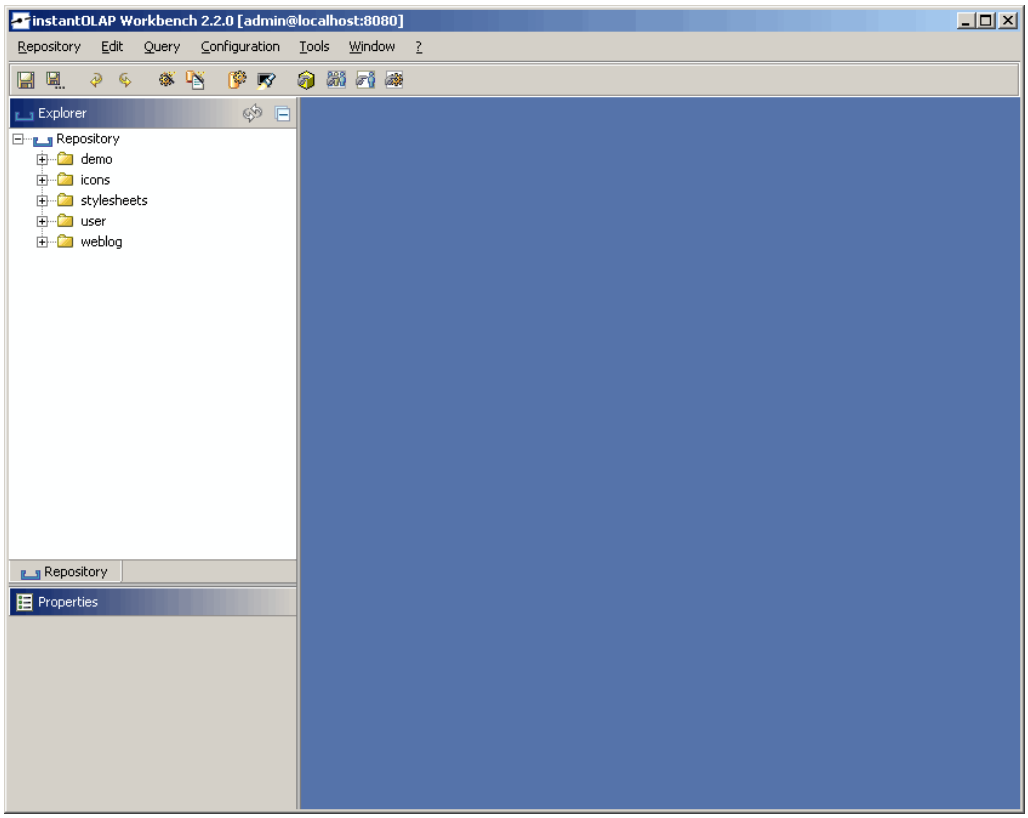
Delete the license

You can delete the license at any time with the "Delete license" button and start the input process from the beginning.

The desktop

Elements on the Workbench Desktop

After logging in, the Workbench desktop opens. The Desktop is divided into three different main areas: The Repository Explorer (at the upper left) the Property View (at the lower left) and the desktop area in the center, where all client-windows open.



Beside this areas, the Desktop also contains a menu-bar with all available functions and a toolbar for the frequently used functions.

The Menu

With the menu you can execute all functions of the Workbench. The menu is divided into different groups:

Repository	
Save	Save the contents of the current editor
Save As	Save the contents of the current editor with another file name

Save all	Save the contents of all editors
Refresh	Refresh the Repository-Explorer
Exit	Exit the Workbench
Edit	
Undo	Revert the last change
Redo	Revert the last undo
Query	
New query	Create a new query
Configuration	
New configuration	Create a new configuration
Import	
Import/From XMLA-Service	Create a configuration from a XMLA data source
Import/From DB2 Cubeviews	Create a configuration from a DB Cubeviews server
Tools	
Model query	Opens the Model-Query tool
Model manager	Opens the Model-Manager
Session manager	Opens the Session-Manager
User manager	Opens the User-Manager
Process manager	Opens the Process-Manager
License manager	Opens the License-Manager
Show/hide log	Shows or hides the logging panel
Window	
Arrange horizontally	Arranges all opened windows next to each other
Arrange vertically	Arranges all opened windows among themselves
?	

Info	Displays some system-information

The Toolbar

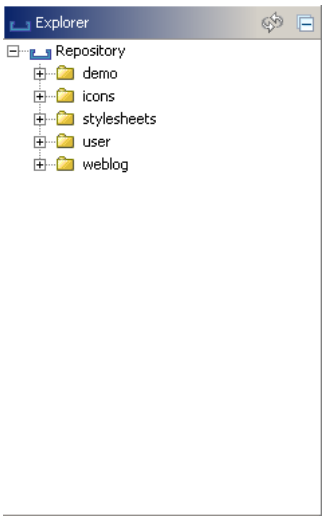
The toolbar offers the important operations from the menu as buttons. The following buttons are available in the toolbar:

	Save the contents of the current editor
	Saves the contents of the current editor with another file name
	Reverts the last change in the current editor
	Reverts the last undo
	Creates a new query
	Creates a new configuration
	Opens the Model-Query tool for querying a model directly
	Opens the logging panel on the desktop
	Opens the Model-Manager
	Opens the User-Manager
	Opens the Session-Manager
	Opens the Process-Manager

The Explorer

The Repository-Explorer in the upper, left corner of the Workbench is used for the file management, i.e. you can create, edit or delete folders, configuration, queries and other types of files here.

The Explorer



The Repository-Explorer also provides access to all important editors for the administration of folders, ie. for access-rights, automation, forwarding users etc. Most of this settings are made on the folder-level and you can open the appropriate editors using the context menu of a folder. You find a detailed description of the explorers and its functions in the section Working with the repository.

Working with the repository

The Repository-Explorer is used for the administration of folders and files and for editing the several folder settings:

- The appearance of the folders (e.g. the titles, logos etc.) are edited with the Folder-Editor
- The access-rights in instantOLAP are edited with the Access-Editor
- Forwarding for users (standard queries and folders, which are opened directly after logging in and/or opening a folder) are edited with the Forward-Editor
- The automation for reports folders (the automatic and time-steered execution, storage, export or mailing of queries) is administrated with the Automation-Editor.
- Files and folders can be imported or exported with the context-menu of the repository.

Every of these editors is reachable over the Repository-Explorer with the context menu of a folder. Additionally queries and model configurations can created within a folder with the context menu.

Working with files

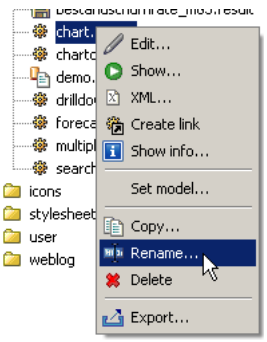
Creating files

How to create a file depends on its type: Configurations and queries can be created with the context menu of a folder or by using the toolbar of the Workbench (read the appropriate sections about the creation of queries or configurations). All other types of files, e.g. pictures, can only be created with an import into the repository (see below).

Renaming files

Existing files can be renamed, independently of the type of the file, with their context menu.

Use the context menu to rename a file



Open the context menu of a file with the right mouse button, while the mouse is over the appropriate file.

Open the menu item "Rename...". Then a small dialog appears, in which you can enter the new file name. Confirm your input thereafter with "Ok".

Note that changing the extension of a file can cause problems. E.g. if you change the extension of a configuration, whose name always with the extension ".config", then the system will no longer recognize the file as a configuration file. However, you can rename a file at any time again.

Copying files

There are two different methods to copy a file: Using the context menu of a file or with drag&drop.

Copying with drag&drop

Drag&drop is the most comfortable and simplest way of copying files within the repository.

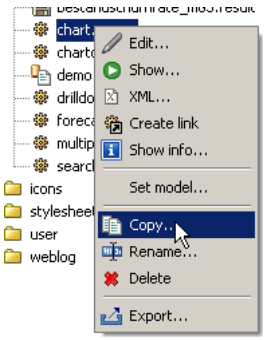
Select the file with the mouse and drag it, with pressed mouse button, into the folder, in which you would like to create the copy. Keep the "CTRL" key pressed when dragging the folder, otherwise the file will be moved instead of being copied.

With drag&drop you can only create copies in other folders than that, in which the file is located. If you want to create a copy of a file at the same place and with a different name, then you must use the context menu of the file.

Copying with the context menu

Use the menu item "Copy..." from the context menu of the file, in order to create a copy of the file in the same folder and with a different name.

Use the context menu to copy a file



After using the menu item, a small dialog appears, in which you are asked for the new name of the copy. Confirm your input with the "Ok" button and the copy of the file will be created.

Moving files

Like copying, moving of files within the repository is also possible with drag&drop. Select the file with the mouse and drag it, with pressed mouse button, to the target folder.

Do not press the "CTRL" key while dragging the file, otherwise the system will create a copy of the file instead of moving it.

Deleting files

Files can be deleted both with the keyboard and with their context menu in the repository.

Deleting with the keyboard

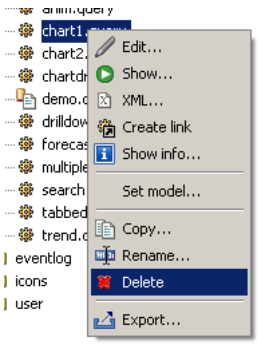
In order to delete a file with the keyboard, you must select it first, so that the background of the file appears dark.

Press the key "Delete" now in order to delete the file. Then a safety inquiry appears, which you must answer with "Yes".

Deleting with the context menu

For the deletion of a file you can also use its context menu.

Use the context menu to delete a file



Open the context menu and use you the menu item "Delete" to delete the file. Now, like when deleting with the keyboard, a safety inquiry appears which you must answer with "Yes".

Exporting and importing files

With the Workbench, you can import and export individual files or complete folders from the local filesystem to the repository or vice versa. Read the section In- and export of files and folders for more information about the import and export.

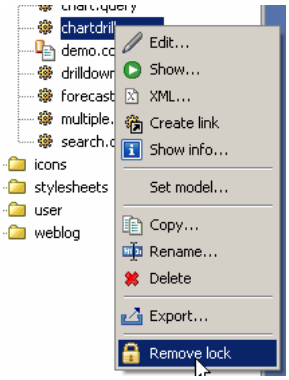
Unlocking files

When a user currently edits a file (e.g. a configuration or a query), the file will be locked until the user stops editing.

If you want to edit the file though it is locked, a message will point this out and the editor will not open until the file is unlocked.

You can unlock the file using the context menu. Use the menu item "Unlock" to delete the lock.

Use the context menu to unlock a file



Also, a file may still be locked if the user who locked the file did not close the Workbench in the regular way or if the Workbench stopped with an error. In this case, the Workbench will not have deleted the lock and you must do it manually.

Working with folders

Like in a file system, all queries and configurations are grouped and organized within folders. Folders can be nested in other folders in order to create a hierarchical organization of queries and configurations.

In the Repository-Explorer you can open and close folders, create new folders and rename or delete existing folders.

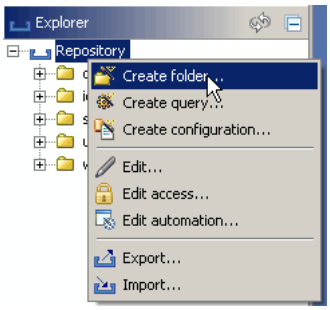
Opening and closing folders

In order to open a folder, you can click either on the small plus symbol before the listing name or double click on the name of the folder.

If the folder contains other folders or files, then these become visible after unfolding. The contents of a folder is only loaded from the server when opening the folder - therefore this can take some depending on the connection between the Workbench and the server.

Creating folders

You can create new folders using the context menu of the folder, under which you want to locate the new folder. If you want to create a new folder at the highest level of the repository then you must use the context menu of the repository node itself, which offers the same menu-item for creating folders.



Open the context menu of the folder or repository with the right mouse button and use you the menu item "Create new subfolder". Then a small dialog appears, which asks

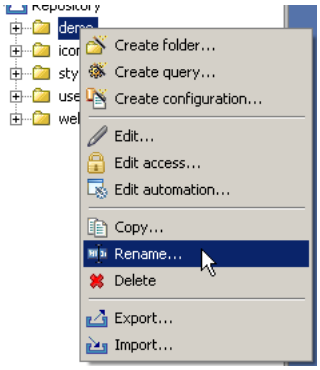
you for the name of the new folder. Enter to the new name here and confirm your input with the "Ok" button.

The name of a folder is not necessarily the name, which the user can see later in Web-frontend, because you have the possibility to assign other names for the presentation in the Folder-Editor.

After creating the new and empty folder it appears below the containing folder in which you created it.

Renaming folders

Existing folders can be also renamed. Again, you must use the context menu to rename a folder.



Use the menu item "Rename□ " of the context menu to rename a folder. Then a small dialog appears, in which you are asked for the new name. Enter the new name and confirm your input with the "Ok" button.

Copying folders

There are two different way to copy folders within the repository: Using the context menu of the folder and using drag&drop. When copying a folder, all files and subfolders contained in folder are automatically copied, too.

Copying with Drag&Drop

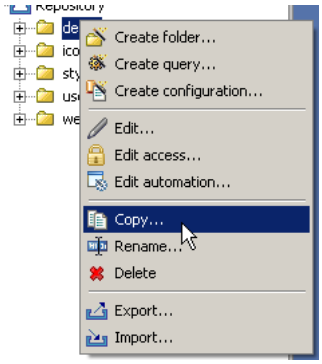
Drag&drop is the most comfortable and simplest way to copy folders.

Select the folder with the mouse and drag it, with pressed mouse button, to the folder in which you like to place the copy. Pay attention to it that you keep the CTRL key pressed when dragging the folder, otherwise it will be moved instead of being copied.

With drag&drop you can only create copies in other folders than that, in which the original folder is located. If you want to create a copy of a folder at the same place and with a different name, then you must use the context menu of the folder.

Copying using the context menu

The context menu of a folder also contains a menu item named "Copy□ ". Select this item, in order to create a copy of the folder in the same folder, but with a different name.



After the usage of the menu item a small dialog appears, in which you are asked for the new name of the copy.

Moving folders

Like copying, the moving of folders within the repository is also possible with drag& drop. Select the folder with the mouse and drag it, with pressed mouse button, to the target folder.

Do not press the "CTRL" key while dragging the folder, otherwise the system will create a copy of the folder instead of moving it.

Deleting folders

Folders can be deleted either with the keyboard or with their context menu. Note that all subfolders and files are also deleted with the deletion of a folder!

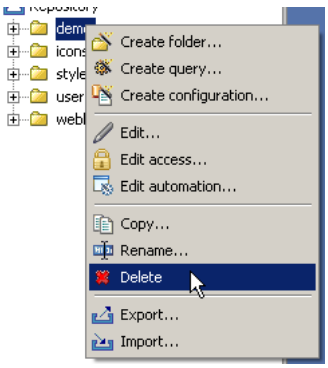
Deleting with the keyboard

In order to delete a folder with the keyboard, you must select it first, so that the background of the folder appears dark.

Press the key "Delete" now in order to delete the folder. Then a a safety inquiry appears, which you must answer with "Yes".

Deleting using the context menu

For the deletion of a folder you can also use its context menu.



Open the context menu and use you the menu item "Delete" to delete the folder. Now, like when deleting with the keyboard, a safety inquiry appears which you must answer with "Yes"

Exporting and importing folders

With the Workbench, you can import and export individual files or complete folders from the local file-system to the repository or vice versa. Read the section In- and export of files and folders for more information about the import and export.

Im- and export of files and folders

The im- and export of files and folders can be used to copy or move contents from the local file-system to the repositories of an instantOLAP-server or vice versa.

You can use the import and the export for different tasks when working with instantOLAP, e.g. for:

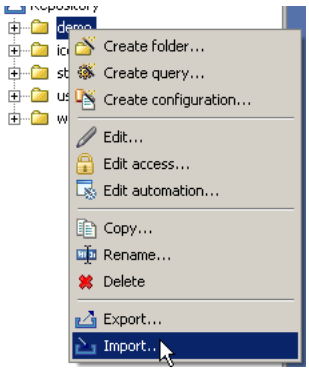
- copying configurations and queries from a development- to a production-server,
- the upload of images and other files onto the server, which are used by your queries,
- backup copies of the server files.

Importing files

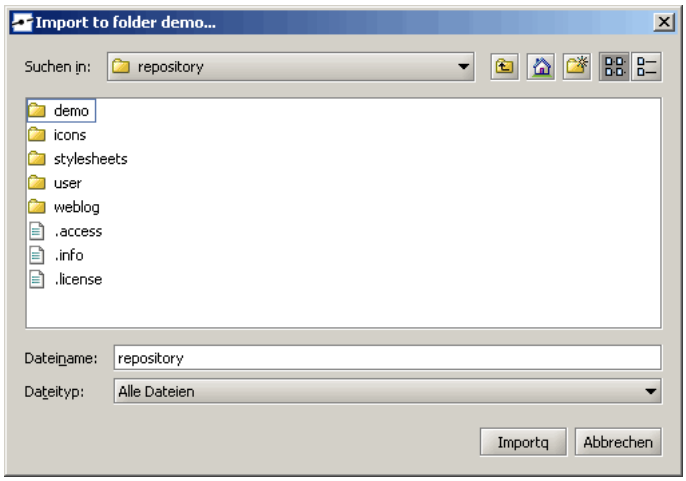
Files and folders can be imported in different ways into the repository of the server: By using the context menu of a folder or by using drag&drop.

Import using the context menu

Use the menu item "Import" from the context menu of a folder to import files into this folder.



After selecting the appropriate menu option, a file dialog appears, with which you can select a file or a folder from your local file system.



After the selection of the import source you must use the "Import" button - then all files are imported into the target folder. That can take, depending upon the number of files and the connection to the server, some seconds.

Import using drag&drop

Drag&drop is the simpler and more comfortable way to import files or folders into the repository of an instantOLAP-server.

Just drag files from the Windows-Explorer (or any other source) and drop them on the appropriate in the Repository Explorer.

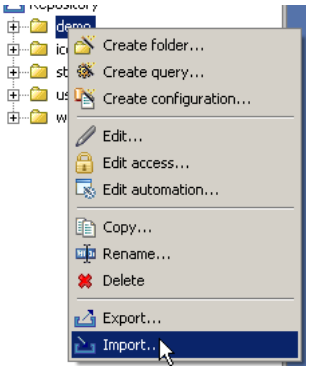
While the import, you should hold the CTRL key, otherwise you will move the files from their original position.

Exporting files

Files and folders can be exported with different methods from the Repository: Using the context menu of a file or folder and/or by using Drag&Drop.

Export using the context menu

Use the context menu of a file or a folder in order to export it to the local file system.



Then a file dialog opens, in which you must select the target folder for the export. Use the button "export" to export the files.

Export using drag&drop

Like with the import, Drag&Drop offers a comfortable alternative to the context menus for the export. Drag the file or folder to be exported from the Repository into the Windows File-Explorer or any other target (e.g. the Windows-Desktop).

Pay attention to keep the CTRL key pressed in order to create a copy of the files and not to move them, otherwise they will be disappear from the Repository.

Manging folders

The administration of folders is a central task in instantOLAP. In addition to create, rename, delete or copy folder (described in the section Working with folders), there are several other tasks and editors for folders:

- With the **Folder-Editor** you can specify the title, stylesheets, logo and other properties for a folder. You can also make folders visible with this editor (new folders are initially not visible for the user of the system).
- With the **Access-Editor** you can specify, who may see and use the contents of a folder.
- With the **Forward-Editor** you can define standard reports for a folder. Standard reports are reports which are executed automatically when opening a folder.
- With the **Automation-Editor** you can define automation-tasks for a folder. Automations are all automatically performed background actions like exporting result or sending them sending as email.

All editors can be reached by using the context menu of a folder. You can change at any time from one of these editors to the others.

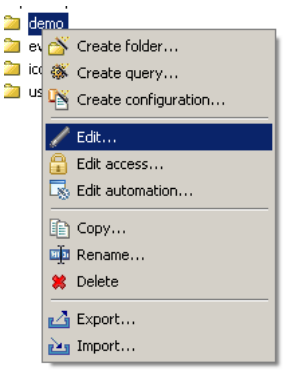
The Info-Editor

The Info-Editor allows to edit the basis properties of a folder. These properties cover:

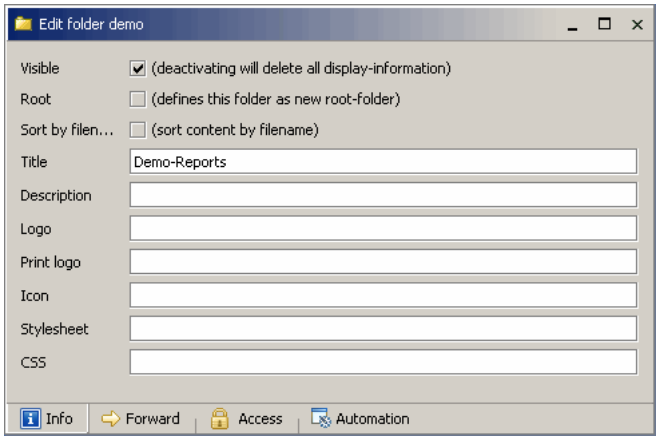
- The visibility of a folder
- The title and the description of the folder
- The logos for a folder and its subfolders
- The stylesheets for a folder and its subfolders
- The sorting-style for the folder-content
- The root-switch to make a folder to a root-folder

Open the Info-Editor

You must open the Info-Editor of a folder by using its context-menu. Select the folder and open this menu using the right mouse button.



Now select the menu option "Edit". Then the editor for this folder appears. Alternatively you can drag the folder on the Desktop area to open the editor.



Changing the folder properties

Visibility

In order to make an invisible folder visible, you must select the "Visible" checkbox. It will also automatically be activated as soon as you change any other, e.g. if you change the title. In order to make the folder invisible, you de-select the checkbox.

If you will make a folder invisibly, all fields in the editor will be deleted when saving the editor.

Root

If you select the "Root" switch, the current folder will behave as a root-folder when the user opens it. This will hide all other folders which are not sub-folders of the current folder.

Sort by filename

Usually all files within a folder are sorted by their display-names (the name is a property which can defined for each reports). By activating this switch, the files will be sorted by their filenames instead and you can manually define the order of the queries by changing their filenames (e.g you may add numbers to each query-name).

Title and description

The title of a folder is the text being displayed in the Web-Frontend of instantOLAP. The actual name of a folder is not visible for a user.

The description of a folder is optional. It will be displayed in the index page of the Web-Frontend under the title of the folder.

Logo and print logo

The Logo and the Print logo are also optional properties. If you define logos here, they become the standard-logos for all queries in this folder and its sub-folder (both for Web and PDF-output). If no logo is defined, the standard-logo '/iolap/logo.gif' is used.

Icon

You can also change the icon of a query which is displayed in the index page of the Web-Frontend. Change this property to a valid URL to change the icon of the folder.

Stylesheet and CSS

With the "stylesheet" property you can change the standard STX-stylesheet of all queries in this folder or its subfolders. With the "CSS" property you can change the default CSS-stylesheets for the queries.

Saving the properties

You can save the changes of the properties by using the "Save" button from the Workbench toolbar. Alternatively you can use also the menu option "Repository/save" or the key CTRL-S to save the currently selected window.

If you made changes in the properties and close the editor without saving the changes before, the system will ask you to store the changes.

The Forward-Editor

With the Forward-Editor you can define standard-reports or -subfolders for a folder. Such a standard-report will be executed automatically when a user opens a folder in the Web-Frontend.

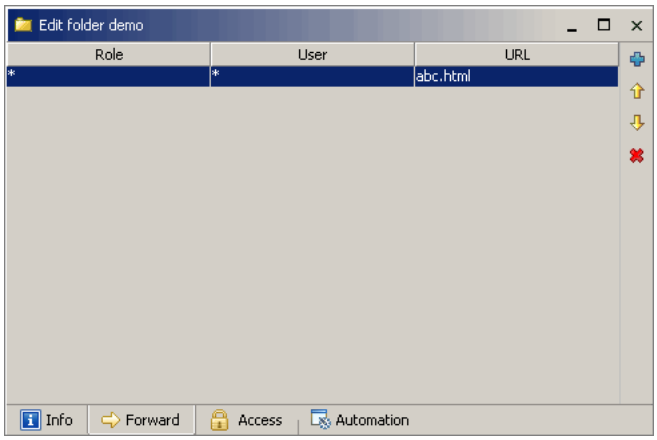
If you define a standard-report for the root-folder of the repository, this will be executed directly after a user opens the Web-Frontend or logs in.

You can also combine forwards of different folders. This means you can define a forward to another folder, which can also contain a forward to another folder or report. Then the user will be re-directed until he reaches the final report or folder.

Opening the editor

To open the editor, you can open the Info-Editor of the folder and then use the "Forward" tab of the editor to switch to the Forward-Editor.

You can open the Folder-Editor using the menu option "Edit" of a folder or drag the folder onto the Desktop area. Read the section about the Info-Editor for a detailed description.



Editing the rules

The forwards are defined with one or more rules, which each contains pattern for a role and a user. If both of these patterns apply (if the user owns one of the roles defined in the roles columns and if his user-name matches the pattern stored in the user column), then the user will be forwarded to the URL defined in the column.

For the patterns you can use the wildcards "*", which stand either for any role or for any user name. E.g. the role-pattern "iolapAdmin" combined with the user-pattern "*" would match for any user owning the role "iolapAdmin".

In the column URL you must define the URL to which a user will be forwarded. This is can be "index.html" or the name of a report combined with an extension. Usually the extension is ".html", since in most cases the user will be forwarded to the HTML output of a report. But you can also use ".pdf" or ".xls" to open the reports as PDF- or Excel-file.. In order to forward a user to the index of a folder or subfolder, the URL must be "[<folder>/]index.html".

If more than one rules exist, they are processed from the top to the bottom until the first rule matches. Therefore the order of the rules is important.

Creating new rules

You can create a new rule with the button "Add new rule". Then a new entry in the appears, with "*" as default-patterns for the user and role. The standard URL will be "index.html".

After you created the rule you can edit it. To change an entry, click on the appropriate field in the list and change its text then.

Moving the entries

With the two arrows from the toolbar you can change the order of the rules among themselves. Select one or more rules from the list and use one of this buttons. Then these rules will be moved.

The sequence of the rules among themselves is very important, since several rules could apply to a user In the upper example, both rules would apply to an administrator, since an administrator owns both roles "iolapAdmin" and "iolapUser". Because the rule, which is based on the role "iolapAdmin", comes first, this is used for the administrator.

Deleting a rule

Use the button "Delete rule" in order to delete the selected rules.

Saving the rules

They can save the changed at any using the "Save" button from the Workbench toolbar. Alternatively you can use the menu "Repository/save" or CTRL-S to save the changes if the editor is the currently selected window.

If you made changes in the properties and close the editor without saving the changes before, the system will ask you to store the changes.

The Access-Editor

With the Access editor you can determine for a directory (and its subdirectories) which user may see, execute or edit which report (or other files).

Function of the access control

The access control within instantOLAP is directory-oriented and works on the basis of rules.

Directory-orients means that you configure the access to reports and other files in the directory in which the files lie. Should a directory have no access rules, then the rules from the parent directory are automatically used, so that you can also control the access for all underlying subdirectories of a directory. Theoretically it is even possible to configure the rules for all files in the uppermost directory - the Repository.

Processing of the rules

Often there is more than one rule in a directory - then from the rules are checked from the first to the last, until a rule which applies on the current user, the desired operation and the affected file. According to the kind of the rule it permits (then it has the type "Grant") or forbids (then it has the type "Deny") the execution of the operation.

The order of the rules is very important. Should a rule apply, all following rules are no more considered by the system.

Matching of rules

Whether a rule matches or not, depends on the entries in the columns "Role", "User", "Action" and "Entry". In these columns the usernames, groups, operations and entries

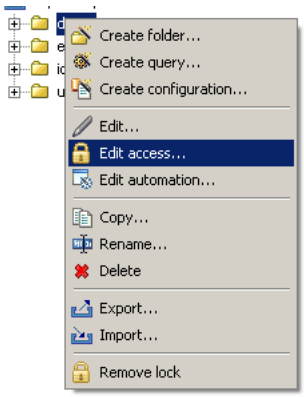
(filenames) are configured to which a rule should apply (unless you use the wildcard "*" in any of the columns, which then matches for any user, group etc).

- **User:** The name of the user who would like to perform an operation. Should the name of the user match with this field or the wildcard "*" is used here, then this column matches.
- **Role:** The name of an user-group (role). Should the current user own this role or "*" is used as a wildcard, then this column matches.
- **Action:** There are different operations whose execution can be controlled each with access rules: Execute, Save result, Save query, Delete and Admin. Also the wildcard "*" can be used to here create a single rule for all operations.
- **Entry:** This is a pattern for the name of the files to which this rule applies. You may also use the Wildcard "*" here - however, in difference to the other columns the wildcard can also match only parts of the name. E.g. a pattern "demo* " matches for all files which begin with "demo". A pattern "*.config" matches for all configurations etc.

Should you also want to configure the rules for files from its subdirectories in a directory, you must put the directory name of the subdirectory (followed by a slash "/") before the filename. E.g. a pattern "demo/" matches for all files from a subdirectory named "demo".*

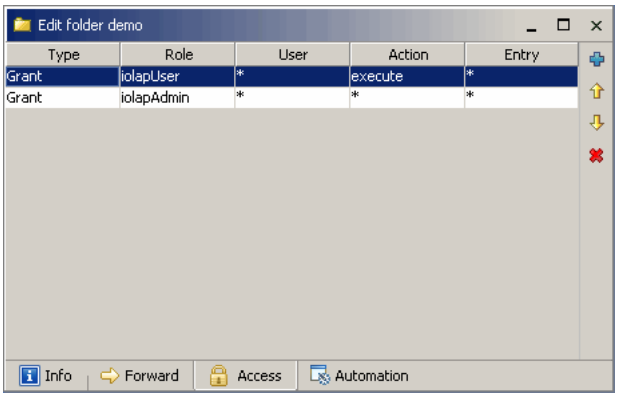
Opening the Access-Editor

Use the context menu of a directory to configure the access rules for a directory.



Open the context menu of the directory with the right mouse button and use you the menu item "Edit access.." This opens the Access editor.

Alternatively you can also open the directory editor and switch to the Access editor using the tabulator "Access". You also reach the directory editor with the context menu (using the menu item "Edit ...) or by dragging the directory on the surface of the Workbench using drag&drop.



Editing the entries

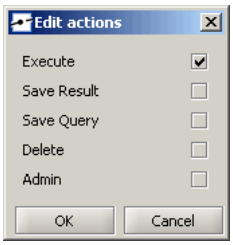
You can create new entries, move entries or delete entries. They with the tool bar at the right margin of the editor.

Editing entries

To change the values of an entry, you must click with the mouse in a column within the table. Now you can change the value and confirm the update with the Enter key.

Exceptions are the columns "Type" and "Action". Both possible values "Grant" and "Deny" for the column "Type" are available as values in a Dropdown box from which you can select the desired one.

For the column "Action" a special editor appears while clicking in it, in whom you can activate or deactivate every possible actions using a small button.



Creating new entries

Use the button "Add new rule" to add a new rule to the list. Then an empty rule is inserted at the end of the list which you can edit now.

Moving the entries

With the arrows from the tool bar you can change the order of the entries.

Choose the entries which you would like to move and press the suitable button. Then the entries are moved one position in the list upwards or down.

Deleting entries

Use the button "Delete rule" to delete the currently selected rules from the list.

Saving the settings

You can save the changed settings any time using the save button from the tool bar of the Workbench.

Alternatively you can also use the menu item "Save" or use the keyboard shortcut CTRL-S if the Access editor is the currently selected window.

If you made changes in the configuration and then close the editor without having saved before, then the Workbench will point out you to these changes and ask whether they should be saved.

The Automation-Editor

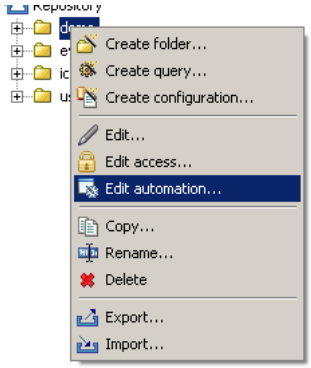
The "automation" means all automatic tasks which can be extecuted in the background within instantOLAP. The following types of tasks are available:

- The mailing of reports as an email in the PDF format ("Send task")
- The export of reports as a PDF file or Excel file in the local file system of the server ("Export task")
- The execution and saving of reports within the repository as a Snapshots which can be viewed in the Web-Frontend afterwards ("Save task")
- The execution of reports to fill the caches in order to speed up the following executions of reports ("Touch task")

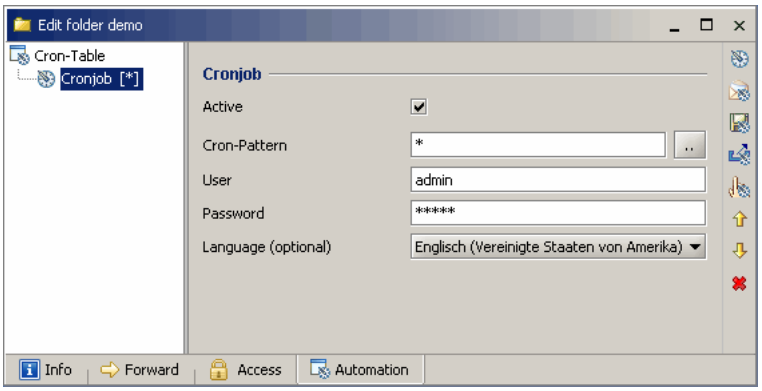
These tasks are summarized in "Cronjobs". A Cronjob is a group of tasks which are executed under a certain user's account and with a certain language at freely definable time-patterns.

Opening the editor

They can open the Automation editor using the context menu of a directory.



Open the context menu of a directory with the right mouse button and use the menu item "Edit automation..". Alternatively you can also use the menu item "Edit..." to open the Info-editor and change from there to the Automation editor using the tabulator "Automation".



The Automation editor is split into parts: On the left all cronjobs and their tasks are displayed in a tree-like structure. On the right the selected cronjob or task with all its details is displayed and can be also edited there.

Creating new cronjobs

Before you can create tasks for the automation, you need a cronjob which you can create with the button "Add cronjob" in the tool bar of the editor.

After using the button a new cronjob appears in the tree representation and the data of the job is displayed at the right side. A cronjob contains the following elements:

- **Active:** This checkbox determines whether a cronjob is active and its tasks (depending on the given time pattern) are executed.
- **Cron-Pattern:** The cron-pattern determines, how often and when a cronjob is executed. The pattern contains single parts for the month, the day or weekday, the hour and the minute.
- **User and password:** Cronjobs must be executed under a certain user's account to determine the access-rights for the execution of the tasks. Also, the used account can influence the contents of the results, e.g. if parts of the used dimensions are

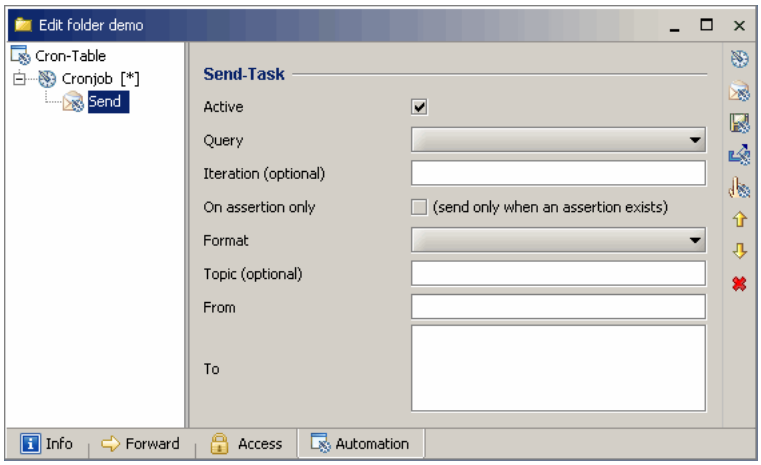
visible only for certain users. If you define no user, the reports are executed under the anonymous "guest" account.

- **Language:** The language under which the queries will be executed. This can influence the queries, e.g. the formatting of the numbers may be different. If no language is defined the cronjob will be executed with the servers default language.

Creation of a Send task

A Send task sends the report results in the PDF format to one or more several EMail addresses.

You can create a new send task with the button "Add send-task" in the tool bar. After pressing the button, a new send task will added to the currently cronjob created and the data of the task will become visible at the right side of the editor. If no cronjob exists when adding a new task, a new one will be automatically created.



For a send task you can edit the following properties:

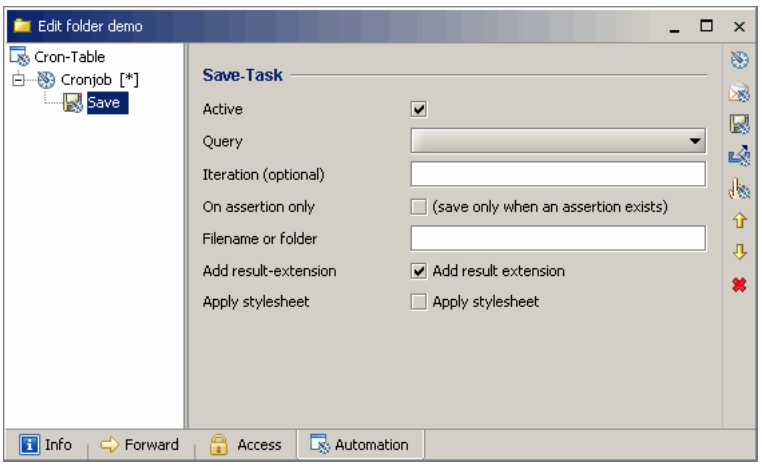
- **Active**: This checkbox determines whether the task is executed when the owning cronjob is executed.
- **Query**: For a send task you must determine which query should be executed and sent. Here you can select a single report out of all reports (which are available in the current directory or a subdirectory of it) in a Drop-Down list.
- **Iteration**: Optionally you can execute and send a report several times, always with another selection. With this property you can define a formula which e.g. has a set of products, weeks etc. as a result. Then this report is executed once for every element of the formula result.
- **On assertion only**: If the used report contains a traffic light analysis, you can use the result of this analysis to control the sending of a report. If this checkbox is activated, the report is only sent if the traffic light analysis signals an exception in the report (thus if a red, green or yellow cell appears in the result).
- **Topic**: Here you can define the optional topic for the generated mail. If your define no topic, then the system will generated one automatically.

- **To**: In this list you can define a number of receivers (their EMail addresses) which should receive the generated emails. If you want to define more than one receiver, then specify only one receiver per row in the edit field.

Creation of a save-task

A save task saves a report as a snapshot in the repository of the server. A snapshot is a XML file which contains the result of a report execution and can be viewed in the Web-Frontend of instantOLAP. Also, a snapshot can still be exported, e.g. into the PDF format, by the user.

Use the button "Add save task" to create such a task. After the creation it will be added to the currently selected cronjob and the properties of the task are displayed on the right side of the editor and can be edited. If no cronjob exists when adding a new task, a new one will be automatically created.



For a save task you can edit the following properties:

- **Active**: This checkbox determines whether the task is executed when the owning cronjob is executed.
- **Query**: For a save task you must determine which query should be executed and saved. Here you can select a single report out of all reports (which are available in the current directory or a subdirectory of it) in a Drop-Down list.
- **Iteration**: Optionally you can execute and save a report several times, always with another selection. With this property you can define a formula which e.g. has a set of products, weeks etc. as a result. Then this report is executed once for every element of the formula result.
- **On assertion only**: If the used report contains a traffic light analysis, you can use the result of this analysis to control the saving of a report. If this checkbox is activated, the report is only saved if the traffic light analysis signals an exception in the report (thus if a red, green or yellow cell appears in the result).
- **Filename**: If you define no filename in this field, it is generated automatically with a timestamp. Then every execution of this task will create a new file and no existing one will be replaced. You can use this to historize reports. If you specify a filename,

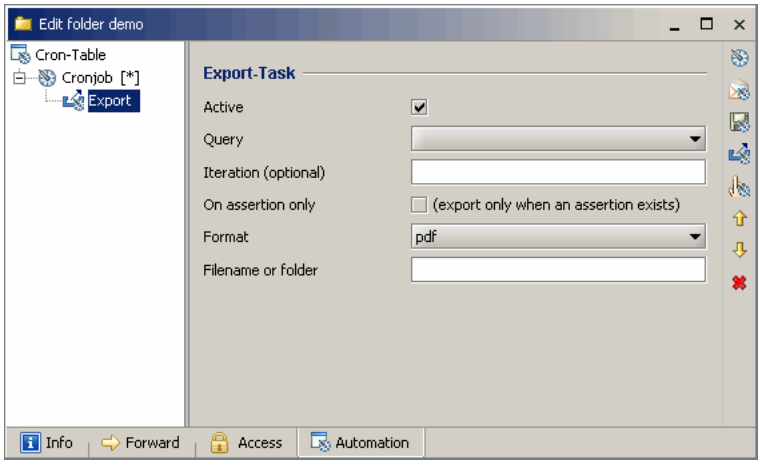
the result is saved with this name and will replace a existing file with every execution of the task.

- **Add result extension**: When this switch is activated (which is the default) the server will automatically add the extension ".result" to the saved file. Deactivate this switch if you want to save files into a different format, e.g. when using a stylesheet to transform the file to another export format. Then you can define the extension in the filename.
- **Apply stylesheet**: When this switch is activated the server will transform the result of the query using the stylesheet referred in the query.

Creation of an export task

An export task saves the result of a report as a PDF file or an Excel-Worksheet at the filesystem of the server.

Use the button "Add export task" to create such a task. After the creation it will be added to the currently selected cronjob and the properties of the task are displayed on the right side of the editor and can be edited. If no cronjob exists when adding a new task, a new one will be automatically created.



For a export task you can edit the following properties:

- **Active**: This checkbox determines whether the task is executed when the owning cronjob is executed.
- **Query**: For a send task you must determine which query should be executed and be exported. Here you can select a single report out of all reports (which are available in the current directory or a subdirectory of it) in a Drop-Down list.
- **Iteration**: Optionally you can execute and export a report several times, always with another selection. With this property you can define a formula which e.g. has a set of products, weeks etc. as a result. Then this report is executed once for every element of the formula result.
- **On assertion only**: If the used report contains a traffic light analysis, you can use the result of this analysis to control the export of a report. If this checkbox is

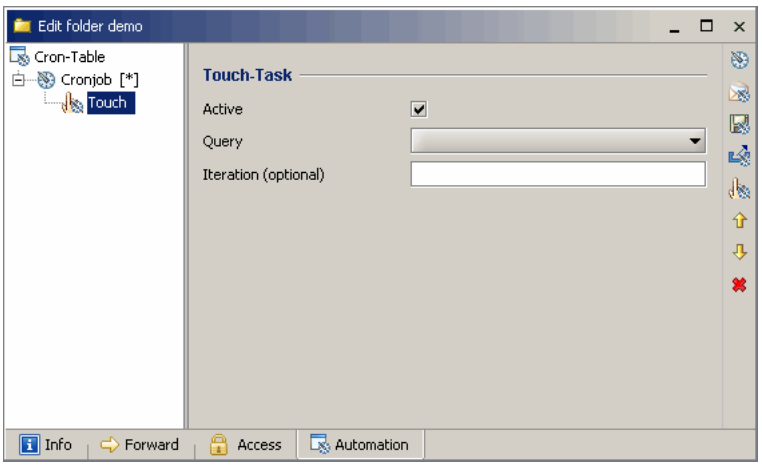
activated, the report is only exported if the traffic light analysis signals an exception in the report (thus if a red, green or yellow cell appears in the result).

- **Format**: With this drop-down box you can determine the file format of the exported report. You have the choice between the the formats "PDF" and "Excel" here.
- **Filename or folder**: For an export you must determine in which directory the exported file will be created or under which filename it will be saved. If you specify a directory here, the filename is generated automatically (with a timestamp), and with every export a new file will be created. If you specify a filename here , the report is saved using exactly this name and previous files are replaced with every export.

Creation of a touch task

A touch task is a very simple task with which just executes a report. A touch task is used to fill the caches of a model by a report execution (e.g. at night) in order to speed up reports, because the required for a report will then be available with the first access of a user.

Use the button "Add touch task" to create such a task. After the creation it will be added to the currently selected cronjob and the properties of the task are displayed on the right side of the editor and can be edited. If no cronjob exists when adding a new task, a new one will be automatically created.



For a touch task you can edit the following properties:

- **Active**: This checkbox determines whether the task is executed when the owning cronjob is executed.
- **Query**: For a send task you must determine which query should be executed. Here you can select a single report out of all reports (which are available in the current directory or a subdirectory of it) in a Drop-Down list.
- **Iteration**: Optionally you can execute a report several times, always with another selection. With this property you can define a formula which e.g. has a set of products, weeks etc. as a result. Then this report is executed once for every element of the formula result.

Editing existing tasks

You can edit existing cronjobs and tasks by selecting them in the tree representation on the left side. Then the data of the job or tasks is displayed at the side of the editor and can be edited.

Moving tasks

The order of the cronjobs and tasks also determines the order of their execution.

If you want to change the order of the executions, you can use the arrow buttons in the tool bar of the editor.

Select the entries you would like to move in the tree representation and use these buttons. Then the entries (and their execution order) are moved within the list.

Deleting rules

Existing rules can be deleted using the button "Delete selected items". First select the entry to be deleted in the tree representation and then use this button. Then the task or the Cronjob is deleted.

Tools

For the administration instantOLAP-servers a set of tools exist, with which you can perform different tasks:

- **User-Manager:** With the User-Manager you can create users and user groups as well as assign users to user groups.
- **Model-Manager:** The Model-Manager shows all existing models on the server. With the manager, models can be started, stopped or rebuild.
- **Process-Manager:** The Process-Manager shows all actively running query or other process. Current processes can also be canceled with this tool.
- **Session-Manager:** The Session-Manager shows all active user sessions on the instantOLAP-server. You can also cancel sessions with this tool.
- **License-Manager:** With the License-Manager you can administer the installed licenses on a instantOLAP-server and add and/or delete new licenses.
- **Logging panel:** The logging panel allows to view internal processes of a instantOLAP-server. E.g. the exact operational sequence of an query can be analyzed with the logging panel, i.e. for error tracing or performance analysis.

User-Manager

The User-Manager allows the administration (the creation or deletion) of user-accounts, -groups and roles. Also, individual users or whole user-groups can be assigned to the roles which are used by the access-control of instantOLAP.

The User-Manager is only available if the instantOLAP server uses its own user management. Depending on your installation it could also be possible that an external user-database is used for your instantOLAP-server, which then cannot be administered with the Workbench (e.g. a Windows Naming server or a LDAP database).

Opening the User-Manager

The user manager can be opened using the menu "Tools/User manager" of the Workbench or with the button "User manager" in the toolbar.

After pressing the buttons or using the menu, the window of the User-manager opens. The window is divided into two main areas, the user-editor and the role-editor. You can reach both editors with the appropriate tab at the lower edge of the window.

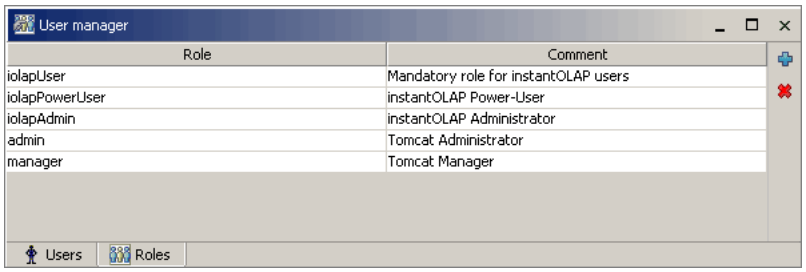
Roles

In instantOLAP, like in all Java Web applications, users are assigned to "roles" in order to simplify the administration of the access-control. All access-rules are defined on the basis of roles and therefore all users, who own a certain role, will be able to perform all actions which a granted for a certain role.

E.g. if assign a user to a role later, he receives automatically all rights of access, which are defined for this role.

Opening the role editor

If you don't see the role-editor currently, you can switch to it with the tab "Roles" at the lower edge of the window.



Creating new roles

Use the button "Add role" to add a new role to the list of all existing roles. Then a dialog opens in which you can enter the name of the new role. The name of the role can be changed afterwards.

Editing roles

You can edit new or existing roles by clicking in a cell within the table and edit its values directly. There are following values for roles which can be edited:

- **Name**: The name of a role
- **Description**: An optional description of a role which is visible only in this editor

All changes in a role become effective immediately - you don't need to save them.

Deleting roles

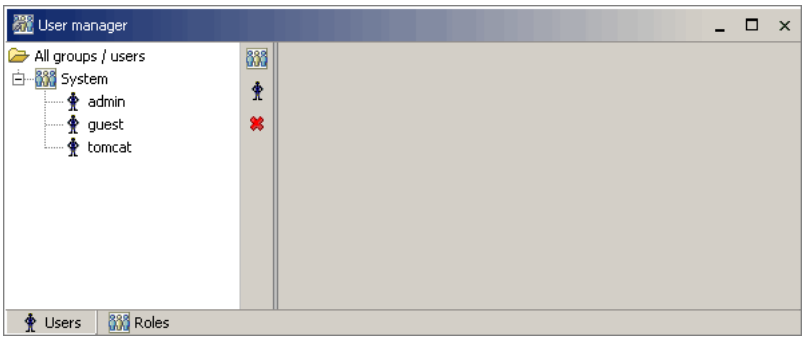
In order to delete an existing role select it first and then use the button "Delete role".

Groups

Groups are used to group users (or other usergroups) in order to make them easier to administrate. Also, you can assign roles to groups and automatically make all users of a group to members of this roles.

Opening the group editor

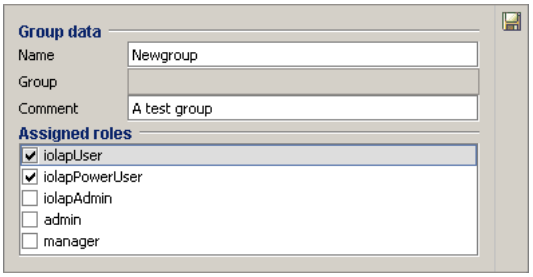
Groups can be edited in the user editor of the manager. If you don't see the user editor in the manager, you can switch to it with the tabulator of "user" in the lower edge of the manager.



Creating new groups

You can create a new user group with the "Add group" button. After pressing the button a new group will appear under the currently selected group in the tree representation at the left side of the user editor.

If no group was selected, the new group will be added under the root element of the tree. If a user was selected, the new group will be added to the group the user belongs to.



At the right side of the editor, the new group with all of its properties will be displayed now. You can edit the following properties:

- **Name**: The name of the new group
- **Comment**: This optionally comment will only be visible inside the User-Manager

Assigning roles to groups

You can also assign roles to groups. All members of this group or any subgroup of this group will automatically inherit the roles being assigned to this group.

To assign one more roles to a group you must simply activate the corresponding checkbox in the list of roles below the group properties. To delete an assignment you must deselect the checkbox of a role.

Saving groups

Use the "Save" button in the tool bar at the right side of the group editor to save the currently edited group.

Editing existing groups

You can also edit an existing group by selecting it in the tree view at the left side of the user manager. After selecting the group, its properties and the assigned rules will be displayed at the right side of the manager.

Don't forget to save the group after changing its properties, otherwise your changes will be lost after closing the manager or displaying another group or user.

Deleting groups

Use the "Delete" button in the tool bar next to the tree view in order to delete a group after you selected it in the tree.

Editing users

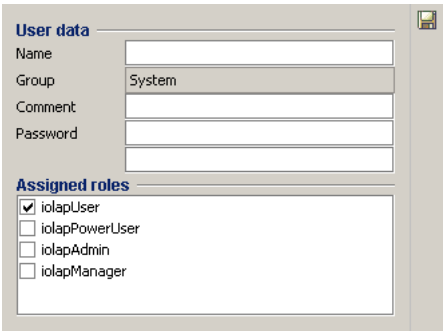
Opening the user editor

Like groups, users can be edited in the user editor of the manager. If you don't see the user editor in the manager, you can switch to it with the tabulator of "user" in the lower edge of the manager.

Creating new users

You can create a new user with the "Add user" button. After pressing the button a new user will appear under the currently selected group in the tree representation at the left side of the user editor.

If no group was selected, the new user will be added under the root element of the tree. If another user was selected, the new user will be added to the group the selected user belongs to.



At the right side of the editor, the new user with all of its properties will be displayed now. You can edit the following properties:

- **Name**: The name of the new group

- **Comment**: This optionally comment will only be visible inside the User-Manager
- **Password**: The password of the user, displayed in two input fields. If you change the password, you must retype it in the second field to confirm it.

Assigning roles to users

To assign one more roles to a user you must simply activate the corresponding checkbox in the list of roles below the user properties. To delete an assignment you must deselect the checkbox of a role.

Saving to users

Use the "Save" button in the tool bar at the right side of the user editor to save the currently edited user.

Editing existing users

You can also edit an existing user by selecting him in the tree view at the left side of the user manager. After selecting the user, his properties and the assigned rules will be displayed at the right side of the manager.

Don't forget to save the user after changing its properties, otherwise your changes will be lost after closing the manager or displaying another group or user.

Deleting users

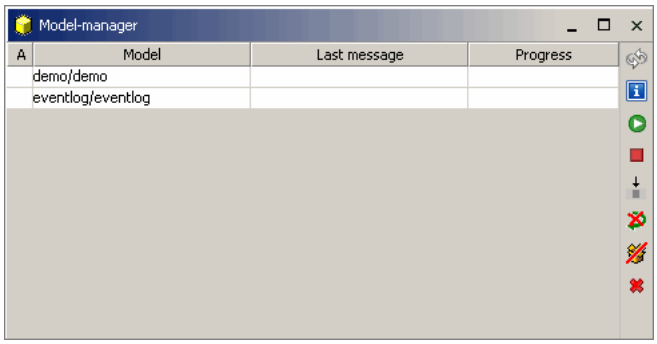
Use the "Delete" button in the tool bar next to the tree view in order to delete a user after you selected him in the tree.

Model-Manager

The Model-Manager is the central tool for the control of all models running on the instantOLAP server. It can be used to start and stop models, to rebuild their dimensions and stores or to clear the caches and persistent data of the models.

Opening the Model-Manager

Use the menu-item "Tools/Model-manager" or the Model-Manager button from the Workbench toolbar to open the manager.



Working with the Model-Manager

Informations in about the manager

For each model there a three different informations being displayed as columns in the table of the manager:

- **State**: The first column of the table shows the state of a model. There are four different possible states: Stopped (no sign), Running, Paused/Offline and synchronizing.
- **Model**: The full name of the model, including the its full path. Temporary models (which are created when editing a configuration) are also displayed here, they contain the character '~' in their name.
- **Last message**: If the start or the synchronization of a model failed, the reason is displayed in this column.
- **Progress**: Shows the progress of a model if it is currently starting.

You can a the model-info window for a single model to see more detailed information.

Refreshing the list

Use the button "Refresh" to refresh the list of the models and their states.

Opening the Model-Info for a model

Use the button "Show model info" to open a window with more detailed information for the currently selected models. Alternativley you can also double-click onto the model inside the table. See below for a description of the Model-Info window.

Starting models

With the button "Start selected models" you can start all models being selected in the model-table.

After you started a model, the progress of the startup will be displayed as a bar (with percentage) in the column "Progress" of the table. If an error occured while starting a model, the last error-message will be displayed inside the table. Otherwise, the model will switch into the "Running" state after it successfully started.

Stopping models

Use the button "Stop selected models" to stop all selected models.

Rebuilding models

With the button "Rebuild selected models" you can rebuild all dimensions of all selected models. Rebuilding dimensions means to reload all elements of a dimensions.

Clearing caches

The button "Clear caches of selected models" will clear all caches. You can also use the cache-tabulator from the Model-Info to clear single caches from the model.

Clearing stores

Use the button "Clear stores of selected models" to empty all stores of the selected models. If the models are configured to auto-rebuild, the system will start filling the stored within the next minute. You can also use the store-tabulator from the model-info to clear single stores of a model.

Deleting the persistent data of a model

The button "Delete persistent data of selected models" will delete all persistent data of a model. This includes the persistent dimensions, filecaches and stores of a model. Note that a model will automatically be stopped before the systems deletes its data.

Working with the Model-Info dialog

Opening the Model-Info for a model

To open the model-info you can double-click onto a model or use the button "Show model-info" from the toolbar of the Model-Manager.

Viewing dimensions

The first tabulator of the model-info displays all dimensions with serveral informations.

Viewing caches

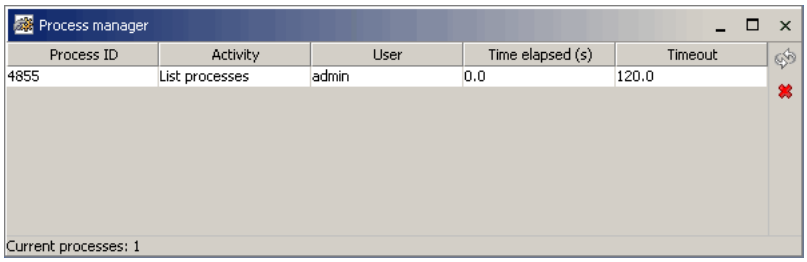
Viewing stores

Process-Manager

The process manager serves for the monitoring and control of all actually running queries, reports and other processes on an instantOLAP server.

Opening the process manager

You can open the process manager, like the other tools, either by using the menu of the Workbench or with the tool bar of the desktop. Use the menu item "Tools/Processes-Manager " or the button "Process-Manager" from the tool bar.



In the manager all actually running queries are shown as a single row in a list. For every query you can see the following information:

- **ProcessID**: The ID of the running process (every execution of a query has its own ID)
- **Query**: The executed query
- **User**: The user who executes the query
- **Time elapsed (s)**: Number of seconds which have passed since the start of the query

Actualization of the view

With the "Refresh" button of the Process-Manager you can update the view of the processes any time.

Stopping Processes

Use the "Kill process" button to terminate running processes: First Select the processes in the list which you would like to kill and then press the button.

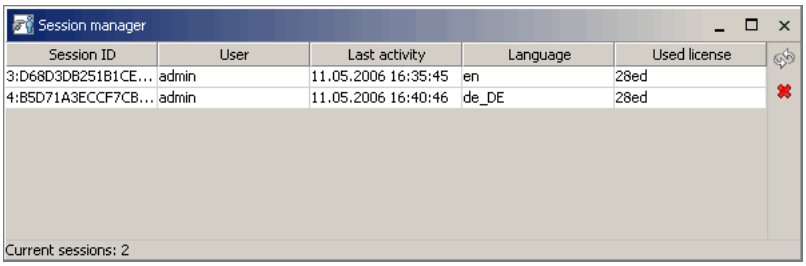
A user will receive a error message in the Web-Frontend if a his report was interrupted the Process-Manager.

Session-Manager

With the Session-Manager you can supervise all active sessions on an instantOLAP server and also kill them if necessary.

Opening the Session-Manager

Use the menu of the Workbench (the menu item "Tools/Session-Manager") or the button "Session-Manager" from the tool bar for opening the Session-Manager.



In the Session-Manager all actual sessions which run on the instantOLAP server (both for the users of the Web-Frontends as well as of the Workbench) are shown.

For every session a number of information is displayed:

- **ID**: The unique ID of the session
- **User**: The user account to which the session belongs
- **Last activity**: The time of the last user activity
- **Language**: The language of the session
- **Used License**: The number of the used license

In the license number only the first four digits of the license code (and not the complete code) are shown. With the help of the license number you can see, to which "Named License" a user is assigned.

Refreshing the list

With the "Refresh" button from the tool bar at the right edge you can refresh the view of the Sessions any time.

Killing a session

To kill a session, you must select this before and then use the "Kill session" button from the tool bar.

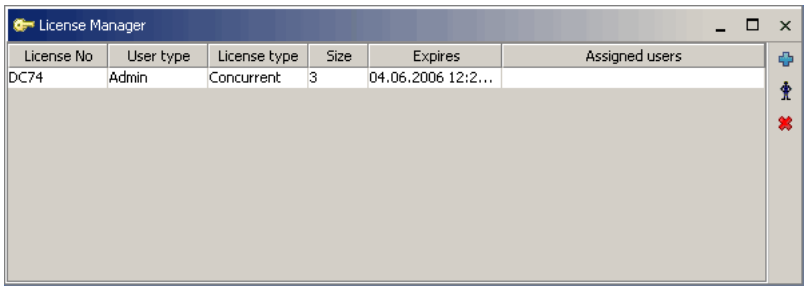
You can kill all running sessions except your own on the server. If you try to kill your own session, you will receive an error message.

License-Manager

The License manager is used to install new licenses on the instantOLAP server, to delete existing licenses and to associate certain user's accounts to "Named License".

Opening the License manager

To open the manager, you must use the menu item "Tools/License-Manager" of the Workbench menu. After that the window of the License managers opens.



In the License-Manager all licenses which are installed currently on the server are displayed in a list. The following information is displayed in each row:

- **The license number:** Every license owns an unique license number which is displayed here.
- **The user type:** A license key permits either sessions for an Administrator, a Power-User or Thin-user. Better licenses are also used for logging on "smaller" user-types (what only happens if no more enough "small" licenses are available).
- **The license type:** Licenses can be either "Named Licenses" or "Concurrent Licenses". "Named Licenses" are associated to certain users - these users have a guarantee that they can log in at any time, because they use a dedicated license. "Concurrent Licenses" are shared between all other users (who are not assigned to any "Named License"). These user may not be able to login if the amount of available "Concurrent Licenses" is used up.
- **Size of the license:** The number of the users who can use this license key at the same time. In case of "Named License" that is the number of the users who can be assigned to this license. In the case of "Concurrent License" it its the number of the users without "Named License" which can log into the system at the same time in.
- **Expiration of the license:** Licenses can have a limited lifetime. In this case, the date of the license's expiration is displayed here. Otherwise the text "Never" is displayed.
- **Assigned users of a Named-License:** Single users can be assigned to a "Named License" - then in this column the names of the users which belong to a "Named License" appear.

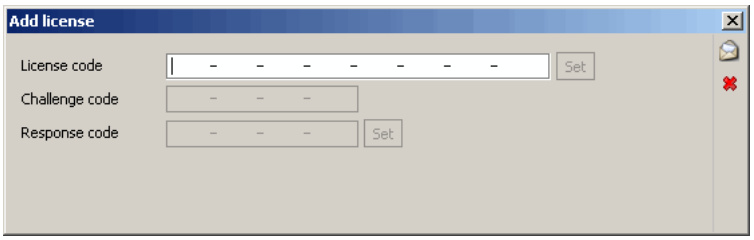
Installation of the first license

With to the first connection with the Workbench to an a newly installed server without no installed license the License-Manager automatically opens and you can install the first license before you work with the server.

Adding a new license

With the "Add license" button from the tool bar in the right margin you can add other licenses to your server.

After using the button another dialog opens in which you can enter the code of the new license and unlock it:



Deleting an installed license

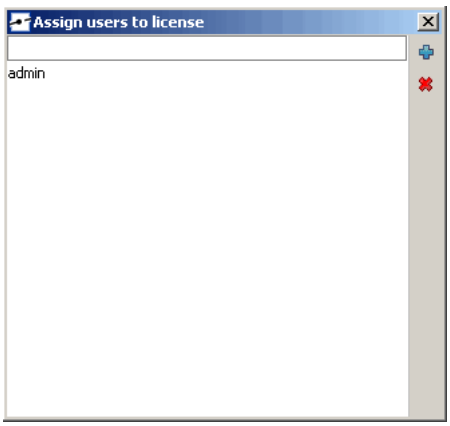
Select the license which you would like to delete in the list and use the button "Delete license". Then, after a query, the license is deleted from the license pool.

If you delete the administrator-license under which you work currently, a new dialog appears immediately which requests you to the input of a new administrator-license. If you enter no new license here, your current session will be killed.

Assigning users to a "Named License"

If you have installed one or more "Named License" on you system, they are not used until you have assigned at least one user to these licenses.

To assign an user to a Named License you must select the license and then use the "Assign Users" button from the tool bar.



Now another dialog, in which all users assigned to the license up to now are displayed in a list, opens. Above the list a small input field, in which you can enter the name of a new user which you would like to assign, is visible.

Adding users

In order to assign another user to a license you must first enter the name of the user in the text field above the list.

The use the button "Assign user" to assign the user and to add him to the lower list. Should the user be already assigned to this license, this operation is ignored.

If an user has been already assigned to another license, the previous assignment gets lost.

Removing users

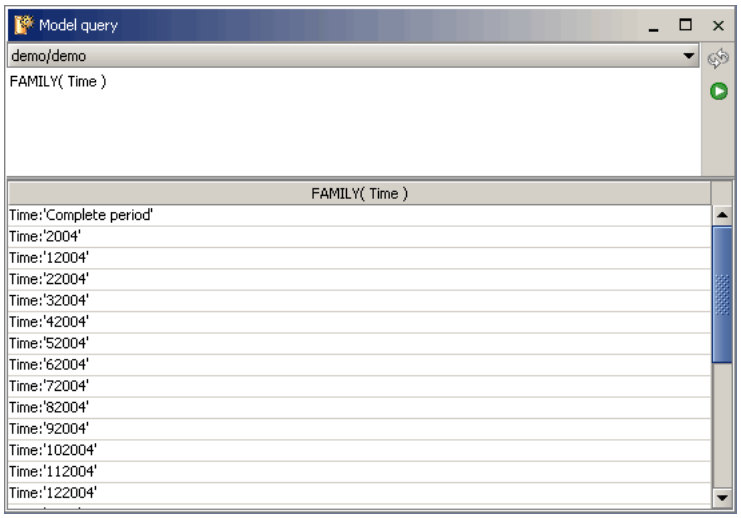
With the button "Delete selected assignments" you can remove the users, which you have selected in the list before, from the license . Only the assignments are deleted by this - not the users themselves.

The Model-Query tool

With the Model-Query tool you can perform adhoc queries in the form of expressions on models. This tool serves primarily for the testing and exploration to know of models. With the Model-Query you can perform *no* graphic report on models.

Opening the Model-Query

Use the button "Model-Query" from the tool bar of the Workbench or the menu item "Tools/Models-Query". Then the window of the tool appears.



In the window different elements are shown:

- The large input field is the place where you can type the expression of the query you want to execute
- The dropdown box above the input field is the model-selector with which you can select the model you want to execute the query on.
- In the toolbar at the right border of the field there are two buttons. With the refresh-button you can refresh the model list and with the execute-button you can execute your query.

In the table below the input field the results of your queries will become visible. This table is empty after opening the tool for the first time.

Selecting a model

First select the model on which you would like to perform a query. A list of all available models is shown in the Drop-Down above the input-field.

Executing to a query

Type your query into the text field "Query". To execute this query on the actually selected model, you must use the button "Execute query" right beside the field. As an alternative, you can also press the keys CTRL+ENTER.

If there is a selection inside the input-field, the model-query will only execute the selected part of the text. Otherwise, the whole expression will be executed.

The result of your query is shown after the successful execution in the table below the input field. In the case of an error, e.g. if the syntax of the expression was not correct, an error message is shown instead of the result.

Refreshing the tool

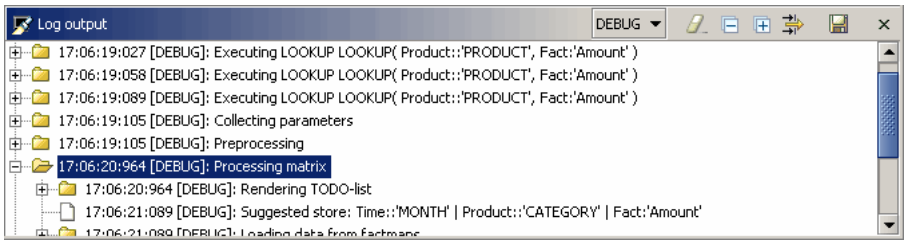
With the button "Refresh" in the upper, right corners you can refresh the list of the models in the dropdown-box. If new models have been added to the system, they will become visible then.

The Logging-Panel

With the Logging tool you can understand and examine the execution of queries and the build of models. The Logging tool is a docking-panel which is shown in the lower edge of the Workbench.

Opening the Logging-Panel

To open the Logging-Panel you can either use the button "Show Log" from the tool bar of the Workbench or the use menu item "Tools/Show Log".



Within the tool all events are shown in a growing tree-list. This list is hierarchical and nested entries are encapsulated into each other and can be folded or unfolded within the tree.

Adjusting the log level

The most important control for the Panel is the adjustment of the log level, with which you can determine how detailed the messages will be. The following log levels are available:

- **NONE:** No messages are generated
- **ERROR:** Only error messages are shown
- **WARN:** Error messages and warnings are shown
- **INFO:** Beside the error messages and warnings all of your activities are reported, e.g the execution of a report
- **DEBUG:** Like INFO, but partial steps of the activities are also listed, each with their duration, the generated SQL statements etc. This level is the best setting for the debugging of reports.
- **TRACE:** In the TRACE level all steps of a report execution are shown, down to the loading of single values from Caches or data bases. This level is extremely detailed and can generate many thousand lines for a complex report.

Use the TRACE level only if you really need such a detailed log. Using this levels can generate a lot of data and take accordingly long. Usually the DEBUG level is sufficient enough for analyzing queries.

Unfolding and folding log entries

With the buttons "Expand all" and " Collapse all" you can open all entries (and make every single entry visible) or the collapse the complete tree again (then you see only uppermost level of the entries).

The log filter

If you activate the log control panel and set a suitable log level, you can only see the events of the operations which were execute by yourselves within the Workbench - e.g. the execution of reports etc.

However, it is also the possible to view the log outputs of all users who work at the moment with the system. This can be used to debug reports which are executed by other users or in the Web-Frontend.

To deactivate the log filter or to reactivate it later, you must use the button "Disable local session filter" from the tool bar of the tool.

Deleting the log

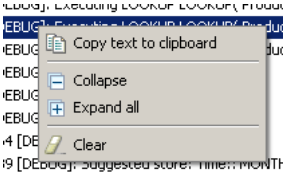
The entries logged up to now can be deleted with the button "Delete log". Deleted entries are lost and cannot be recreated.

Saving the log

If you want to save the logged entries as a text file on your local file system, you can do this with the button "Save log".

The context menu for log entries

Beside the tool bar there is also a context menu for the single entries of the log. You can be opening this menu with the right mouse button.



Beside the functions Collapse, Expand all and Clear described before, the menu offers an other menu item "Copy text to clipboard" with which you can transfer the text of a single entry to the Clipboard of your computer.

This function is useful if you like to copy generated SQL statements from the Log and execute them, e.g. to check the result of a single SQL query.

Closing the panel

To close the panel, you can use the button "Close log" from the tool bar of the tool. Alternatively you also can the use menu item "Tools/Show/Hide log" or the button from the tool bar of the Workbench with which you have opened the tool.

Closing the control panel does not quit the recording of log entries, i.e. a closed log control panel will show all events logged after the closing if you open it again. To switch off the logging you must change the log level on NONE before.

CHAPTER 3:

Using the Query-Editor

Contents of this chapter:

Structure of a query

Before you create your own reports, you should get familiar with the structure of a query, the query elements and their functions. This section gives you a detailed overview of the structure and the elements of queries.

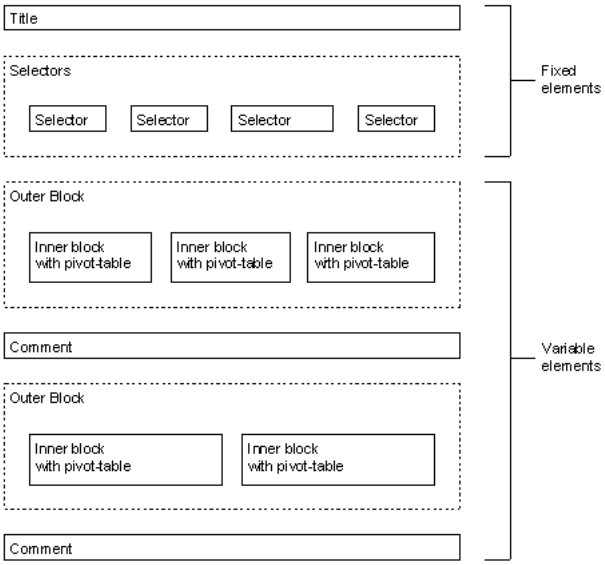
Elements of a query

Within a query there are fixed elements (which exactly occur once) and variable elements, which you can use as much as desired (or leave them out).

The fixed elements are the title and the selector area. These elements appear in each query, even if you did not specify any title or selector. The title and the selector are always placed in the topmost position.

The variable elements are the blocks (there are inner and outer blocks, which are encapsulated into each other) with pivot tables and comments. You can use as much as wanted of this elements and set their order freely.

Structure of a report



Inner blocks are always placed within outer blocks and pivot tables are always placed within inner blocks, so that the pivot tables of a report are organized in a table-like structure. On the highest level of a query, only outer blocks and comments are allowed.

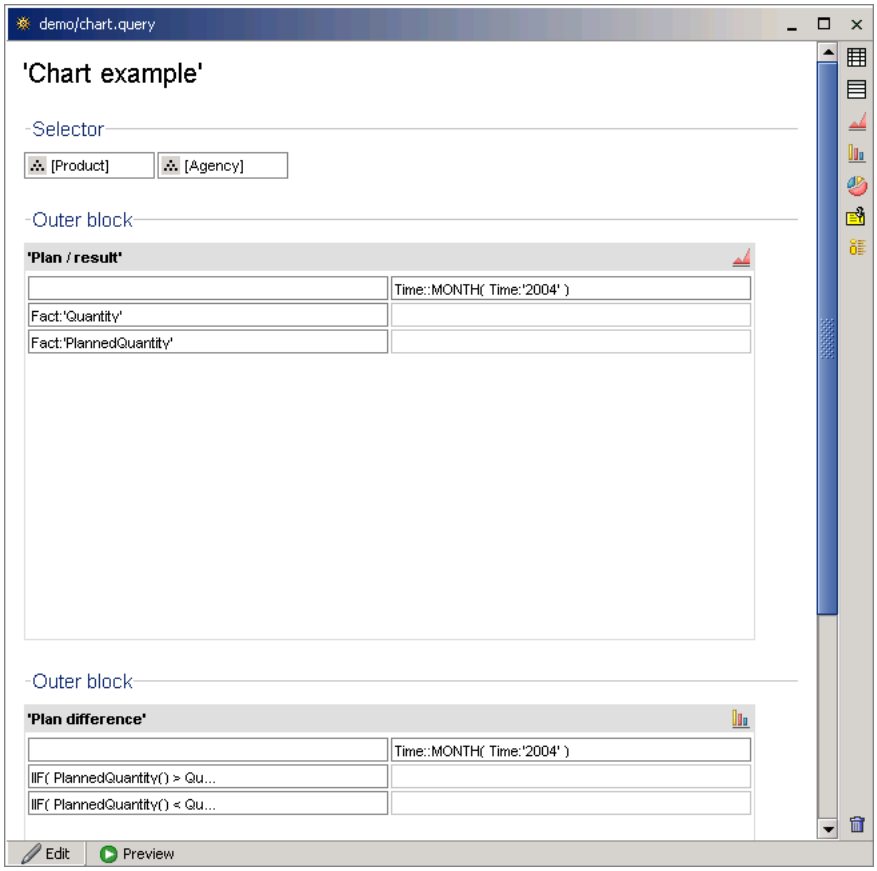
Order of the elements

The order of the outer blocks and the comments can be defined freely. Also, you can define the order of the inner blocks within an outer block at will.

The worksheet

The worksheet is the main work area in the editor, within which you can edit the structure and contents of the query. While editing, it shows a schematic view where all elements of the query are visible and editable but contain no real data.

The Query-Editor worksheet



The worksheet is divided into several, vertically arranged, areas:

- The title in topmost position,
- the selector area, which is always positioned directly under the title and
- the further sections with the outer blocks, inner blocks, pivot tables and comments. A horizontal line and the text "vertical block" is shown above each outer block.

Selectors

Selectors are interactive elements giving the user the possibility to limit the query to certain elements of a dimension (keys).

There are different kinds of selectors, with which the user (depending on the type of the selector) can select one or more keys in different ways. E.g. there are simple lists, search-fields, range-selectors etc.

The selector-block

All selectors of a query are positioned inside the "Selector-area" directly under the title of the query and shown as small boxes. Selectors can be inserted, edited or deleted here. On the small symbol at the left side of a selector you can recognize its type, the text inside the selector shows its options.

Selector-types

There are 9 different kinds of selectors, each with different possible selection sizes and different ways of displaying their options:

	Type	Description
	Single	Allows the selection of a single element with a drop down box.
	Multiple	Allows the selection of no, one or several elements in a list.
	Checkbox	Allows the the selection of no, one or several elements. All elements are visible at the same time and displayed as check boxes.
	Radio	Allows the selection of a single element. All elements are visible at the same time and displayed as radio buttons.
	Interval	Allows the selection of an interval with two drop down boxes, in which the user can select first and last element of the interval.
	Hierarchy	Allows the selection of a single element with a hierarchical search. The selection starts with a single drop down box, in which the user can select an element from the highest dimension-level. Then the children of the element will be displayed in a second drop down box and so on until the desired element is selected.
	Hierarchy-Applet	An interactive Java Applet. This Applet allows to navigate in the dimensions hierarchy without executing the query meanwhile.
	Button	Allows the selection of a single element with a button. The selectable element is defined by the creator of the report.
	Input	Allows the search of elements with a search pattern. The search-box accepts the wildcards "*" and "?".
	Calendar	Allows the selection of years, quarters, months, weeks or days from a classic calendar, which is displayed as an Java Applet in the query.

f0	Key-Expression	A Key-Expression selector allows advanced users to create a expression with an interactive Java-Applet, which can be used as a text-variable inside the query.
AB	Constant	Displays and selects a certain element and is not changeable by the user.

Selector options

Each selector offers a defined set of options, from which the user can select one or more elements and filters the query with. A selector does not always offer all keys of a dimension or dimension-level, only the keys defined by the options are selectable.

A Multiple-selector offering all elements of a dimensions-level

The same selector with only the three first options

A special case is the input-selector, with which the user can search keys using a search-pattern: Since this selector initially offers no keys but only an empty search-field, the options here define the restriction of the search. This means the result of a search is always limited to the keys which also occur in the options.

Alternatively you can define a boolean expression as a match-filter for a selector, which delimits the options when they are created or whenever a value is selected. Especially for search- and hierarchy-selectors this is the better way to define the options.

Default selections

Each selector can define a default selection, which is pre-selected if the user executes a query for the first time. If no default selection was defined for a selector, then the selector is empty after the first execution of the query and the selection-result of the selector is empty and that the report may show no data.

A selector without default

Selector with default

A selector of the type "single" without a standard selection contains an additional empty option as its first element, which stands for the missing selection and shows the user that this selector must be selected. After choosing an option, this empty option is omitted and disappears.

Dependence of selectors on parameters and among themselves

Usually the options and the selections of selectors do not affect each other - e.g. you can create two selectors for the same dimension and select different values with them. Also, the options and the standard selection are normally not affected by the parameters of a report.

In some cases it can be wanted that the options and the standard selection are dependant on other selectors or on parameters. E.g. imagine two selectors with which the user can select a "customer" and a "product". You could configure the second selector in that way that it only offers the products which are bought by the selected customer.

The interdependence on selectors must be activated explicitly for each selector.

Design options for selectors

Selectors offers some design options, with which you can affect their appearance:

- The **title** of a selector can be defined (otherwise the name of the offered dimension is used as title),
- the **texts** of the options can be defined (otherwise the IDs of the offered dimension-elements are used as option-texts),
- the **width** of the selector can be set (otherwise it's determined automatically on the basis of the longest option)
- the **breakafter** option of the selector will put all following selectors in a new row inside the selector-area.

Automatic refresh

Single selectors and others allows the automatic re-execution of the report directly after changing their selection. Dependant on the report you can switch this feature off, in order to avoid the execution of the query between the manipulation of several selectors.

Providing date-formats for the calendar

For the Calendar-Selector you must define which elements of your Time-Dimensions should be selectable and how their format is. In practise this means you have to provide the date-format of all Elements which should become selectable. There are 5 different formats you can specify. All of this formats are available as properties in the Property-Editor when selecting the selector:

Property	Description
dayFormat	Format for days, e.g. "dd.MM.yyyy"
monthFormat	Format for months, e.g. "MMM/yyyy"
quarterFormat	Format for quarters, e.g. "q/yyyy"

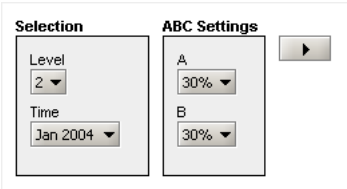
weekFormat	Format for weeks, e.g. "ww yyyy"
yearFormat	Format for years, e.g. "yyyy"

The formats you provide must match the elements of your dimension, otherwise the selector won't work. If you leave out formats, the selector will not be able to select this elements.

Selector groups

Multiple selectors can be grouped together to a single selector group. Usually, selectors are placed on a horizontal row, but inside a group they are arranged vertically and therefore use less space in the report.

Selector-Groups



Like selectors, the groups allow a number of different styles which can be changed to adopt the appearance of the group. E.g. the background color, the border or the size of a group can be altered.

Outer blocks

Outer blocks (the "vertical blocks") organize the contents of a query in a vertical order. Except comments, only outer blocks may be placed directly on the worksheet of a query. Within an outer block, only inner blocks may be placed.

All elements inside an outer block (which are always inner blocks) are displayed next to each other - a vertical block arranges its contents in a row.

Adjustment (orientation) of the inner blocks

Usually the inner blocks lying in an outer block are positioned in a row next to each other. However, this behaviour can be changed by the report-designer. The following adjustments are possible:

- **In a row (horizontally):** All inner blocks are displayed next to each other.
- **In a column (vertically):** All inner blocks are displayed one below the other.
- **With riders (tabbed):** The inner blocks are displayed behind each other and only one block is visible. The users can choose the block he would like to see with a tabulator.

An inner block, iterated by month and layed out as tabs

Jan Feb Mrz Apr Mai Jun Jul Aug Sep Okt Nov Dez

	Desktop		Notebook		Pocket PC		Tablet PC	
	Quantity	Amount	Quantity	Amount	Quantity	Amount	Quantity	Amount
Acer			2.189	5.675.775			2.059	5.156.689
Apple	4.307	11.351.102	2.044	7.706.419				
Casio					2.084	944.752		
Compaq	1.976	1.894.065	1.971	3.226.680	2.162	1.134.996	2.179	6.091.981
HP	2.016	2.328.776	2.151	5.303.952	2.026	1.504.937		
IBM			2.212	7.296.610				
Sony	2.046	5.184.116	4.025	10.480.274	1.980	1.395.771		
Toshiba			1.919	5.094.254			2.067	6.550.285

Visibility

Outer blocks can be made invisible by a condition. Invisible block will not only be invisible in the result, they also consume no computationtime when the query is processed on the server.

Iteration

Outer blocks offer the possibility to repeat their contents for a set of dimension-keys (e.g. for products or months). This repetition of blocks or other elements is called "iteration".

Each block generated by an iteration is an exact copy of the original block but uses another filter (e.g. if a block repeats for a set of products, then each copy uses a filter which limits the contents of the block to a specific product).

Iteration of an outer block over months

Jan 2004

	Quantity	Planned	Amount
Compaq Presario 6400NX Desktop PC	2.233	2.254	2.082.382

Feb 2004

	Quantity	Planned	Amount
Compaq Presario 6400NX Desktop PC	2.023	2.063	2.013.312

Mrz 2004

	Quantity	Planned	Amount
Compaq Presario 6400NX Desktop PC	2.104	2.153	2.019.925

Apr 2004

	Quantity	Planned	Amount
Compaq Presario 6400NX Desktop PC	1.976	1.948	1.894.065

If an outer block contains multiple inner blocks, then all of them are copied by the iteration. E.g. a block with three inner blocks and an iteration over 10 elements produces 30 blocks in the result.

Filter

Outer blocks can additionally (like all other block elements, too) influence their filter and the filter of the elements contained in them, which affects all their expressions and properties using the filtered dimensions. The iteration mentioned above also influences the filter.

You find a detailed description of filters in the section Filters and subcubes.

Layout options

Outer blocks do not offer further layout options, since they are not visible in the report. They function as a pure grouping of inner blocks.

Inner blocks

Inner blocks may only by placed inside outer blocks. Only pivot-tables can be placed inside an inner block. The pivot-tables can be displayed in different formats, e.g. as a table, chart or newsticker, which is defined by the containing inner block.

Formatting the contents

The inner block containing a pivot table also defines the displayformat of its table. There is a set of different formats available for formatting the contents of a table. Tables may be formatted:

- As a simple **table**,
- as a **chart** (line-, bar- or pie-chart),
- as a **newsticker**

Also chart are simple pivot tables, which however are displayed in another format. In order to generate charts, you must know how the pivot tables are converted to them.

If a table is being displayed as a chart or newsticker, the system will convert the table into the target format using the following conversion-scheme:

Conversion into charts

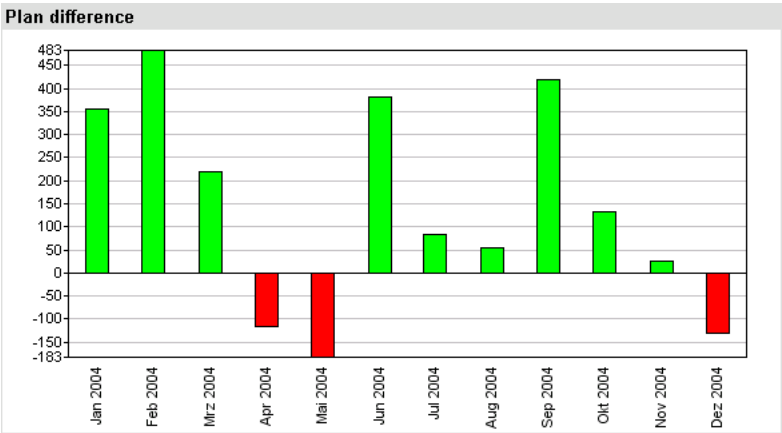
For charts, each column of the table becomes one data point, i.e. a point in a line-chart, a bar in a bar-chart or a slice in a pie-chart. Each row of the table becomes its own data series, i.e. a line in a line-chart, a bar series or a pie in a pie-chart.

Input data for a barchart

Plan difference

Jan 2004	Feb 2004	Mrz 2004	Apr 2004	Mai 2004	Jun 2004	Jul 2004	Aug 2004	Sep 2004	Okt 2004	Nov 2004	Dez 200
355	483	221	-117	-183	383	85	55	419	132	27	-13

Same table displayed as barchart



Links in pivot tables are automatically converted into links within the charts, so that, by clicking onto the elements of the chart, the user can jump to other reports or drilldown within a chart.

Additionally, inner blocks offer a large number of further design properties, which can be used to lay out charts and to change their appearance. With these properties

almost all characteristics of a chart, e.g. the colors, the scaling, the texts etc. can be changed.

Note that the size of the inner block also determines the size of the chart. In order to display a chart, the size for the block must always be set, otherwise the chart remains invisible.

Conversion into newstickers

In a newsticker, all rows (without the headers) will be converted into a single line, starting with the first row and from the leftmost to the last column.

The input-data for a newsticker

	Jan 2004	Fe
Desktop	Desktop / Jan 2004: 20.708.942 USD	Desktop / Feb
Notebook	Notebook / Jan 2004: 45.343.858 USD	Notebook / Feb
Pocket PC	Pocket PC / Jan 2004: 4.944.120 USD	Pocket PC / Feb
Tablet PC	Tablet PC / Jan 2004: 16.179.583 USD	Tablet PC / Feb

The same table displayed as newsticker

Jan 2004: 20.708.942 USD Desktop / Feb 2004: 21.155.679 USD Desktop

Visibility

Inner blocks, as outer blocks, can be made invisible by a condition. Invisible block will not only be invisible in the result, they also consume no computation-time when the query is processed on the server.

Iteration

Inner blocks, like outer blocks, offer the possibility to repeat their contents, e.g. for a set of products or months. This repetition of blocks or other elements is called "iteration".

Each block produced by an iteration is an exact copy of the original block but uses another filter (e.g. if a block is being repeated for a set of products, then each copy uses a filter with another product and limits the product-dimension to this product).

An inner block, iterated by month and layed out as tabs

Jan 2004

	Quantity	Planned	Amount
Compaq Presario 6400NX Desktop PC	2.233	2.254	2.082.382

Feb 2004

	Quantity	Planned	Amount
Compaq Presario 6400NX Desktop PC	2.023	2.063	2.013.312

Mrz 2004

	Quantity	Planned	Amount
Compaq Presario 6400NX Desktop PC	2.104	2.153	2.019.925

Apr 2004

	Quantity	Planned	Amount
Compaq Presario 6400NX Desktop PC	1.976	1.948	1.894.065

All inner blocks generated by an iteration are lined up next to each other in a row, since they all belong to the same outer block (unless you change the the orientation of the outer block).

Filter and subcubes

Internal blocks can additionally, like all other block elements, influence their filter and the filter of all elements contained in them. All expressions and properties of the

contained elements will use the changed filter. The iteration mentioned before has also influence on the filter.

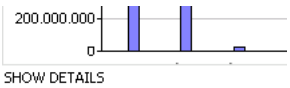
Filters only affect the current selection of dimensions (i.e. the current product), but still permit to access all other, unfiltered elements of the dimensions. You can also define subcubes for inner blocks, which completely limit the access to dimensions to the elements contained in the subcube.

You find a detailed description of filters and subcubes in the section "Filters and subcubes".

Linking internal blocks

Internal blocks can be provided with an additional link, which is displayed directly underneath the block as a right-justified text. Using this link you can offer e.g. further detail data for a block.

Links are displayed under the block



The link can be, like all links in HTML, displayed as a pure text or with a symbol and refer to other queries or any other URL.

Laying out inner blocks

Contrary to the outer blocks, inner blocks are visible, because they display a the frame around the pivot table or around the chart. For an inner block, you can set different design properties which affect their appearance:

- The optional **title**, which is displayed above the contents of the block,
- the **colors** of its background, frame and title,
- the **font** of the title,
- the **size** (width and height) of the block,
- the **inner padding** (the distance between the frame and the contents) of the block

A block with changed layout

Jan

	Desktop		Notebook		Pocke
	Quantity	Amount	Quantity	Amount	Quantity
Acer			2.015	5.530.856	
Apple	3.976	10.720.472	2.188	7.603.237	
Casio					2.035
Compaq	2.233	2.082.382	1.976	3.535.250	2.019
HP	1.927	2.157.849	2.078	5.094.064	2.058
IBM			2.183	7.309.708	
Sony	2.216	5.748.239	4.135	10.548.763	2.022

The title of a block is the text being displayed directly above the pivot table. With block titles it is possible to display additional sub-titles for the individual sections of a query additionally to the main-title. Block titles are optional.

With the color-settings you can determine, how the frame around the block, the background of the block and the title should look like. If the block displays a title, then the background color of the title is equal to the frame color.

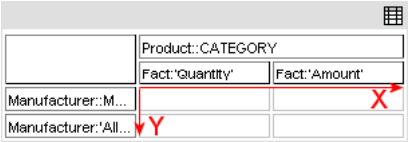
Additionally, the used font and the font-size of the title can be changed.

Pivot-tables

The actual data or a report is displayed inside "pivot tables". A query can contain more than one pivot table, but each table must be placed within its own inner block (means that you can only display one table per block).

Pivot tables are tables which determine their structure and their later contents over the definition of their row- and column-headers. Since tables own two axes (one x and one y axis), you can define such headers for both axes.

The axes of a pivot table



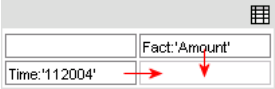
A header usually determines, which key (e.g. a product or a fact) is to be displayed. Additionally, a header can define a large quantity of properties for a whole row or column of the table, e.g. texts, formulas or colors.

Determination of the table contents

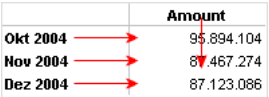
The combination of the headers determines, among other things, the contents of the table in the result and how each cell of the table is is affected (e.g the cell in the lower example is influenced the header "Fact:Amount" in the x axis and by the header "Time:'112004" in the y axis).

If a headers refers to a fact, than this fact will become automatically the fact being displayed inside the cells. All other headers determine the filter for the fact. In the lower example the turnaround for the month "112004" will be displayed, since this is the second header which has influence on the cell.

Influence of the headers on the cells (design view)



Influence of different headers on their cells



	Amount
Okt 2004	95.894.104
Nov 2004	87.467.274
Dez 2004	87.123.086

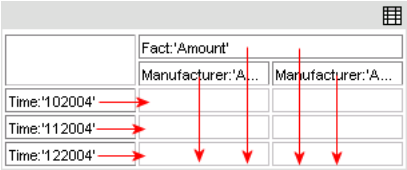
Whether facts are placed the x axis (like in the upper example) or in the y axis does not make any difference.

Encapsulated headers

The upper example used only one dimension (time) and one fact (Amount) and is very simply. It is also possible, by nesting headers, to display more than one dimension inside a pivot-table.

Nesting headers means that other headers can be placed below a header. In the lower example, two keys of the dimension "Manufacturer" were nested under the fact "Amount".

Nesting headers



When this query is executed, a table with two columns will be generated (as you can see in the upper picture, is the header containing the fact became a spanning header for the two other headers).

The cells of this query are affected, like in the first example, by their headers: The cells of the first column are affected by the header with the fact and the header with the first manufacturer and the column therefore shows the amount for the respective week and this manufacturer. The second column is also affected by the header containing the fact but by the other manufacturer. Therefore it shows the amount for the second manufacturer.

Influence of the nested headers

	Amount	
	Acer	Apple
Okt 2004	10.658.522	23.977.652
Nov 2004	10.732.414	18.273.092
Dez 2004	10.704.940	18.931.004

Nesting headers can be combined at will (at both axes) and you can theoretically display a large number of dimensions within a single pivot table.

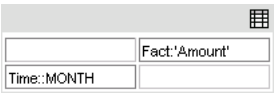
Iteration of headers

In the upper examples each header represented exactly one key or fact. In case of a large dimension this can be very complex and uncomfortable because you had to drag single each key into the table. Also, the header would not be influenced by the selection of the user, which is usually wanted (in the upper examples this wouldn't happen, because the headers always shows certain keys).

If several keys of a dimension are represented with individual headers, e.g. all products of a product dimension, this also has the disadvantage that new products wouldn't appear in the query before they are manually manually added to the table.

Therefore headers usually work with iterations (repetitions of keys), which are determined by formulas. Such a formula could e.g. have "all products", "the last 2 years" or other sets of keys as a result.

A table using an iteration on its y axis



In the upper example, the y-headers were replaced by a single header showing all filtered weeks (more exact: All filtered elements of the level "Zeit_L2" of the dimension "time"). The result the table therefore shows several rows (one per week).

The result shows all months on the y axis

	Amount
Jan 2004	87.176.503
Feb 2004	124.015.669
Mrz 2004	88.918.576
Apr 2004	88.321.434
Mai 2004	105.090.346

In practice most headers use an iteration, even if this has only one key as result. In the first examples the simplest form of an iteration "<Dimension>:<Key>" was used which returns a single certain key of a dimension (e.g. "Time:'112004'" or "Fact:Turnaround").

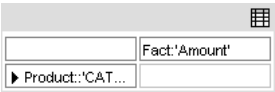
If a header has no iteration, it is not repeated and results to exactly one row or column. Also, headers without iteration do not affect the filters of their cells.

Drilldown

Pivot tables offer the possibility of navigating within the headers with "drilldown". Drilldown means to "unfold" a header for the presentation of its detail data, e.g. for showing all months weeks of a year which was originally displayed in a table.

Drilldown must be activated for each individual header - headers do not offer the possibility to drill down by default. If drilldown is activated, the user will see a drilldown-symbol (a small triangle) and can unfold to the detail data by clicking onto this symbol or fold it by clicking on the symbol again.

Activated drilldown for a header



Drilldown is also possible for all headers produced by a previous drilldown, i.e. if a user drilled down from one year to its quarters, then he can thereafter drill down from a quarter to its months and so on.

Usually, the drilldown shows the hierarchy of a dimension and all child elements of a key are visible after a drilldown (e.g. weeks for years or product for product groups). This standard behavior can be changed by the designer of a query, e.g. if he wants to display elements of another dimension when the user unfolds a header (e.g. all customers to a product).

Layout options for drilldown

There are two different layout forms for drilldown: The vertical layout and the encapsulated layout. In the vertical form all new headers are displayed below the original header after the drilldown (or next to it in the x-axis). In the encapsulated variant the new headers are displayed nested to the original header, so that the hierarchy becomes more visible and the tables uses less rows.

Encapsulated drilldown

		Amount
▸ Desktop		524.724.708
▾ Notebook	Acer TravelMate 233XV Notebook	136.610.138
	Apple iBook Notebook 141 M9009LL/A	187.316.766
	Compaq Presario 1215 Notebook	83.587.564
	HP Pavilion xt5335qv Notebook PC	126.021.524
	IBM ThinkPad R40 (27232FU) Notebook	171.743.184
	Sony VAIO PCG-FRV25 Notebook	102.129.680
	Sony VAIO PCG-GRT170 Notebook	163.092.202
	Toshiba Satellite 1135-S125 Notebook	127.860.266
▸ Pocket PC		122.257.350
▸ Tablet PC		495.949.726

Formulas

Till now we only created pivot-tables, in which the used fact was determined by the iteration of the headers. Instead of referring a fact in such a way you can also define a formula for a header. This is useful e.g. if you don't want to create a header for a fact

(for layout reasons) or if the needed fact or computation is not available in the query-model and you need an adhoc-computation.

Usage of a formula in a header

		Umsatz
▼ 1998	KW 40/1998	37.752.899,86
	KW 41/1998	37.531.905,69
	KW 42/1998	37.940.533,34
	KW 43/1998	41.312.171,86
	KW 44/1998	[illegible]

In the upper example the formula "Amount()/Amount(YEAR)" is used to compute the proportional portion of the annual turnover. There is a large set of functions and possibilities available for the computation of values in formulas. You find a detailed description of the expression language on the "instantOLAP Reference".

Result of the formula

Jan 2004	0,078
Feb 2004	0,111
Mrz 2004	0,079
Apr 2004	0,079
Mai 2004	0,094
Jun 2004	0,078
Jul 2004	0,083

If you define a formula in a header, this formula will be used instead of any fact influencing a cell.

Texts and formatting

Tables offer different possibilities for the layout and format of their texts inside headers and cells.

In headers, the display-text can be set freely and has not to correspond to the key used in the header. You can also use constant texts, attribute of keys (e.g. the short-name of a product) or anything else. The text of a header can be defined with formulas, with the same possibilities like formulas.

Apart from the formulas you can also define the display-formatting for the cells, which formats the result of a formula (or fact, if you work without formulas). By formatting cells you can determine, how a number or a date is to be represented, e.g. with or without leading zeros or with a thousand-delimiter.

The upper example with title and cell-format

	% of year
Jan 2004	08 %
Feb 2004	11 %
Mrz 2004	08 %
Apr 2004	08 %
Mai 2004	09 %
Jun 2004	08 %

Laying out pivot tables

Pivot tables offer a large variety of further layout options, both for the table itself and for the headers and cells inside the table.

A table offers general properties, i.e for the design of its upper, left corner. The headers offer properties for the layout of the headers and the cells generated by them. There are the following groups of properties:

- **Colors** of the texts, backgrounds and borders (for the corner, the headers and for the cells produced by them),
- **fonts** and **font-sizes** for the corner, headers and cells,
- **text alignment** for the corner, headers and cells,
- **sizes** (the width and height of headers),
- **rotation** of texts (vertical display of headers)
- **padding** (the inner space between the text and borders) for headers and cells

All properties of tables and headers have to be defined with formulas and everything can be computed at the time of the query execution. Therefore you can make anything dependant on certain conditions.

A table with different background, border and font-size

	% of year
Jan 2004	08 %
Feb 2004	11 %
Mrz 2004	08 %
Apr 2004	08 %
Mai 2004	09 %
Jun 2004	08 %

Comments

Comments are used to add notes to queries.

A comment below a block

Okt 2004	09 %
Nov 2004	08 %
Dez 2004	08 %

Extracted from the central data-warehouse

Like vertical blocks, comments may only be placed on the highest level of a worksheet and not within blocks or pivot tables.

Computed comments

A comment can be a static text or be computed at the time of the query execution, e.g. in order to display a comment only on a certain condition.

The computation of comments is based on formulas and you can use all expression capabilities of instantOLAP to compute comments.

Author and date of a comment

Comments can contain optional information like the author and at the time of its creation. If any of you these information is given, it will be displayed together with the comment.

Export of comments

For comments you can define if they are exported when storing snapshots (stored report results) or when results are converted to PDF or Excel documents.

If the export of a comment (e.g. a usage hint) makes no sense and may therefore be unwanted, the export for these comment should be deactivated.

Using HTML and JavaScript in comments

Comments can also be used to display HTML or execute JavaScript code in the HTML-output. By beginning the comment text with "javascript:", the comment will automatically put into a JavaScript block in the HTML-output instead of being displayed.

Filters and subcubes

Filters and subcubes are not elements of a query but very important concepts, which affect all elements of the queries and are very important for their understanding. Particularly filters have a large influence on the query-design and you should be familiar with their concept.

Filters

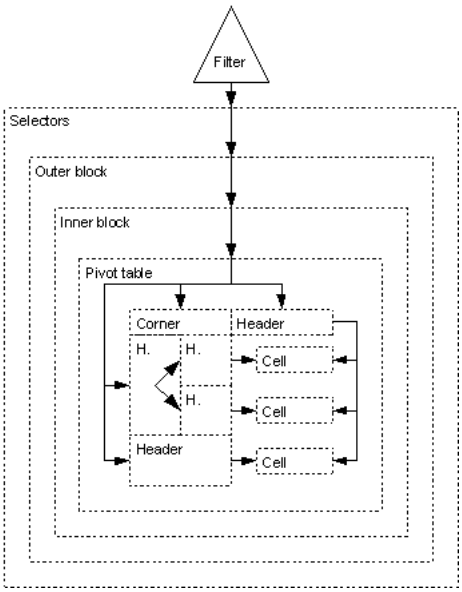
A filter can be interpreted as the "current selection" of a dimension, i.e. it determines which elements of a dimension a query, a block, a table or a header currently displays. A filter may contain as many elements of a dimension as desired.

Passing a filter within a query

Filters are passed inside a report from the top to the bottom - starting from the parameters of a query. It will be passed to the selectors, they will pass it (after they changed it) to the outer blocks, then to the inner blocks and so on. Each element of a report can manipulate or limit the filter, particularly selectors and headers in pivot tables have a large influence on the filter.

The start-filter of a reports is either a set of all root elements of all dimensions (like "All products", "All customers" etc.) or influenced by the query-parameters (if the reports was called from another query).

Passing filters inside a query



The filter of each element can be limited with the property filter of the element. Instead of using this property, the usage of the property iteration (which is used to repeat elements) is more common, particularly for headers. E.g. a header could be repeated for each week of a year, in order to produce multiple rows or columns. This iteration

has a large influence on the filter, because the filter is limited for each element generated by the iteration and the manipulated filter is passed to the sub-elements of the iterated element.

A special case are the cells, which are the end of the filter chain and are always affected by two headers: By the header in the x axis and by the header in the y axis belonging to the cell. Therefore a cell represents exactly the the part of the data model which is specified by both headers (e.g. if a header with the iteration "Fact:Turnaround" lies on the a x-axis and another header with the iteration "Product:A" on the y-axis, the the cell shows exactly the turnaround for this product "A").

Influence of the filter on properties and formulas

All functions in the properties refer to the current filter the element is generated with. Particularly, if you work with relative functions inside the expressions (e.g with dimension-names or dimension-levelnames), these always refer to the filter.

E.g. a header with an iteration "week" (here "week" should be a level name of the dimension "time") would return all weeks of the current time (independent if that would be one year or one day). The current time is exactly the element (or the elements) from the current filter belonging to the dimension "time".

If you use only a dimension name inside a formula, then this expression returns the elements of this dimension stored in the filter.

Note that you can access, independent of the current filter, all other elements of a dimension in formulas (e.g. if the current filter contains a "Product:A" you still can use a "Product:B" in iterations). Therefore a filter is no access limitation.

Function of the filter property

Almost any element owns a property filter, with which you can limit the filter of its sub-elements. These property accepts expressions which return dimension elements.

These dimension-elements are applied to the filter and all dimensions with at least on element in the result occur are limited to exactly these elements - all other dimension remain uninfluenced by the filter property!

Subcubes

While filters define only the current selection of the dimensions for elements and their properties, you can define real access restrictions for the dimensions with the subcube property. "Real access restrictions" means that no header inside a table has access to the dimension-elements not being part of the subcube. Therefore tables would act as if the dimension would not contain these elements at all.

Subcubes are mostly used to limit dimensions to the interesting part (usually the elements for with present data). E.g. if you want to display pivot-table where only a small part would be filled with data, a subcube is often a good way to limit the dimensions to the part with data.

rows in the chapter Pivot-tables. The functions LOOKUP and DLOOKUP (to find in the "instantOLAP Reference") also also interesting for suppressing blank rows and

columns. They are used for searching dimension-elements which provide data for a certain fact and often used in subcubes.

To define a subcube you can use the property "subcube" of a pivot table. This property expects a formula which returns a set of dimension elements. All returned elements form then the new subcube (whereby more than one element for each dimension can be returned) - and all elements, which are not part of the result, do not occur in the subcube (also the facts!). You find a detailed description of the subcube property in the "instantOLAP Reference".

Starting the editor

Reports can be created and edited in two different ways: With the Web-Frontend or with the instantOLAP-Workbench.

Within the Web-Frontend the editor is available as a Java Applet and you can open and use the editor within the Browser (provided that you to have Java 1.4 or higher installed on your computer). In order to be able to use the editor, you must be an administrator or Power-User.

The Workbench (the administration tool of instantOLAP) is a standalone desktop application, which is used for remote maintenance and administration of the server. The Workbench must be installed on your computer before its use. In order to be able to use the Workbench, you must use a user-account with administration rights.

Starting the editor in the Web-Frontend

System-Requirements

To use the editor within the Web-Frontend, you must enable the execution of Java Applets in your browser. These are the Java- and memory requirements for your browser:

- Java 1.4 (or higher)
- 64 MB of free memory (or more)

If no Java is installed on your computer, you can download and install the most current Java version from the Sun Java website (http://www.java.com). Also make sure that your browser allows the execution of Java Applets, otherwise you won't be able to edit reports inside the Web-Frontend.

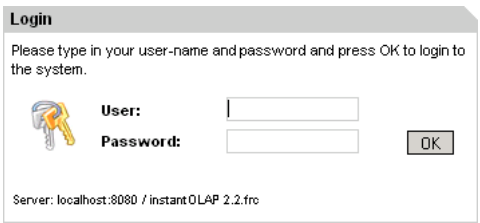
Logging into the Web-Frontend

Before you can create new reports within the Web-Frontend or work on existing reports, you must be logged in as a Power-User or administrator.

The login button

If you are not logged in with a Power-User or administrator-account, use the "Login" button from the toolbar to change the account you are currently working with. Then the following login-dialog appears, in which you must type in your user name and your password.

The login dialog



Confirm your input with the "OK" button. Then will be forwarded to the Web-Frontend using the new account. Read the section "Logging in" in for a detailed description of the login procedure.

Creating new queries

In order to create a new query, you must open the folder in which you would like create the new query. If you own the sufficient rights in the system and for this folder, you will see a button named "New", with which you can create a new query. You also can create a new query while watching another one, then the it will be created in the same folder as one you are currently watching.

The new query button

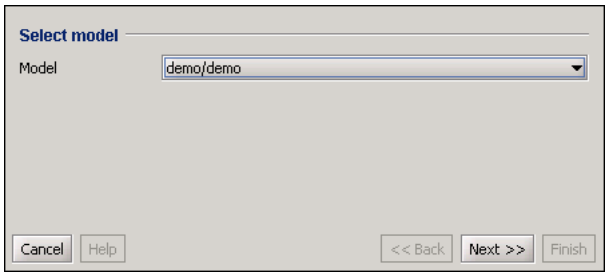
Use the button "New" to create new queries.

After pressing the button, the Editor-Applet will be downloaded from the server onto your computer. This can take, depending on the quality of the connection to the server, a few seconds or up to one minute (the Applet will be only loaded once - when creating or editing a second report the Applet will be already available).

After the Applet was successfully loaded onto your computer, it starts connecting to the server. The current activities of the editor are always visible in the status line of your browser - if you don't see any status line you should activate it (e.g. with the menu item "View/Status Line" in the Internet Explorer).

After the Applet successfully connected the server, a new dialog will open in the browser, in which you must make some base settings for your new query:

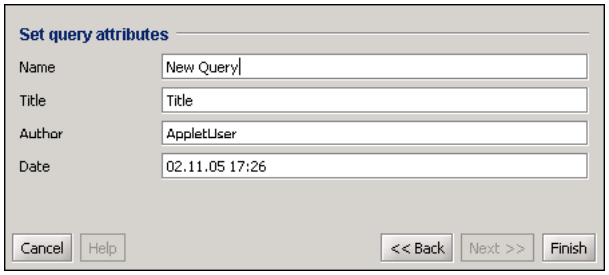
Selection of the model



1. Selection of the model: At first you have to choose the data model on which the new query will be based. The drop down box "Model" will offer all available models you have the sufficient read access for.

Select the desired model and continue by pressing on the button "Next >>".

Setting the query properties



2. Setting the general query properties: Afterwards you can set the name, the title, the date and the author of the query. All fields already have a default value, which can be changed by you:

- **Name:** The (short) name of the query, which appears in the navigation in the Web-Frontend.
- **Title:** The (detailed) title of the query, which will be displayed as a title above the report at the time of the report execution.
- **Author:** Name of the author, thus your name.
- **Date:** Time of the query creation.

Confirm your input by pressing on the button "Finish". Now the query will be created and the empty worksheet of the editor will appear.

Opening existing queries

In order to work on an existing query, you must open and execute it before. Then (if your account has the appropriate a rights to edit this query) a button named "Edit" will appear in the menu above the query.

The edit button

After pressing this button, the editor opens inside the browser. Like in the case of creating a new query, the browser will download the program code of the Editor-Applet first. After the download has finished, the editor will become visible, showing the existing report and its settings.

Starting the editor in the Workbench

System-Requirements

In order to create or edit query within the Workbench, it must be installed on your computer and you need an account with the appropriate access-rights of (administration rights) for the instantOLAP server. Read the Workbench documentation for further details.

Creation of new queries

There are three different ways to create a new query within the Workbench: Using the menu, using the toolbar of the Workbench or using the context menu of a folder within the Repository-Explorer.

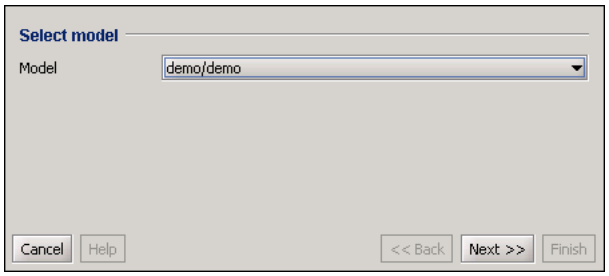
Creating queries using the menu or the toolbar

Use the menu-item "Query/New query" in order to create a new query. Then a dialog opens in which you have to specify the model, the folder and the filename and some general properties of the new report.

The new-query button

Alternatively to the menu you can also use the button "New query" from the toolbar of the Workbench.

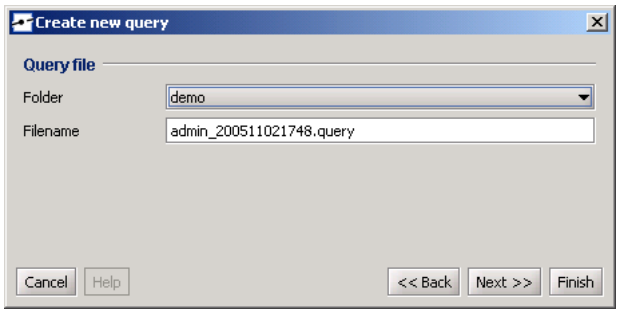
Setting the model



1. Selection of the model: At first you have to determine the model on which your new query will be based. In the drop down field "Model" only the models are shown for which you own sufficient access-rights.

After you have chosen the model press the button "NEXT >>" to proceed with the wizard and to continue with the next page (selection of the folder and filename).

Selection the queryfolder and -name

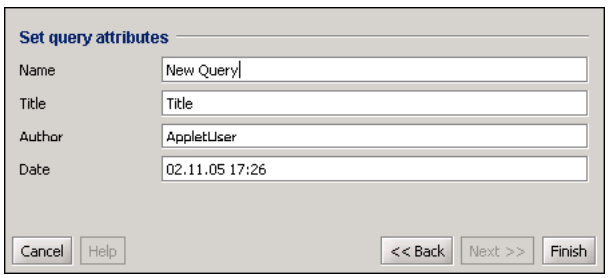


2. Selection of the folder and filename: Contrary to Web-Frontend can you can define a filename for your query and define the folder where it will be stored. An additional page will appear in the wizard after the selection of the model.

Select the folder with "Folder" drop down box and change the filename in the field "Filename" if wanted.

The filename must always end with the extension ".query", otherwise the wizard will not accept the filename and show an error message. You should always give clear names to your queries. Again, confirm the wizard page with the "Next >>" button after you changed the settings (you also may use the "Finish" button to create the report immediately and skip the following page).

Setting the query-attributes



3. General query properties: At last a wizard-page is shown containing the general report properties. You may change the properties at will or keep the default values and change them later in the editor. The following properties can be changed:

- **Name:** The (short) name of the query, which is shown in the navigation of the Web-Frontend.
- **Title:** The (detailed) title of the query, which will be displayed as a title above the report at the time of the report execution.
- **Author:** Name of the author, thus your name.
- **Date:** Time of the query creation.

Confirm your input using the "Finish" button. Now the query will be created and the empty worksheet of the editor will appear.

Creating query using the Repository-Explorer

Instead of using the toolbar you can also use the context menu of the folder in the Repository-Explorer (on the left hand side) in which you would like to create the new query. Use the menu-item "New query". Then the same wizard as above (with a pre-defined folder) appears.

Opening existing queries

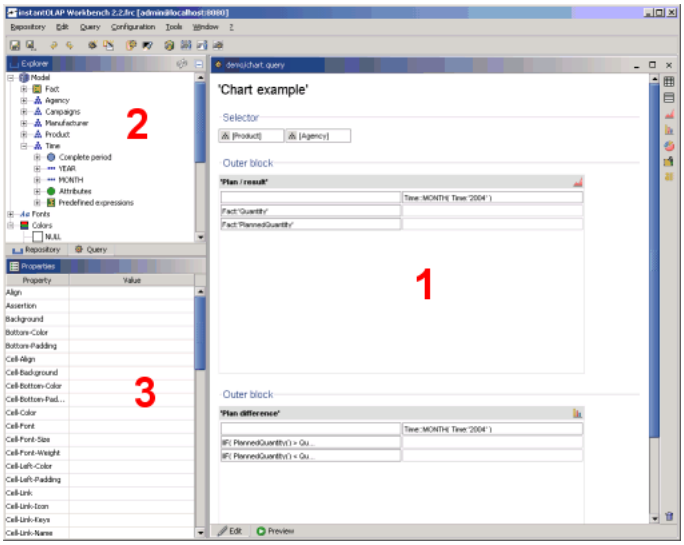
Existing queries can be opened by locating them in the Repository-Explorer (on the left hand side) and using the menu-item "Edit" from their context-menu. Alternatively you can open editor for a query with a double click using the mouse or by dragging the report file onto the desktop-panel of the Workbench.

In order to work on existing queries you must own the appropriate access-rights for it and for the model the query is based on. Otherwise you cannot see the query within the Repository-Explorer or you will receive an error message when trying to edit the report.

Structure of the editor

After creating or editing a new or existing query, the worksheet of the editor appears in the browser or in a separate window within the instantOLAP Workbench.

The editor



The editor is divided into three different areas:

Differences between the Web-Frontend and the instantOLAP Workbench

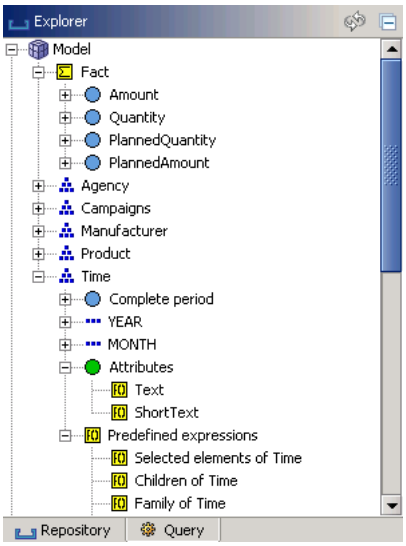
The editor in the Web-Frontend looks very similar to the editor in the Workbench but is slightly different, since you can edit several reports at the same time with the Workbench. Also, in the Workbench the Model-Explorer and the Property-Editor are not part of the editor window itself but fixed elements attached to the left border of the Workbench desktop.

All screenshots in this documentation show the editor in the Workbench.

The Model-Explorer

The Model-Explorer is positioned in the upper left corner of the browser or the Workbench. With this explorer you can investigate the data model used by the query with all its dimensions, dimension levels, facts, attributes, standard formulas etc.

The Model-Explorer



Also, the explorer is the most important source for developing queries using drag& drop. All elements of the model as well as some additional elements (like colors or fonts) can be dragged out of the explorer into the query with the mouse. You can drag this elements onto the worksheet or onto the properties, which are displayed in the Property-Editor.

The Model-Explorer is displayed as in a tree-view (comparable with the File-Explorer from Microsoft Windows). Some elements can contain sub-elements, which can be opened (and closed) by clicking on the plus symbol to the left of the elements.

When working with the Workbench, the Model-Explorer uses the same location as all other explorers. You can switch back and forth between these with the tabulators below the explorer. If you use the editor in the Web-Frontend, only one explorer will be visible and switching is not necessary.

Information for models

When opening the model node (the first entry in the explorer) the following information about the model will be displayed:

- **Fact:** Under the first node (the sum symbol) all facts of the model are shown. The name of the node is equal to the name of the fact-dimension of the model (an auxiliary dimension in which all facts are stored) and can differ from model to model.
- **All dimensions:** Directly under the fact-node, all dimensions (with the dimension symbol) are shown with their name.
- **The entire hierarchy of a dimension, starting with the rootkey:** For each dimension the root-node of the dimension is shown. By opening this node you can investigate the entire hierarchy of the dimension with all its keys. Keys are marked by the key symbol, a filled out blue circle.
- **All attributes of the dimensions:** Each dimension owns a set of attributes, which you can find under the "attribute" node. All keys of the dimension share these attributes, however not all keys must have values for each attribute. Attributes are marked by the attribute symbol, a filled out green circle.

- **Standard formulas for the dimensions:** The explorer offers some standard formulas for each dimension, which are useful when creating reports. These formulas become visible when opening the node "Predefined of expressions". For formulas, the formula symbol is used.
- **All levels to the dimensions:** Each level of a dimension hierarchy has its own name. All levels with their names are displayed below the dimension node. By opening the level-node (marked by the level symbol) the standard formulas for each level will become visible (see below).
- **Standard formulas for the levels:** As is the case for the dimensions there are standard formulas for levels. These formulas also use the formula symbol.

Fonts

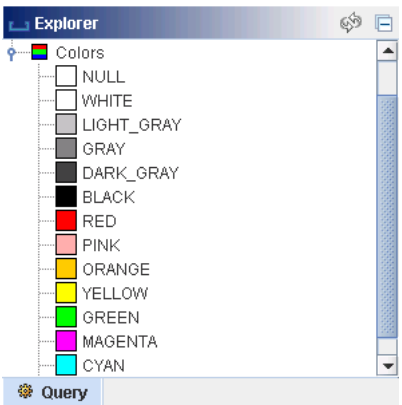
Below the node "Fonts" some standard fonts are offered, which can be used you when creating queries. However, these are not all available fonts - you may use any font being installed on you computer and on the report server.

Fonts in the Model-Explorer

Colors

As is the case of fonts, there are also some standard colors under the node "Colors", which can use you in your queries. Also, these are not all available colors you can use but only a small cutout.

Colors in the Model-Explorer



The Property-Editor

The Property-Editor is the table in the left, lower corner of the editor (or of the Workbench). The editor is used for investigating and editing the properties of all elements. Whenever you select an element in the worksheet by clicking on it with the mouse, then all properties of this element are displayed and can be edited.

The table showing the properties

Properties

Property	Value
Author	admin
CSS	
Date	30.10.05 14:25
Description	
Filter	
Formats	HTML,EXCEL,PDF,CSV
Icon	
Logo	
Minimum Width	400
Model	demo
Name	Chart example
Print Height	'29.7cm'
Print Logo	
Print Stylesheet	

Editing properties

In order to change a property, you must click into the value field in the second column, right to the name of the property. After the cursor appears in the table cell, you can edit the value.

Changing properties in the table

Stylesheet	
Title	'Turnover distribution'
Visible	true

Press the "Enter" key or click anywhere outside the cell to confirm the input. Then the system will check the new property value (e.g. for syntax errors in formulas). If an error was detected, it will be displayed in the status line of the browser or Workbench.

Note that you only can see the error message in the browser if the status line of the browser is visible. If you cannot see the status line, you should activate it, e.g. by using the menu-item "View/Status line" in the Internet Explorer.

If no error was detected, the editor will stop the input mode and you will be able to edit other elements afterwards. Otherwise, the input cell will show a red background.

An input containing an error

Stylesheet	
Title	'Turnover distribution
Visible	true

In the case of an incorrect input you must correct this before you can leave the input field. Alternatively you can use also the "ESC" to reset the property to the previous value and stop editing.

When selecting an element or after changing a property, the system checks all properties of the element and shows an error message if any error was detected. This is useful, because some properties depend on other properties or could have become faulty, e.g. when a dimension was removed and is still referred in the properties.

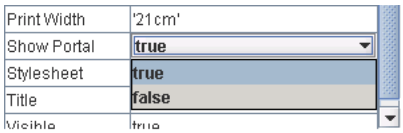
An error message produced by the property checker

Error in element detected: No height defined for chart

Default values

Some properties offer default values from which you one by opening the combo box (the small triangle beside the input field).

Default values for a property

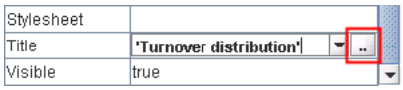


The previous value of the properties is replaced by clicking on a value. After the selection you still can change the property back to its previous value by using of the "ESC" key.

Special editors

For some properties (formulas or colors) the Property-Editor offers special editors, which you can use to edit a value instead of using the simple input-field. If a property offers such an editor, a button containing three points "..." will be visible right to the input field. Press this button in order to edit the property with the editor.

The edit button for properties

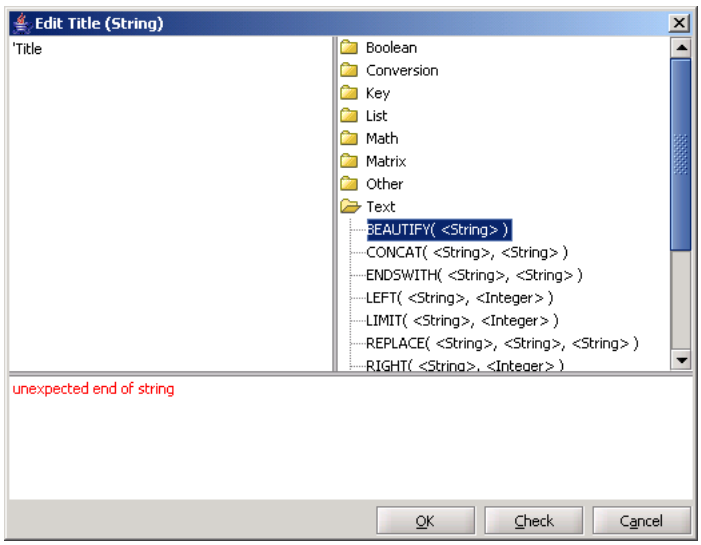


Also, most editors offer a check function, which you can use to examine your input without leaving the editor. Press the button "Check" to examine your input.

The Expression-Editor

The Expression-Editor is probably the most frequently used editor for properties. With this editor you can create and edit formulas, which are used in many properties of the report-elements. Read the "instantOLAP Reference" for a complete overview of the instantOLAP expression-language and its functions.

The Expression-Editor



The editor is divided into three different areas: The input-area in the upper left area (for editing the text of the formula), the function-table on the right side and the message field at the lower edge.

In the function-table all functions offered by the instantOLAP server (grouped by their topic) are listed. For each function you can see its syntax (i.e. the expected data types of their arguments).

With a double-click on a function, the function will be insert at the current caret-position inside the input-field.

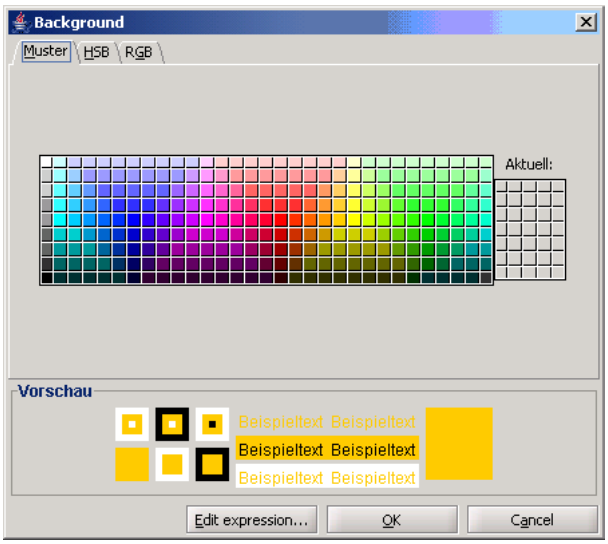
The message field shows possible errors being detected in the current formula.

Use the "Check" button in order to examine your formula on its correctness and to, if any error was found, show the error in the message-field.

The Color-Chooser

The Color-Chooser is used for choosing colors, e.g. for backgroundor font-colors. It offers three different kinds of a selection: You can directly choose a color from a number of pre-defined colors, define a color with its RGB value or define a color using a HSB model (Hue, Saturation and Brightness).

The Color-Chooser



Like other editors, the Color-Chooser has a formula as result, which contains only the selected color as a string constant. If you like to compute a color with a more complex formula, you can change from the Color-Chooser to the Expression-Editor with the button "Edit expression..." at any time.

If you edit a color property which already contains a complex formula, the Expression-Editor will open directly when pressing the edit button of the property.

The Toolbars

The toolbar of the Web-Frontend

If you opened the Query-Editor from the browser, a toolbar is displayed above the editor. This toolbar offers general functions like saving the report or closing the editor. The following buttons are available:

×	The "Close" button closes the editor and returns to the HTML view of the query.
	The "Save" button saves the current query.

	With the "Save as" button can you save the current query with another filename.
	With the "Undo" button you can undo the last change in your query.
	With the "Redo" button you can redo the last change in your query and revert the last undo operation.

The toolbar of the Workbench

Like the browser, the Workbench contains a main toolbar with general functions. However, not all of these functions are needed for editing queries. The following buttons are available for within the workbench which are useful when editing queries:

	The "Save" button saves the current query.
	The "Save as" button stores the current query under another filename.
	With the "Undo" button you can undo the last change in your query.
	With the "Redo" button you can redo the last change in your query and revert the last undo operation.
	The "New query" creates a new query.

All functions of the toolbar can also be called from the menu of the Workbench.

The toolbar of the worksheet

Beside the main toolbar for the general functions there is a second toolbar at the right edge of the worksheet, which offers functions for the creation of new elements within the queries.

The following functions are available in this toolbar:

	The "Add pivot-table" button adds you a further pivot table to the query.
	The button "Add line-chart" adds a pivot table to your query, which will be displayed as a line chart.
	The button "Add bar-chart" adds a pivot table to your query, which will be displayed as a bar chart.
	The button "Add pie chart" adds a pivot table to your query, which will be displayed as a pie chart.
	The button "Add comment" adds a comment to the query.
	The "Add list" button adds a list to your query.

	The button "Add newsticker" adds a newsticker to the query.

The preview panel

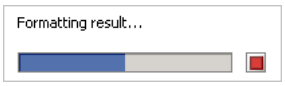
While editing a query, you can perform a preview of its result at any time without leaving the editor.

Performing a preview

The preview tabulator below the worksheet

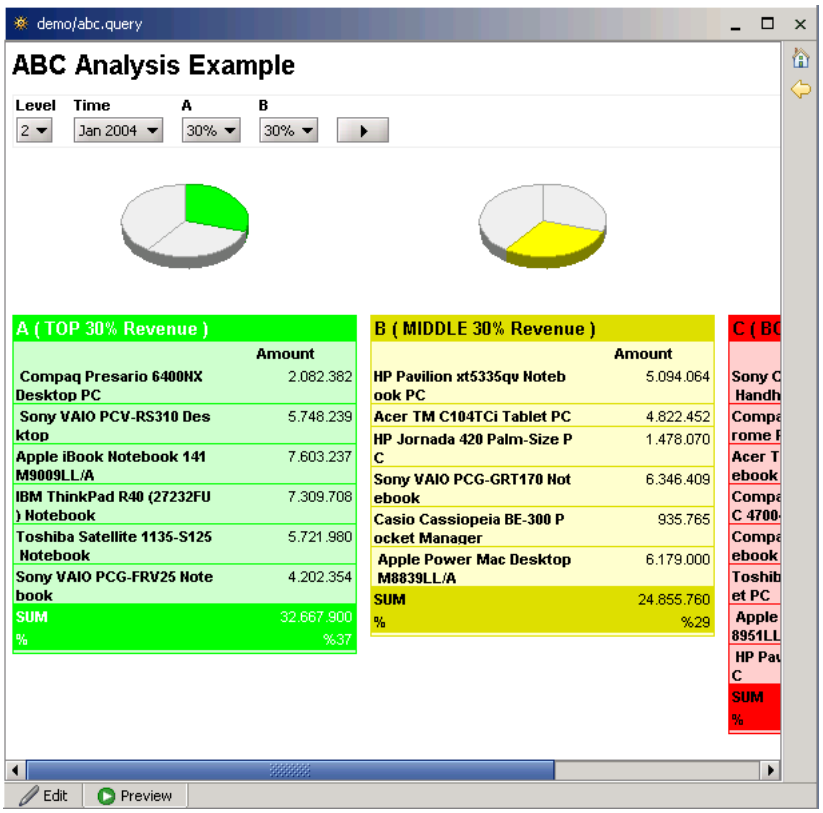
In order to switch from the design mode (when the worksheet is visible) into the preview mode, you must use the "Preview" tabulator underneath the worksheet. After switching to the preview, the query will be executed immediately which is indicated by a progress-bar.

The progress-bar



After the server successfully processed the query, the result will be displayed in the preview window.

Previewing a query in the Workbench



If the query and the model used by it are error-free, you will see the preview of the query after its execution. Otherwise, an error message will be displayed inside the preview window.

Leaving the preview

The "Edit" tabulator below the worksheet

In order to return to the worksheet, you must use the "Edit" tabulator (left of the "Preview" tabulator).

Restrictions of the preview

The preview does not offer not all features of the later HTML view or PDF export. These are the main restrictions:

- **Different display of queries:** Though the presentation inside the preview is quite similar to the HTML-presentation, it can slightly differ from. Also, some display-formats (e.g. the newsticker) are not supported.
- **No post-processing of queries:** The export of queries is not possible in the preview. Also, other workgroup-functions like saving queries or snapshots, are not available.

Editing queries

This section describes how to work with queries and how to create, edit or delete of individual elements.

Storing queries

Queries can be saved at any time using the menu-item "Save" or "Save as" (in the Workbench) or using the "Save" or "Save as" buttons from the toolbar.

Unless you save a query, the current changes are only visible for you and not for other users.

Editing elements

There is generally the same number of possible operations for nearly each element: You can create, move, delete or select most elements and change their property-values.

The only exception is the worksheet, since this can't be created, moved or deleted because the worksheet represents the query itself. But you can edit the properties of the worksheet (and thus the properties of query).

For each element-type its most important properties are described here.

Creating elements

Usually you can use the mouse to create new elements, e.g. with drag&drop out of the Model-Explorer, using a context-menu or button or by copying existing elements.

This section describes how to create new elements for all types of query-elements.

Moving elements

All elements except the worksheet can be positioned (with certain restrictions) freely. This section describes how to move elements for each type.

Deleting elements

Each createable element can also be deleted. The only exception is the worksheet itself.

Selecting elements

All elements of a query (including the worksheet) can be selected in order to change their properties in the Property-Editor. Selected elements are displayed with a blue border.

Changing properties

The properties of the currently selected element can be edited in the Property-Editor in order to change its behaviour and appearance.

Nearly all properties of all elements are formulas and are computed at the time of the report execution. Though you also may use simple constants in properties (e.g. if you want to set a background color to a certain and independent value), you must code this value in the correct syntax for instantOLAP-formulas.

Nearly all properties of all elements are controlled by formulas. Therefore you must use the correct syntax for instantOLAP-formulas - also for constant values. Otherwise the system will show an error message.

E.g. the syntax for a constant blue color would be "'blue'" or "'#0000FF", in each case enclosed into simple delimiters. These delimiters mark a constant string in an expression.

Read the "instantOLAP reference" for a detailed description of formulas and their syntax.

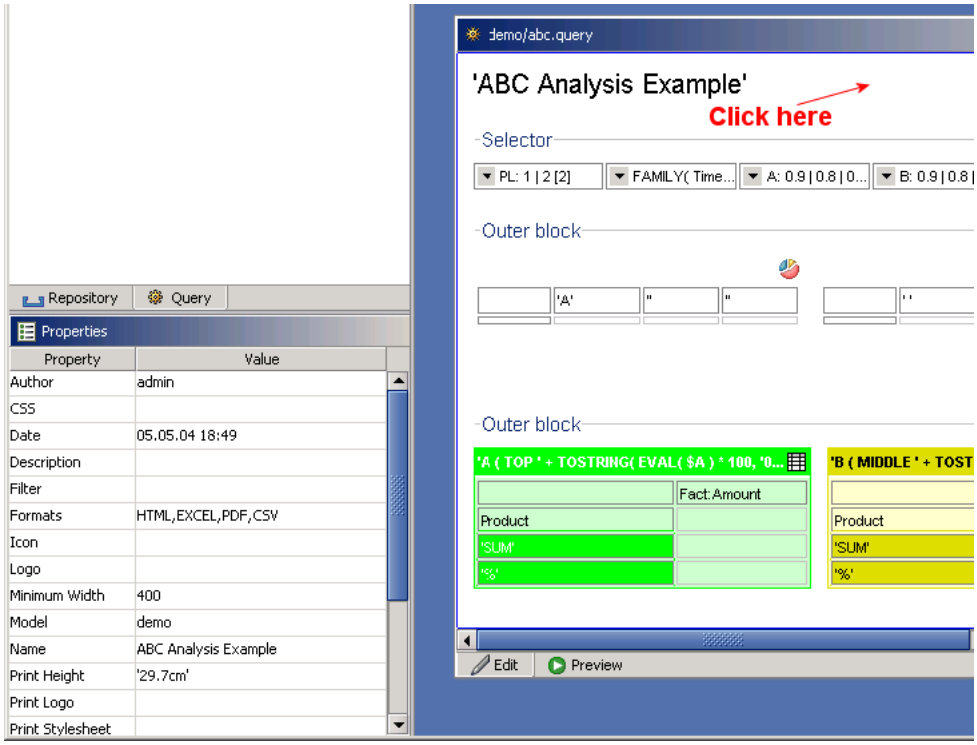
Editing the worksheet

For the worksheet there are only very few possible operations - you can only view and edit the properties of a worksheet.

Viewing and editing the properties

To view the properties of a query must select its worksheet. Click onto the title or any free space of the worksheet to select it.

The properties of a worksheet are displayed after its selection



After you successfully selected the worksheet, it will be displayed with a blue border and its properties will become visible in the Property-Editor.

Setting the query-title

The title of a report can be changed by editing the Title property of the worksheet. You must select the worksheet before editing its properties.

When creating a new query, the Title property will contain the standard title which was defined in the Query-Wizard.

Setting the query name

The name of a query is a constant text which is shown in the index-pages of the Web-Frontend.

In order to change of a query, you must must change the value of the property "Name" of the worksheet.

This Name property expect no formula and you must enter the name without text-delimiters here.

Setting the description

The description of a query is also shown in the index-pages of the Web-Frontend. It is defined by the property "Description".

Like the name of of a report, the Description property expects a constant value and no formula. You must enter the description with text-delimiters.

Setting the visibility

Queries can be visible or hidden. Users may still use invisible queries but they won't see them in the navigation of the Web-Frontend. Invisible queries are mostly used a linked detail query and are not directly to be called by the user.

You can change the visibility of a query by changing its property "Visible".

This property expects no formula and therefore only accepts the constant values "true" or "false".

Setting the symbol

The symbol of a query, like its name and the description, is shown in the index-page of the Web-Frontend. By changing the symbol you can replace the standard picture of a query, e.g. with you own symbol a small screen-capture as preview.

To set the new symbol you must change the property Icon of the worksheet.

This property expects a constant value (no formula) which must contain the URL (relative or absolute) of the icon.

It is important that the user has access to the symbol-URL from his workstation. The best way to ensure this is to copy the symbol into the same repository folder as the queries and only use relative URLs in the queries.

Setting the logo

The logo of a query is shown in the upper left corner of the Web-Frontend. You can define a logo both for the index-pages and for individual queries. If no logo was specified, the standard logo "/iolap/logo.gif" will be used.

In order to use a different logo you must edit the property Logo. Since the logo will be only displayed when executing a query, the property can be computed and expects a formulas as its value, contrary to the name, description or symbol of a query.

The URL of a logo must be a relative or absolute URL and always end with the extension ".gif", ".jpg" or ".png".

Beside the logo for the Web-Frontend you can also determine a print logo that will be used when converting a report in to a PDF file. If no print-logo was defined, the system uses the standard logo of the query for the PDF conversion.

Setting the print-format

For the conversion of queries into the PDF format you can define the measures of the pages with the properties Print Width and Print Height.

These properties expect a number followed by a unit, e.g. "cm" or "in". This combination of numbers and text makes it necessarily that you put both in a string, e.g. "'21 cm'".

The most common print-formats can be chosen directly using the the context-menu (see below) of the worksheet without having to change these properties manually.

Setting the stylesheets

instantOLAP uses stylesheets to format the queries when being displayed (in the Web-Frontend). You may adapt these stylesheets and the layout of a query for your needs and use different stylesheets for different queries (which has to be created before by your systemadministrator).

There are two different kinds of stylesheets: CSS files and STX stylesheets. CSS files are the simpler form of stylesheets with which you can change most display-properties of your query, e.g. the colors, fonts, distances etc. But the basic structure and layout of a reports is controlled by a STX-stylesheet and cannot be changed by using a different CSS file.

In difference to CSS files, STX Stylesheets offer the flexibility of changing the layout of the HTML result in every was but are more complex.

For both the CSS file and the STX-stylesheets there is a default implementation, which is provided with the installation of instantOLAP.

Setting the CSS stylesheets (for HTML)

The CSS stylesheet is specified by the property "CSS" of the worksheet. This property expects an URL, which (either relative or absolutely) refers to the CSS file. CSS files usually end with the extension ".css", therefore your URL should likewise end on ".css".

The stylesheet is used at the time of the query execution and is computed with formulas. Therefore this property always expects a formula which has a string as result.

Setting the STX-stylesheets

Like the CSS Stylesheet, you can also determine the STX-stylesheet with a formula, which is computed at the time of the report execution and used for the HTML-conversion of a query.

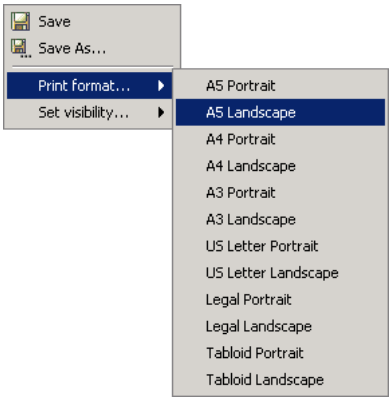
The STX-stylesheet is defined by the property Stylesheet of the worksheet. Contrary to the CSS property, this property expects no URL (because the STX-stylesheet is not referred by the browser), it simply expects the name of the stylesheet.

All STX-stylesheets used by reports must be placed in the instantOLAP-repository under the folder "stylesheets".

The context menu

The worksheet has its own context menu, which appears when clicking with the right mouse button on any free part of the worksheet (or on its title).

The context menu of a worksheet



Apart from saving the report (with "Save" or "Save as") you can set some properties of the report to pre-defined values here. Those are the print-format and the visibility of the report.

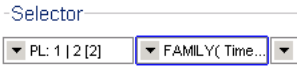
Editing selectors

Viewing and editing properties

In order to edit the properties of a selector you must select it with the mouse before.

After you successfully selected the selector, a blue border will be drawn around the element and its properties will be shown in the Property-Editor.

A selected selector (with blue border)



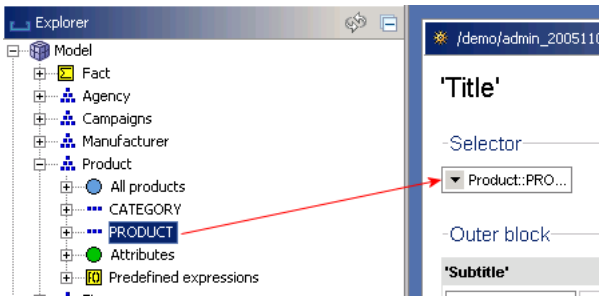
Creating selectors

There are different ways to create a selector:

Creating a selector using drag&drop out of the Model-Explorer

The simplest way to add a new selector to a query is to drag an element out of the Model-Explorer. Navigate to an expression (e.g. a dimension, a dimension-level, an individual key or fact or a formula) inside the explorer and drag this with mouse on a free space within the selector-area.

Create of a selector by dragging an expression out of the Model-Explorer



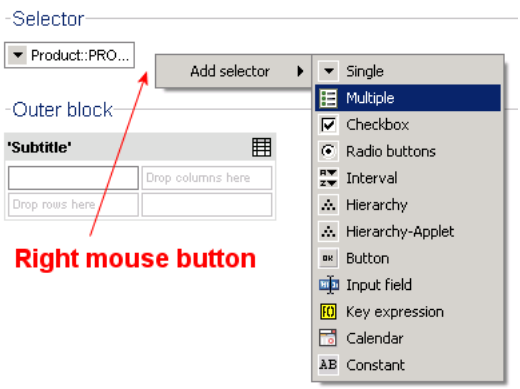
When dropping the expression, a new selector will be added at the end of the selector list (with the expression as its option-expression). The new selector will have the type "Single" or "Hierarchy-Applet" and has no standard selection (you can change both later). You can change the position of the selector within the selector-area afterwards with the mouse (read the section "moving selectors").

The type of the new selector depends on the element you drew out of the Model-Explorer: If you used a dimension, a Hierarchy selector will be created (which allows the user to select a key of a dimension in a hierarchical way). Otherwise a selector of the type "Single" will be created, which allows the selection of an individual key from a list of keys.

Creating selectors using the context menu

You can also create an empty selector manually by using the context menu of the selector-area and set its formulas and options afterwards. Use the right mouse-button to open the menu and choose the menu-item "Add select".

Creating a selector using the context menu



Extending selectors using drag&drop

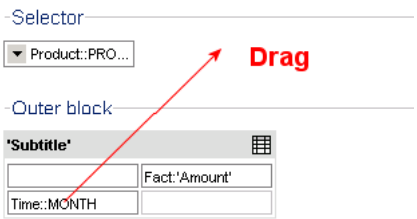
You may also extend the options of an existing selector with drag&drop. Drag an expression from the Model-Explorer on a selector and drop it there (a blue border will signal that a selector is the target of your drag&drop operation).

Then the Options property of the selector will be extended by the new expression and the selector will offer this new options when executing the query.

Moving a header from a pivot table with drag&drop

A further possibility to create a new selector is to drag a header out of a table into the selector-area. Like dragging an expression out of the Model-Explorer, this will create a new selector of the type "Single" and without a standard selection. The options of the new selector is equal to the iteration of the header.

Creating a selector from a header

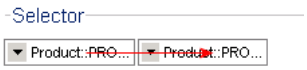


The header used as source disappears after moving it into the selector-area. If you wish to keep the header, press the key "CTRL" while dragging the header.

Copying selectors with drag&drop

In order to copy a selector with drag&drop, you must drag it to the desired position within the selector area, like you would do when moving a selector. While dragging the selector you must keep the key "CTRL" pressed so that the mouse cursor shows the copying symbol (a plus sign).

Copying a selector with drag&drop



Deleting selectors

There different ways to delete a selector:

Deleting selectors using the keyboard

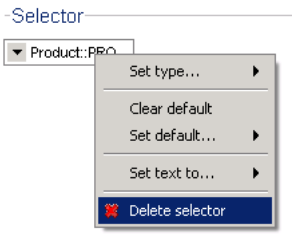
You can use the key "Delete" to remove a selector from the query. First, you must select it with the mouse (a blue border around the selector will show its selection). If you press the key now, the selector will be deleted.

The system will delete the selector without asking you before. If you falsely deleted a selector you can use the "Undo" button from the toolbar to undo the deletion.

Deleting selectors using the context menu

Instead of the keyboard you can also use the context menu of a selector to delete it. Open the context menu using the right mouse button and use the menu item "Delete" to remove the selector.

Deleting a selector using its context menu



Changing the type of a selector

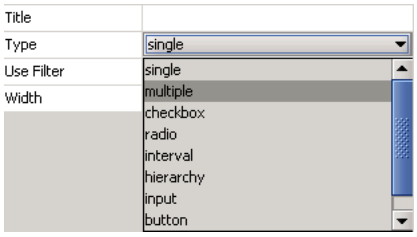
The type of a selector can be changed either by editing its properties or using the context menu:

Changing the type using the properties

In order to change the type you must select the selector first. Click on the selector until a blue border indicates, that the selector is currently selected. Now you will see its properties in the Property-Editor.

The property "Type" holds the type of the selector. This property offers a list of all possible selector-types, from which you can select the desired.

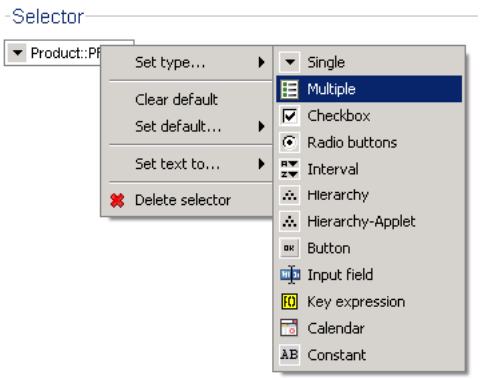
Changing the type of a selector using the property



Changing the type using the context-menu

The faster way to change the type is to use the context menu of a selector. Open the context menu using the right mouse button while holding the mouse over the selector. Now choose the desired type from the submenu of the menu-item "Set type...".

Changing the selector type using the context menu



Setting the options

The property Options of a selector determines the options of a selector from which the user can select entries when executing a query (e.g. "every year", months, products etc.). The options are defined with a formula, the result of this formula is converted into the options.

There are two different ways of setting the options: By dragging expressions out of the Model-Explorer or by directly manipulating the property "Options" of the selector.

Extending the options with drag&drop

The Model-Explorer offers different elements of the model and standard formulas, from which you can drag the needed into the selector-area or onto an existing selector to define the selector options.

The options of the selector are then extended by this new expression (the expression is added to the previous expression using the join operator "|").

Changing the options property

You can alternatively edit the property Options manually and change the formula with which the options are computed. Before editing the property you must select the selector-element so that its properties become visible in the Property-Editor. This is the common way to set the options for more complex expressions.

Setting the default selection

The default selection of a selector determines, which element of the options is selected when the user executes the query the first time. If you set no standard selection for a selector (means the property "Default" is empty) the selector will be empty with the first call of the query.

There are two different ways of setting the standard selection: By directly editing the property Default or by using the context-menu of the selector:

Changing the default property

The property Default determines (using a formula) which options of this selector are initially selected. Select the selector (until it is framed with a blue border) and change the property in the Property-Editor.

The count of the default values the formula should return depends of the selector type. Interval selectors need two default values, the first one is used for the first (start) selection and the second one for the second (to) selection.

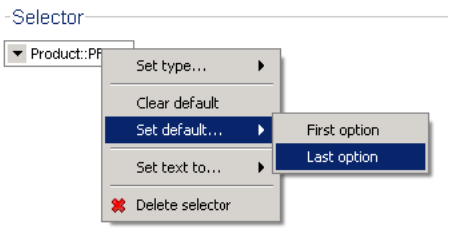
Make sure that the result of the function is a part of the options.

Using the context menu

Using the context-menu of the selector is a very simple way of setting the default selection for a selector. You can only set the default selection to the first or last option offered by the selector with the menu.

Open the context-menu of the selector by using the right mouse button. Then use the menu-item "Set default..." and choose the first or last options as the new default selection for the selector.

Setting the default selection with the context menu



Changing the default selection using the context menu also changes the property "Default".

Deleting the default selection

You can delete the default selection of at any time, either by deleting the value of the property Default or by using the menu-item "Clear default" from the context menu of the selector.

Setting the title of a selector

The title of a selector is being displayed above the selector in the result. Usually the name of the selector or name of the selectable dimension is used its title.

Alternatively you can assign a different title to a selector. Edit the formula stored in the property Title to change the title. The formula must have a text as result (usually a constant string is used here, e.g. "'month'" or "'product'").

Selector with title

If you change the title of a selector, then the formula stored in the "Title" property will be displayed inside the worksheet directly above the selector.

Defining the option-texts

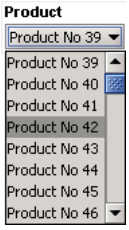
Apart from the title of the selector you can change the texts being displayed for the individual elements (the options) of a selector. A formula is used for the computation of the option-texts.

Usually the IDs of the option-elements are used as text within the selector. By changing the property text you can define a formula which returns different texts for its options.

This formula will be computed for each single option of the selector, whereby the filter for the computation always refers another element of the options (therefore inside the text-formula you can e.g. use the expression "product" to refer the currently selected and displayed product).

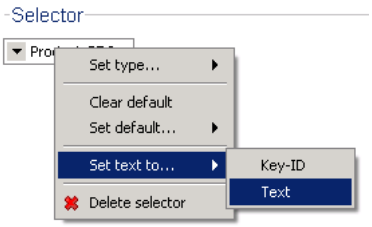
E.g. a formula "Product.ProductNo" would return the productnumbers of the products offered as options and the selector would display these product-numbers instead of the product-ids.

A selector with changed option texts



Instead of manually editing this property you can also use the context-menu of the selector to change the text-property. The menu will suggest a list of standard texts.

Setting the options texts with the context menu

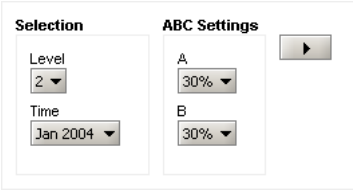


All text attributes of the options are shown here. Which attributes are offered depends on dimensions used in the options-expression, since different dimension own different attributes. Also, the menu could contain useless attributes because they could be technical values (e.g. technical database-IDs of the elements etc.).

Grouping selectors

Multiple selectors can be grouped to a selector-group. This groups arrange all their selectors vertically and there need less space in the selector-area than the selectors would need if they were not grouped.

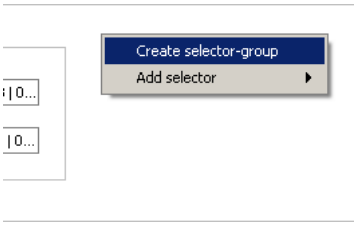
A sample selector-group



Creating a selector-group

To create a selector-group you must open the context-menu of the selector-area. Use the right mouse button above the area to open the menu.

Creating selector-groups using the context-menu of the selector-area

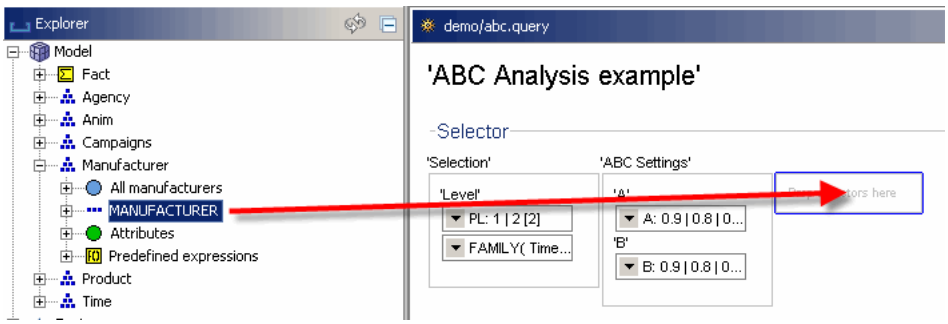


The use the menu item "Create selector-group". Then a new, empty selector-group will be appended to the selector-are.

Adding selectors to selector-groups

Adding new selectors to a selector-group works similary like adding selectors to the selector-area: You can drag key-expressions from the Model-Explorer or existing selectors or headers into the group. A blue border will signal the moment you can drop the selectors.

Adding selectors to a group using drag& drop

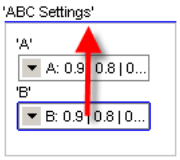


After you dropped the selector it will be added to the group. Depending on the border you dropped the selector, it will be appended at the end or added at the beginning of the group.

Aranging selectors within a group

After you added some selectors to a group you can arrange their order within the group using drag&drop. Select a header and drop it at the upper or lower border of anoter header inside the group or at the upper or lower border of the group. This will move the selector before or behind the other selector or to the top or bottom of the group.

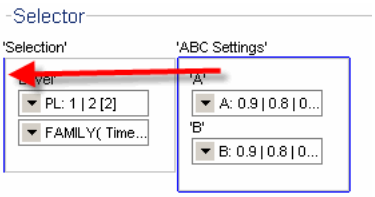
Arranging selectors within their group



Moving selector-groups

You can also move the whole selector-groups and change their position inside the selector area. Moving groups inside the area is like moving single selectors: Select the group and drag it onto the border of another selector or selector-group.

Moving selector-groups using drag&drop



Then the editor will move the group before or behind the other selector or group.

Copying selector-groups with drag&drop

You can also use drag&drop to copy selector-groups. Like when moving a group, drag it to the desired position within the selector-area but keep the CTRL key pressed.

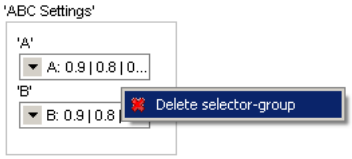
Then the editor will create a copy of the group and all its selectors and place it at the target position.

Deleting selector-groups

There are two different ways to delete a selector: Using its context-menu or by using the keyboard.

The context menu of a selector offers a menu-item "Delete selector-group". Open the context menu and use this item to delete a group.

Deleting a selector-group using its context-menu



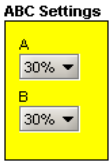
In order to delete a selector-group with the keyboard you must select it first by clicking onto its title. Then press the delete-key to remove the group from the selector-area.

Changing the appearance of selector-groups

There are several properties for selector-groups which define their appearance in the report:

- The "Width" and "Height" determine the size of the group
- "Title" defines the title of the group. Is not title is specified, it will remain empty
- "Background" defines the background-color of the group
- "Border" sets the border-color of the group

A selector-group with title, background- and border-color



The context-menu for selectors

Selectors have, as mentioned before, their own context-menu, with which different operations on the selector can be performed:

- Changing the type of a selector
- Setting and resetting the standard selection of a selector
- Setting the option-texts to a certain attribute
- Deleting selectors

Editing outer blocks

Viewing and editing the properties

In order view the properties of an outer block you must select the block with the mouse first. To select an outer block you must click onto its title or onto the free space under the dividing line of the outer block.

Select an outer block by clicking into its free space or title

After you successfully selected the block, a blue border will be drawn around the block and its properties will be displayed in the Property-Editor.

Creating outer blocks

There are different ways to create an outer block: Using the toolbar of the Query-Editor or by copying existing blocks within the query.

Creating blocks using the toolbar

Outer blocks cannot be created directly using the toolbar since no button is available for this operation. However, you can create a table or chart by using the toolbar which will also create an inner and outer block for this new element.

The "New table" button

E.g. use the "New table" button to create a new pivot table. Then an outer and inner block would be created together with the pivot-table.

Copying an existing blocks with drag&drop

Existing blocks can be copied with drag&drop. Select the block you want to copy and drag this, while pressing the "CTRL" key, to the desired position. Whenever the mouse is located on a valid target-position, a blue line will become visible in the worksheet and indicate this. Now you can drop the block here and create a copy of it (the copy will also contain copies of all nested inner blocks and pivot tables, too).

Moving outer blocks

Existing block can also be moved with drag&drop. Select the block and drag it, using the mouse, to the desired position. Again, a blue line in the worksheet indicates valid target-positions.

Deleting outer blocks

Outer blocks can be deleted using the keyboard or with their context menu:

Deleting blocks using the keyboard

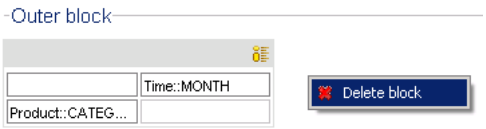
You can use the "Delete" key to remove an existing outer block. Before that, you must select the block with the mouse until a blue border is drawn around it. The press the key and the block will be deleted.

The system will delete the block without asking before. If you falsely deleted a block you can use the "Undo" button from the toolbar to undo the deletion.

Deleting blocks using their context menu

Alternatively you can use the context menu of an outer block to delete it. Select the block and open its context menu using the right mouse button. Then use the menu item "Delete block".

Deleting a block with its context menu



Setting the iteration

For blocks (both for outer and inner block) you can define an iteration. A block with an iteration will be repeated for each key which is part of the result of the formula stored in the Iteration property.

The formula must return a number of dimension-elements (key) and the block will be repeated for each key which ist part of the result of the formula.

An outer block iterated over the manufacturer dimension

Jan 2004			
	Quantity	Planned	Amount
Compaq Presario 6400NX Desktop PC	2.233	2.254	2.082.382

Feb 2004			
	Quantity	Planned	Amount
Compaq Presario 6400NX Desktop PC	2.023	2.063	2.013.312

Mrz 2004			
	Quantity	Planned	Amount
Compaq Presario 6400NX Desktop PC	2.104	2.153	2.019.925

Apr 2004			
	Quantity	Planned	Amount
Compaq Presario 6400NX Desktop PC	1.976	1.948	1.894.065

Setting the iteration using the property

The most direct way to set the iteration is to edit the formula stored in the property "Iteration". This property expects a formula with dimension-keys as its result.

Setting the iteration using drag&drop

You can also edit the iteration by dragging an element (e.g. a dimension level or any other expression) from the Model-Explorer into the block. Then the iteration of the block will be set to this expression or it will be extended (if the iteration is already set) with the new expression.

The context menu for outer blocks

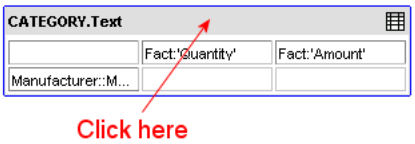
All blocks own, as mentioned before, a context menu. The only operation offered by this menu is the deletion of an existing block.

Editing inner blocks

Viewing and editing properties

In order view the properties of an inner block you must select the block with the mouse first. To select an inner block, you must click on its title.

Select an inner block by clicking onto its title



After you successfully selected the block, a blue border will be drawn around it and its properties will be shown in the Property-Editor.

Properties for charts

If you display a block as a chart (when using a chart-format as the display format for a block), a set of additional properties will become visible in the Property-Editor. This properties define the layout and the presentation-form of the chart.

Creating inner blocks

There are different methods to create an inner block: You can use the toolbar of the Query-Editor or you can copy existing blocks within the query.

Creating blocks using the toolbar

Inner blocks cannot be create directly with the toolbar, since no button for creating inner blocks exists. However, you may create a table (or chart) using the toolbar which also will create a new outer and inner block.

The "Add table" button

E.g. use "Add table" button to create a new pivot table. Then an outer and inner block would be created together with the pivot-table.

Copying blocks using drag&drop

Existing inner blocks can copied with drag&drop. Select the block and then drag it, with pressed "CTRL" key, to the desired position. Whenever the mouse is located on a valid target-position for the block, a blue line will become visible in the worksheet. Now you can drop the block here and create a copy of it (the copy also will contain copies of all nested pivot-tables, too).

Inner blocks can be copied both into existing outer blocks and directly onto the worksheet of the query. A new outer block will be automatically created when dragging an inner block onto the worksheet.

Moving inner blocks

Existing block can also be moved with drag&drop. Select the block and drag it, using the mouse, to the desired position. Again, a blue line in the worksheet indicates valid target-positions.

Deleting of inner blocks

Inner blocks can be deleted using the keyboard or by using their context menu:

Deleting blocks using the keyboard

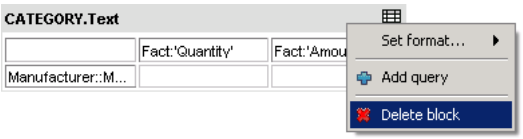
You can use the "Delete" key to remove an existing inner block. Before that, you must select the block with the mouse, until a blue border is drawn around it. Then press the key and the block will be deleted.

The system will delete the block without asking before. If you falsely deleted a block you can use the "Undo" button from the toolbar to undo the deletion.

Deleting blocks using their context menu

Alternatively you can use the context menu of an inner in order block to delete it. Select the block and open its context menu using the right mouse button. Then select the menu item "Delete block" to delete the block.

Delete a block by its context menu

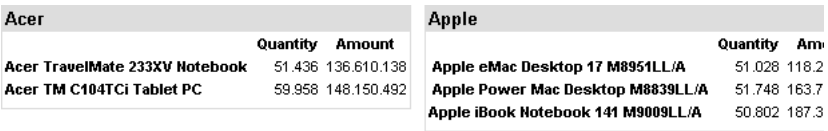


Setting the iteration

For blocks (both for outer and inner block) you can define an iteration. A block with an iteration will be repeated for each key which is part of the result of the formula stored in the Iteration property.

The formula must return a number of dimension-elements (key) and the block will be repeated for each key which ist part of the result of the formula.

An inner block which repeated over the months



Acer

	Quantity	Amount
Acer TravelMate 233XV Notebook	51.436	136.610.138
Acer TM C104TCi Tablet PC	59.958	148.150.492

Apple

	Quantity	Am
Apple eMac Desktop 17 M8951LL/A	51.028	118.2
Apple Power Mac Desktop M8839LL/A	51.748	163.7
Apple iBook Notebook 141 M9009LL/A	50.802	187.3

Setting the iteration using the property

The direct way to set the iteration is to edit the formula stored in the property "Iteration". This property expects a formula with dimension-keys as result.

Setting the iteration using drag&drop

You can also edit the iteration by dragging an element (e.g. a dimension level or any other expression) from the Model-Explorer into the block. Then the iteration of the

block will be set to this expression or it will be extended (if the iteration is already set) with the new expression.

Setting the block title and -font

Inner block can have sub-titles which are displayed above their contents. To define such a sub-title you must set the property "Title" to a appropriate formula.

You may also set the font and size for the title. Edit the properties "Font" and "Font Size" to change them.

Setting the frame-color

A frame is drawn around the block, the color of this frame is always equal to the background color of the block title. If you want to change the frame color, you can define the color using the property Border (which expects a formula returning a color-name or the RGB value of the color).

An inner block with black frame

Complete period		
	Quantity	Amount
Acer	111.394	284.760.630
Apple	153.578	469.249.168
Casio	51.438	22.425.776
Compaq	211.480	330.706.060
HP	158.306	224.483.638
IBM	51.638	171.743.184
Sony	203.616	432.825.626
Toshiba	109.136	305.099.026

If you want to make the frame invisible, leaving the property "Border" empty would not help - the frame is painted with the standard color for frames then. Instead you must set the frame color equal to the background color of the query (usually "'white'").

Setting the block format

The block format of an inner block determines the representational form of the pivot tables contained in it. Inner blocks may display their contents as tables, charts or newsticker.

Setting the block format by editing the format property

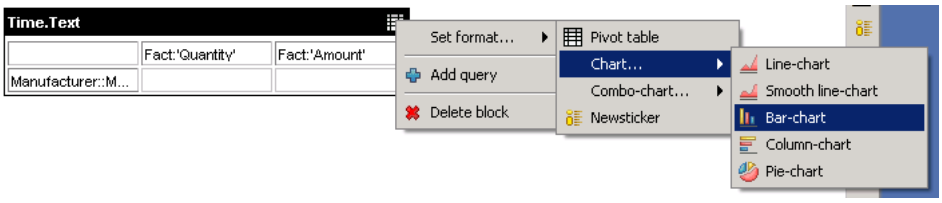
The property format of an inner block determines the representational form of its pivot tables. This property does not expect a formula, only a constant string with one of the following values:

- **table** displays the pivot tables as a simple table. This is the standard display format for blocks.
- **easychart** displays the pivot tables as charts, using an Java Applet in the browser. With the further property "Chartformat" you can define the type (line, bar, pie or spline) of the chart.
- **easychartservlet** also displays the pivot tables as charts but without using a Java Applet in the browser. The charts are generated as PNG images on the server side then. With this format you can use charts in your reports even if the viewer has problems with Java Applets.
- **newsticker** represents pivot tables as a newsticker.

Setting the block format using the context menu

With the context menu of an inner block you can determine its display format. All charts types mentioned above are offered as menu-items. For charts, all charts type are available as menu-items under the items and "Combo Chart".

Setting the block format with the context menu

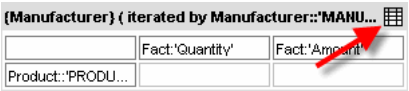


When setting a chart format using the context-menu, some chart properties will be set to predefined values which can save you some work. E.g. the block size, the legend settings and other properties will be set.

Always set the format for inner blocks with their context menu, especially if you want to use charts. Then, dependant on the chart type, many properties of a block and c hart are initialized with default values.

The current format of an inner block is always shown as a small symbol in the upper right corner of the block.

The format of a block (here a bar chart) is displayed as symbol in the upper right corner



Setting the block size

The properties Width and Height define the size of an inner block in pixels.

For blocks, whose contents is formatted as a simple table, you do not have to set these properties, since for these the size can be computed automatically by the system (although setting their size is still possible and will change the appearance of the block).

However, if the block is displayed as a chart, this properties must be set, since they also determine the size of the chart. If you have set the chart format of a block using its context menu, then size will be already set to default values.

Always set the size for inner blocks, whose contents is displayed as a chart. For this use the properties Width and Height.

Providing a link for a block

For inner blocks you can define links in order to refer to other report or URLs. The URL of a link must be set with the property Link. When the formula in this property returns a value, the link will be displayed underneath the block in the result.

Optionally you also add a name and an icon to the link by editing the properties "Link Name" and "Link Icon". The icon will be displayed next to the link text. The name will be (if defined) displayed instead of the link URL.

A block with an additional link

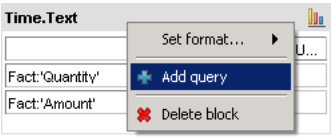
Adding further pivot tables to an inner block

Usually inner blocks contain only one pivot table.

However, for overlay charts (combined charts which consist e.g. of a bar and a line chart) it is necessary to define more than one pivot table in a single block. Each pivot table is then converted into a single layer of the overlay chart.

Use the context menu of the inner block to add a new pivot table to it. Open the menu using the right mouse button and then use the menu item "Add query" to add a new pivot-table to the block.

Creating a new pivot table using the context menu of the block



The context menu for internal blocks

Inner block own a context menu offering the following operations:

- Changing the display-format of the block
- Adding further pivot tables to the block
- Deletion of the block

Editing pivot-tables

Viewing and editing the properties of a table

In order to view and edit the properties of a table you must select it with the mouse before.

Since a table is not visible (only the corner and the headers of a table are visible), you must click on the upper, left corner of the pivot table to select it.

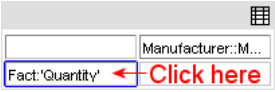
Click on the upper left corner of a pivot table to select it

After you successfully selected a table, a blue border will be painted around the corner of the table and the table properties will be displayed in the Property-Editor.

Viewing and editing properties of an individual header

Headers are placed within a pivot table and can also be selected. Click directly on a header in order to select it and view its properties in the Property-Editor.

Click directly on a header to select it



Creating new pivot tables

There are several ways to create a new pivot table:

Creating pivot tables using the toolbar

The common way to create a new pivot table is to use the button "New table" from the toolbar of the Query-Editor.

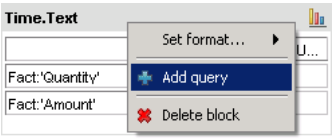
The "Add table" button

Use the button "Add table" to create a new pivot table. When creating a table, an additional outer and inner block are created and contain the new pivot-table.

Creating pivot tables using the context menu of the inner block

If you want to create an additional pivot table in an existing block you can use the context menu of the inner block. Select the block first and open its context menu using the right mouse button. Then you can create a new pivot table with the menu item "Add query".

Adding a new pivot table to a block using the context menu of the block



Usually you will not create pivot tables without a block. Only for creating combo chart there is a need for additional pivot tables in an existing block. Then use the context menu to create the (see below).

Creating and moving pivot tables with drag&drop

Pivot tables cannot be moved or copied using drag&drop. But you may move or copy the inner block containing the pivot table in order to move or copy the table with it.

Read the sections Editing outer blocks and Editing inner blocks to learn how to copy or move blocks with drag&drop.

Deleting pivot tables

Deleting pivot tables together with the containing blocks

Usually you wont delete a pivot table directly but together with the inner our outer block containing the table. To delete a block you must select it first and then press the "Delete" key or use the context menu of the block.

The system will delete the table without asking before. If you falsely deleted a table you can use the "Undo" button from the toolbar to undo the deletion.

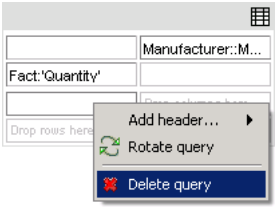
Deleting pivot tables using the context menu

You can also delete individual tables from an inner block until only the empty block remains.

For this select the pivot table you want to delete by selecting its upper left corner and then open its context menu using the right mouse button.

Use the menu item "Delete query" in order to remove a pivot table from its containing block.

Use the context menu to remove a table from its block



Creating headers inside a pivot table

Adding headers to a pivot tables is one of the main tasks when using the Query-Editor. The headers of a table determine the structure and the contents of the pivot tables and therefore of the entire query.

There are different methods to add new headers to a table: By dragging elements out of the Model-Explorer, using the context menu of the table corner or other headers or by dragging existing headers.

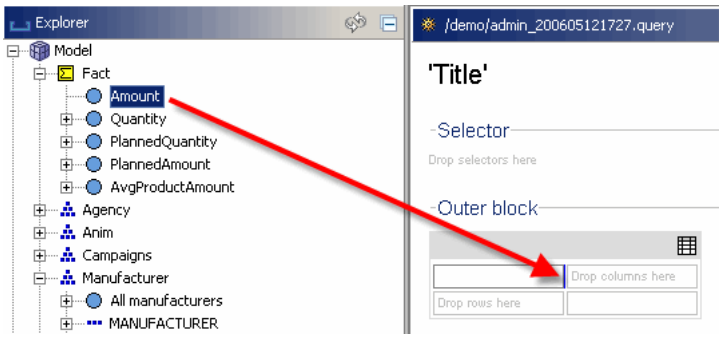
Creating headers using the Model-Explorer

The common way to add a new header to a pivot table is to drag an element out of the Model-Explorer. In the Model-Explorer you can find the mostly used elements of a model, including dimensions, dimension levels, dimension elements and facts, with which a query can be built up.

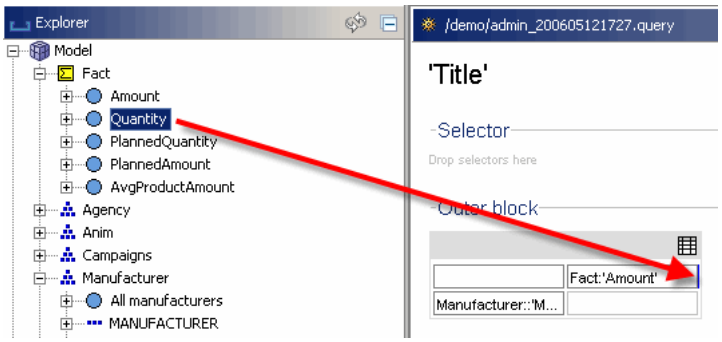
Drag the element you want to use out of the explorer and drop it at the position inside the pivot table where you want to create a new header. When placing elements you have to pay attention to place the elements at one of the edges of the table corner or of an existing header. Depending on the edge, the new header will be added and nested to table table in different ways:

- If you drop the new header at the right border of the table corner or at the left or right edge of an existing header in the x axis, then the new header will be added on the x-axis next to the existing header.

Drop new elements at the edge of an empty table

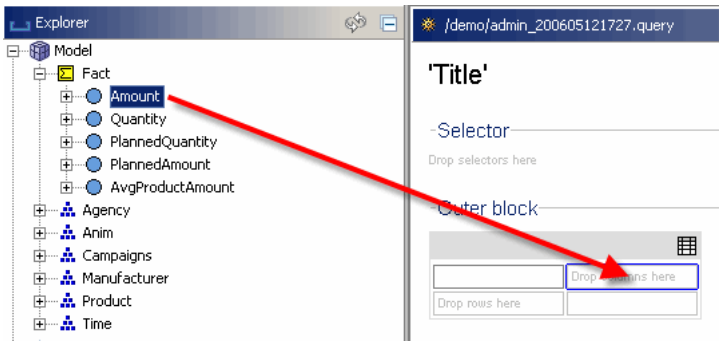


Extend the x axis at the right or left border of an existing header



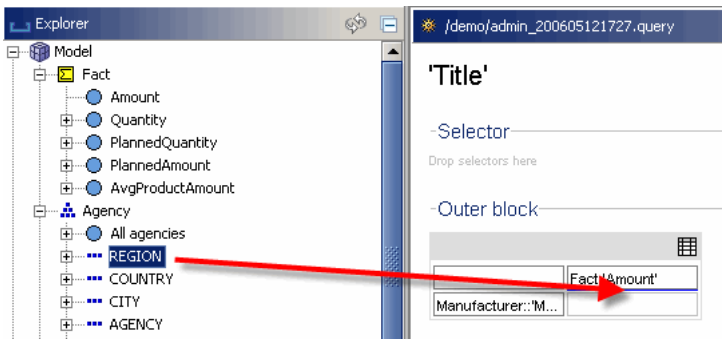
- On empty tables, you can also drop a header in the center of the placeholder named "Drop columns here".

Drop headers on empty tables on the placeholder



- If you drop the new header at the upper or lower border of an existing header in the x-axis, the new header will be nested under the existing one or the existing header will be nested under the new one (which then takes over the position of the existing header).

New headers can be nested under existing ones



- If you drop the new header at the lower border of the table corner or at the upper or lower edge of an existing header in the y axis, then the new header will be added on the y-axis next to the existing header.

Drop new elements at the edge of an empty table

Extend the y axis at the top or bottom border of an existing header

- On empty tables, you can also drop a header in the center of the placeholder named "Drop columns here"

Drop headers on empty tables on the placeholder

If you drop the new header at the left or right border of an existing header in the y-axis, the new header will be nested under the existing one or the existing header will be nested under the new one (which then takes over the position of the existing header).

New headers can be nested under existing ones

Whether a new header may be dropped at the desired position will be always indicated by a blue line at a valid position.

Creating headers using drag&drop within the table

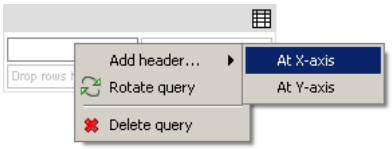
Not only the Model-Explorer can be used as source for drag&drop when creating a header, but also existing headers and selectors. To copy an existing header, you must press the key "CTRL" while dragging it to the desired position. Pressing the key changes drag&drop from the "move" the "copy" mode, which is indicated by a little plus sign in the mouse-cursor.

When using a selector as source, its options expression will be used as the iteration for the new header. When using another header as source, all properties of the existing header will be copied with it. Nested header will not be copied with a header, therefore you only can copy single headers.

Creating headers using the context menu of the corner

You can add empty header to a pivot table by using the context menu of its corner. Open the menu by using the right mouse button and use the menu item "Add header/At X-Axis" or "Add header/At Y-Axis" in order to create a new header on the x or y axis of the table.

The context menu of the corner can be used to create headers

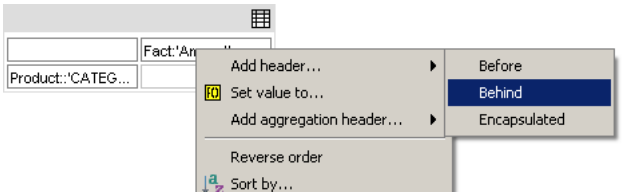


If the appropriate axis already contains headers, the new header will be inserted before all other headers and becomes the first header in the axis.

Creating headers using the context menu of other headers

You can also create new headers using the context menu of existing headers. Open the menu and use one of the menu items "Add header/Before", ".../Behind" or ".../Encapsulated".

Use the context menu of existing headers to create new ones



The new empty header will be created and inserted, depending the used menu item, before, behind or nested to the existing header.

Also you can create "aggregation headers" with the context menu. Aggregated headers will contain a formula which will compute e.g. the sum or average for a table row or column.

Moving headers

You can change the position of a header inside a table with drag&drop. For this, drag a header with the mouse to its new position.

Like when copying headers with drag&drop, the exact target position at which you drop the header will define where and how the new header is located (see above).

After moving a header it disappears from its old position. Its nested headers will then take over the old position of the header.

Deleting headers

Headers can be deleted in two different ways, using the keyboard or using their context menu. The deletion of an header only deletes the header itself, all nested headers remain and will take over the position of the deleted header.

Deleting headers using the keyboard

First select a header with the mouse. Then delete the header with the "Delete" key.

Deleting headers using their context menu

Header own a context menu which you can open using the right mouse button.

Use the menu item "Delete" from its context menu in order to delete a header.

Setting the contents and layout of the table corner

Setting the corner-text and -font

The corner of a pivot table can contain a text which is displayed in the upper left corner of the table. This text can be defined with the property Corner Text of the table.

Displaying a text in the corner of a table

Sales		
All amounts in EURO	Quantity	Amount
Desktop	258.260	524.724.708
Notebook	409.184	1.098.361.324
Pocket PC	203.920	122.257.350
Tablet PC	179.222	495.949.726

You can also set the color, font and font-size of the corner by changing the properties Corner Foreground, Corner Font, Corner Font Weight and Corner Font Size.

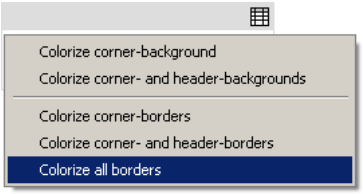
Setting the border- and background-color of the corner

Like for headers you can change the background color and the border color for each single edge of the corner. The background is defined by the property Corner

Background, the border color by the four properties "Corner Top Color", "Corner Bottom Color", "Corner Left Color" and "Corner Right Color".

A faster than editing this properties is to drag a color out of the Model-Explorer and to drop it on the corner. You can either change the color of a border directly by targeting them exactly with mouse or you can drop the color in the center of the corner. Then a popup menu will appear, offering different options for the coloration of the corner.

Colorize the corner's borders by dropping a color on it



Defining the layout for headers

For headers there is a large number of layout possibilities. Each header offers properties which define the layout of the header itself and properties for the cells generated by the header at runtime. This means, the layout of the cells will be defined by the header next to it or above it and not by the cells (which do not exist at design time).

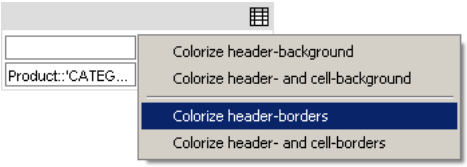
Setting the border colors for headers and cells

The border colors for headers are defined with the properties "Left Color", "Right Color", "Top Color" and "Bottom Color". The border colors for the cells generated by an header are defined with the properties "Cell Left Color", "Cell Right Color", "Cell Top Color" and "Cell Bottom Color".

Like with the corner, you can also use drag&drop to add colors to the borders of headers and avoid to edit their properties directly. Drag a color out of the Model-Explorer and place it exactly on the border of the header you want to set a color for. This will set the color of the border for the header and for the cells generated by this header.

When dropping a color in the center of a header, a popup menu will open and offer some operations for colorizing headers. E.g. use the menu-item "Colorize header-borders" to set the color for all four borders of a header at once.

The popup menu used to colorize headers



If you don't want to set the border color for each single header you can also drop a color on the corner of the table (as described above). Then a menu will appear with which you also can colorize all headers of a table at the same time.

Setting the background colors

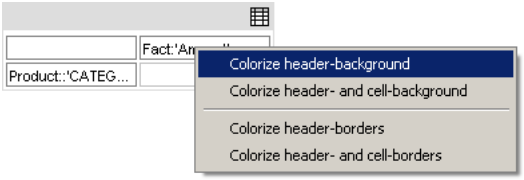
The background color for a headers is defined by its property "Background", the background color of the cells generated by the header is defined by its property "Cell Background".

Using colors for backgrounds and borders

	Amount
Desktop	524.724.708
Notebook	1.098.361.324
Pocket PC	122.257.350
Tablet PC	495.949.726

You can use drag&drop to colorize header backgrounds. Drag a color out of the Model-Explorer and drop in the center of a header. Now a popup menu appears which offers several operations for the colorization of header backgrounds.

The popup menu used to colorize headers



If you don't want to set the background color for each single header you can also drop a color on the corner of a table (as described above). Then a menu will appear with which you also can colorize all backgrounds at the same time.

Setting the font, text size and text color

With the properties Font and Cell Font you can determine the used font for headers and their cells, the font weight is determined by the properties "Font Weight" and "Cell Font Weight". Also, the font and font-weight for the corner can be changed with the properties "Corner Font" and "Corner Font Weight". To change the size of the fonts you must define them (in pixels) with the properties "Font Size" and "Cell Font Size". The color of the texts is defined by the properties Color (both for headers and the corner) and Cell Color.

A changed font, fontsize and font-color in a table

	Amount
Desktop	524.724.708
Notebook	1.098.361.324
Pocket PC	122.257.350
Tablet PC	495.949.726

You may also use the pre-defined fonts in the Model-Explorer with drag&drop to set the fonts of the headers and their cells. Simply drag the fonts out of the explorer on the headers.

Setting the text alignment

You can change the text alignment (both horizontally and vertically) for the corner, headers and for the cells generated by a header. Use the properties "Corner-Align", "Corner-VAlign", "Align", "Cell Align", "VAlign" and "Cell VAlign" to set the alignments.

If you do not define the alignment, then this default values are used for the alignment of corners, headers and cells:

- The corner is always aligned top / left
- Headers on the x axis use the alignment 'center'

- Headers on the y axis use the alignment 'left'
- Cells are always are right aligned

The fastest method to change the alignment of a header is to use its context menu. Open the context menu and use one of the four menu items "Align text...", "Align cell-text...", "Vertical align text..." and "Vertical align cell-text..." to set the alignment.

Rotating headers

In order to save place in the result of a query you can rotate header by 90 degrees. This particularly can compact headers on the x axis of a pivot table.

Rotated headers in a pivot table

	Graz	Salzburg	Bruxelles	Bern	Genève	Berlin	Frankfurt a.M.	München
Desktop	7.550.216	8.864.104	9.830.270	8.046.594	9.917.670	22.393.728	26.261.666	25.977.
Notebook	16.708.498	19.923.390	20.866.400	17.620.276	22.986.674	47.643.268	56.454.426	58.678.
Pocket PC	1.801.178	2.150.626	2.364.698	1.906.974	2.417.526	4.988.886	6.425.348	6.463.
Tablet PC	7.265.566	8.749.280	9.070.726	8.111.628	9.883.846	21.528.576	25.987.750	27.144.

There are two different ways to rotate headers: By setting the property "Rotate" of a header to "true" or by using its context menu. There you can use the menu item "Text orientation" to switch the rotation of a header on or off.

Using number formats

The number format of a header defines the representation of numbers, percentages or dates in their cells. If you set no format, then the following standard formats for the cells are used:

- If a fact is used as an iteration for a header, then the standard format of the fact will be used in its cells (presumed the fact has a standard format).
- Otherwise the standard format of the displayed data type is used.

You may also set an explicit format for a header to format a complete table row or column. You will particularly need this to format cells being computed with formulas (then no fact is used as iteration and no standard format will be available).

The format is determined with Format property of the header. This property expects a valid number- or date-format of instantOLAP. You find a description of formats with examples in the "instantOLAP Reference".

The alternative way to specify a format is to use the context menu of the header. Here you find pre-defined formats under the menu-item "Set format to...".

Visibility of headers and cells

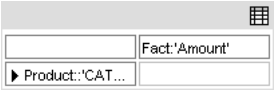
Headers and cells can be made invisible in order to let so individual cells or whole rows or columns disappear. Use the Visible property or the context menu of a header (the menu item "Visibility...") to change their visibility.

Activating drilldown

You can use the property Drilldown to allow the the hierarchical navigation in a dimension.

This property must be edited for each individual header, because headers do not offer drilldown by default. To enable the drilldown, you must set the property to "true" or use the context menu of a header (the menu item "Set drilldown...").

A header with activated drilldown (design view)



There are two different layouts for drilldown, a vertical and a encapsulated layout. With the vertical layout the unfolded headers appear under the original header, with the encapsulated their appear nested under the original header. To activate the encapsulated layout (the vertical layout is the default-setting) you must set the property Drilldown-Encapsulated to "true". When activated drilldown using the context menu of a header, you can choose between these two types.

Vertical drilldown

	Amount
Desktop	524.724.708
Notebook	1.098.361.324
Acer TravelMate 233XV Notebook	136.610.138
Apple iBook Notebook 141 M9009LL/A	187.316.766
Compaq Presario 1215 Notebook	83.587.564
HP Pavilion xt5335qv Notebook PC	126.021.524
IBM ThinkPad R40 (27232FU) Notebook	171.743.184
Sony VAIO PCG-FRV25 Notebook	102.129.680
Sony VAIO PCG-GRT170 Notebook	163.092.202
Toshiba Satellite 1135-S125 Notebook	127.860.266
Pocket PC	122.257.350
Tablet PC	495.949.726

Encapsulated drilldown

		Amount
Desktop		524.724.708
Notebook	Acer TravelMate 233XV Notebook	136.610.138
	Apple iBook Notebook 141 M9009LL/A	187.316.766
	Compaq Presario 1215 Notebook	83.587.564
	HP Pavilion xt5335qv Notebook PC	126.021.524
	IBM ThinkPad R40 (27232FU) Notebook	171.743.184
	Sony VAIO PCG-FRV25 Notebook	102.129.680
	Sony VAIO PCG-GRT170 Notebook	163.092.202
	Toshiba Satellite 1135-S125 Notebook	127.860.266
Pocket PC		122.257.350
Tablet PC		495.949.726

Apart from the layout you can also define the behavior of the drilldown with other properties: The property Drilldown-Iteration defines which dimension elements are shown when unfolding a header (usually this will be the children of the element being displayed in the header). With the property Drilldown-Prefetch you can determine if the drilldown-symbol of a header is displayed always, independent of the number of keys being displayed when unfolding a header (in some case this can speed up the report).

*When using the vertical layout for drilldown you can define a padding for the unfolded headers based on their nesting level. Use the property "Left Padding" of the header to define such a padding and e.g. set the property to the formula "DRILLLEVEL() * 10". When activating a vertical drilldown using the context menu of a header, this formula will be used automatically for this property.*

Sorting pivot tables

Often the headers of a table should be sorted by a criterion (usually a fact) - more exactly the headers generated by a header iteration should be sorted. There are different possibilities to define the assortment of a header: You can use their properties "Sort" and "Sort Descending", the context menu of a header or use the SORT function (for top 10 queries) in their Iteration property.

Using the properties Sort and Sort Descending

With the property "Sort" you can define a sort criterion for a header which is used when sorting the headers generated by a header-iteration. Make sure this formula only returns one value for each header, otherwise the system will report the error message "Ambiguous sort value" (if more than one value would exist for any element, sorting would be impossible).

Usually the headers are sorted ascending to their sorting value. If you wish to sort the header descending, you must set its property "Sort Descending" to "true".

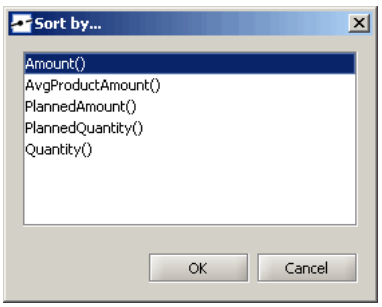
Headers sorted by the fact "Amount"

	Amount
Apple iBook Notebook 141 M9009LL/A	187.316.766
Toshiba Portege 3505 Tablet PC	177.238.760
IBM ThinkPad R40 (27232FU) Notebook	171.743.184
Compaq TC1000T Tablet PC 470045-149	170.560.474
Apple Power Mac Desktop M8839LL/A	163.731.764
Sony VAIO PCG-GRT170 Notebook	163.092.202
Acer TM C104TCi Tablet PC	148.150.492
Acer TravelMate 233XV Notebook	136.610.138
Sony VAIO PCV-RS310 Desktop	132.972.196

Using the context menu

Usually headers are sorted by a fact. You can use the context menu of a header to select a fact with the menu item "Sort...".

The sort-dialog opened by the context-menu of the header



With the menu item "Sort order..." you can choose between the ascending and descending sort order.

This menu items manipulate the properties "Sort" and "Sort Descending" mentioned above.

Top-10 queries

For top 10 queries it's not enough to specify the assortment of a header since you only want to display a part of its iteration result (e.g. the 10 products with the most turnaround). Therefore you must manipulate its iteration and use the SORT function, which allows to specify a sort criterion and a limit for the size of its result.

The simplest way to create such a limited search is to use the context menu of the header. Select a fact from the menu item "Top 10...". Then the property "Iteration" will be changed and use the SORT function. You can change the limit for the SORT function later (the default limit used for the SORT function is 10) to generate less or more than 10 iterations.

Suppressing blank rows and columns

When creating queries it is often necessary to remove blank rows or columns from the result since only a few cells may contain values. Particularly nesting headers can produce sparse filled tables.

There are different ways to eliminate blank rows and columns from reports: Using the properties Supress Rows and Supress Columns, using LOOKUPs and using subcubes:

Suppression using properties Supress Rows and Supress Columns

The simplest way to remove blank rows or columns from a pivot table is set the properties "Supress Rows" or "Supress Cols" of the table to "true". Then the empty rows or columns will be deleted after the creation and before the display of the result.

Note that also with activated Supress Rows or Supress Colums the complete pivot table will be generated in memory and therefore loaded from the databases, even if only a small part of the table will be visible in the result. This can decrease the performance of the report execution or even lead to error messages, if the size of the generated table becomes too large.

Suppression using the LOOKUP function

You should use the LOOKUP function if you want to remove more than a few blank rows or columns from a pivot-table. This function computes the keys of a dimension with present data for a certain day, customer etc. (considering the current filter of the pivot table).

When using the LOOKUP function in the iteration of a header, its iteration will be limited to those dimension elements with data before the pivot table is generated in the memory of the server and the table remains slim.

The simplest way to create an iteration using a LOOKUP function is to use the context menu of a header. Select the fact you want to perform a lookup for from the menu item "Lookup...". Then the iteration of the header will be manipulated and use the function.

Suppression using subcubes

The more complex but also more powerful way to eliminate blank rows and columns from a pivot table is to create a subcube with the property Subcube. A subcube limits all dimensions for an individual pivot table to a subset of their elements and all headers of table will be limited to this elements.

In the Subcube property you can also use the LOOKUP function mentioned above to reduce the dimensions the their part providing data. This will concern all headers of a pivot table automatically. Read the section Filters and subcubes for further information.

The context menu for pivot tables

The context menu for pivot tables offers the following operations:

- Creation of new headers
- Rotating pivot tables
- Deletion of pivot tables

You can open the context menu of pivot tables using the right mouse button on the upper, left corner of the pivot table.

The context menu for headers

The context menu for headers offers the following operations:

- Creation of new headers and aggregating headers
- Sorting and top-10 reports
- Alignment for header texts and cell contents
- Formatting of cell contents
- Deletion of headers

Editing comments

Viewing and editing the properties

Select a comment with the mouse to view and edit its properties in the Property-Editor. Since a comment is not visible (only its text) you must click onto the free space under the separator line of the comment.

Click on the title or the free space next to a comment to select it

After you successfully selected the comment, a blue border will be drawn around it and its properties will be displayed in the Property-Editor.

Creating comments

The only way to create a new comment is to use the toolbar of the Query-Editor.

The "Add comment" button

Use the button "Add comment" from the toolbar in order to append a new comment to your query. The new comment will be appended at the end of the worksheet and can be moved to the desired position afterwards.

Moving and copying comments

Comments can be moved inside a query with drag&drop. First select the comment (by clicking on the free space under its separator line) and move it with the mouse. A blue line indicate valid target positions for the comment.

Copying comments with drag&drop works exactly the same, but you must press the "CTRL" key while moving the comment. The mouse pointer will show a little plus symbol then. When dropping the comment at the target position, a copy of it will be created and the comment does not disappear from its original position.

Deleting comments

There are two different ways to delete a comment, using the keyboard or using the context menu of a comment:

Deleting comments using the the keyboard

In order to delete a comment you can use the "Delete" key. First you must select the comment with the mouse. Then press the "Delete" key to delete the comment.

The system will delete the comment without asking before. If you falsely deleted a comment you can use the "Undo" button from the toolbar to undo the deletion.

Deleting comments using their context menu

Alternatively you can use the context menu of a comment to delete it. Select the comment first and open its context menu with the right mouse button. Then use the menu item "Delete comment" to delete the comment.

Deleting a comment with its context menu

Changing the comment-text

Change the text in the worksheet

The text of a comment can be changed directly on the worksheet. Click onto the text and edit it afterwards. When you stop editing (e.g. by selecting another element of the worksheet) the changes will be stored in the comment.

Editing a comment in the worksheet

Changing the text property

Alternatively you can change the comment text by editing the property "Text". Since this property always contains a constant string, no formula is accepted for this property.

Computing the comment text

Instead of displaying a constant text you can use also use a formula for the computation of comment text. This formula must be stored in the property "Formula" of the comment.

The computed text of a comment (defined with the property Formula) is used instead of the constant text defined with the property "Text".

Deactivating the export of comments

The properties Copy To Result and Export determine whether a comment will be transferred into a Snapshot (Copy To Result) or PDF or Excel output (Export). Set the appropriate property to "false" to prevent a comment from being transferred.

The context menu for comments

Comments offer, as mentioned above, a context menu. This menu only allows to delete an existing comment.

CHAPTER 4:

Using the Config-Editor

Contents of this chapter:

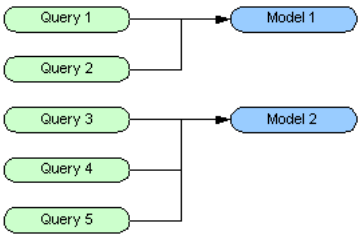
Models and configurations

Models

Reports within instantOLAP never directly access the datasources - they are based on a logical middle layer, the "models". A model is the logical abstraction of the data sources and represent their master data as "dimensions" and the values as "facts". These dimensions and facts determine the reporting possibilities for the query creator.

Several reports can access the same model, but each report can only use one model.

Queries can share models



Models are not stored physically on a instantOLAP-server, they are build dynamically whenever they are used the first time and then afterwards synchronized constantly. Also, the models only build their dimensions in ahead but don't need to store the values of the facts. These are queried dynamically during the execution of a query or taken out of "Caches" or "Stores", which can be used to accelerate queries.

Elements

A model consists of the following elements, which are described in detail in the further text of this section:

- **Dimensions** with their **Keys**, **Hierarchies** and **Attributes**
- **Facts** (which are hold in a special fact-dimension)
- **Cubes** as source for the values of the Facts. These cubes don't contain values, they only build a interface to the datasources. Then the facts are queried at the runtime of the execution from this cubes.
- **Formulas** for the calculation of facts which are not provided by any cubes
- **Caches** for storing and reusing results
- **Stores** which can hold pre-aggregated values for facts - Stores act like caches build are always filled in ahead.

Queries are only based on the dimensions and facts of a model. All other model-elements (like the cubes, formulas and caches) are invisible and only used for the internal processing of a model.

Configurations

An administrator cannot create models directly, he can only create configurations which *describe* the structure and generation (e.g. for the dimensions) of a model.

Configurations use different elements, e.g. "Key-Loaders" for the loading of dimensions or Cubes which describe how to load facts during the report execution.

At the runtime, a model will be created from this configuration and stored in the memory or on the hard-disk of the server. A model will be created automatically on its first use on basis of its configuration.

Configurations are stored as XML-files but can be edited, without the knowledge of XML, with the graphical Configuration-Editor of the Workbench. You could, however, optionally edit the XML-files for configuration manually by using the XML-editor of the Workbench.

Name of a model

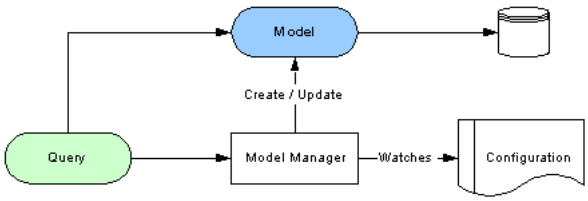
The name of a model always corresponds to the file name of its configuration, but without the mandatory extension ".config" of the configuration file - i.e. if a configuration named "demo.config" exists, it will create a model named "demo". A configuration "test/test.config" would create a model "test/test" and so on.

Because configurations can be placed in subfolders, the names of the generated models will contain the whole path of a configuration using the sign "/" as a delimiter for the folder names.

Synchronization of models

Between the configurations and the models a relationship exists, also after the first generation of a model by the configuration. Then, if a configuration is changed, this model will be rebuild automatically again before a report is executed and the query will use the new version of the model.

Creation and synchronization of models



Also, there is also an automatic and time-steered actualization of the dimensions, caches and stores, which is defined within the configuration. With this you can ensure that the keys of the dimensions (which represent the master data from your data sources) always are synchronized with contents of the data sources.

Temporary models

Beside the productive models, which are created from a configuration when using a query, there are also temporary models, which are created automatically when working on a configuration in the background. These temporary models will disappear after closing the configuration editor.

These temporary models are created while you work on a new version of a configuration and test it. They will not affect the productive models, which still exist and can be used. Only after storing a configuration the productive model is replaced by the new version.

Dimensions, Keys and Hierarchies

What are dimensions?

In multidimensional data models intended for OLAP, data are viewed as points in a multidimensional space. Each dimension has structure, described by a directed graph of categories (levels), a set of members (keys) for each level, and a child/parent relation between keys (the hierarchy).

As an example, a dimension "Time" would e.g. contain the levels "Year", "Month" and "Day". The years, months and days would be keys (members) of this categories. Between the keys a parent/child relation would exist, each day would be a child of a specific month etc.

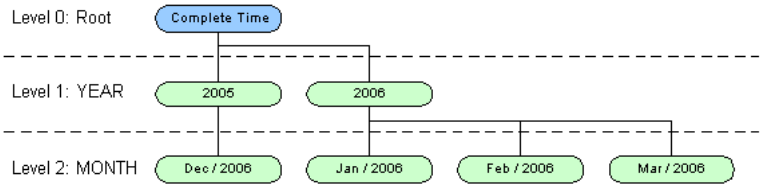
An important application of this structure is to use it to infer summarizability, that is, whether an aggregate view defined for some level can be correctly derived from a set of precomputed views defined for deeper categories.

Dimensions in instantOLAP

In instantOLAP, dimensions can contain an arbitrarily number of levels and keys. In the current version, only one hierarchy is allowed per dimension, therefore each keys only has (exactly) one parent key.

The topmost level of a instantOLAP dimension is called the "root-level" and always contains exactly (!) one key, the "root-key". This root key is mandatory and stands for the topmost aggregation possible for a dimension. Because this key is mandatory, each dimension in instantOLAP contains at least one key and one level. The root-key is the only key in a dimension without a parent key.

Levels and keys in an example time-dimension



Examples for a root key would be "Complete period" for a time-dimension or "All products" for a product dimension.

In the type-system of instantOLAP, each dimension becomes a single type in the type hierarchy and each key of a dimension is a value of this type. Each dimension-type is automatically a sub-type of the type "Key" in the type-system. Dimension-types are very important for links between dimensions (see below).

Keys (Members)

Keys are the members of a dimension in instantOLAP. Each key owns a unique ID within the dimension, a parent key (except for the root-key) and no, one or more children and a arbitrarily number of attributes.

The ID of a key is always a string (text) and *must* be unique within the dimension. Usually, the ID is defined by the loaders you create for a dimension when building the model.

Because each dimension builds its own type in the type-system of instantOLAP, each key is a value with the type of its dimension.

Levels (Categories)

In instantOLAP, keys are organized in levels within an dimension. Each key is a member of exactly one level and each level (except the root level) can have a arbitrarily number of keys as members.

Levels have both an index and a name. The index of the first level (the root level) is always 0, the following levels have the indices 1, 2 and so on. The names of the levels are defined by the creator of a configuration. Level-names must be unique inside a dimension and they must be valid names (they must not start with numbers and not contain any special characters).

Attributes and Links

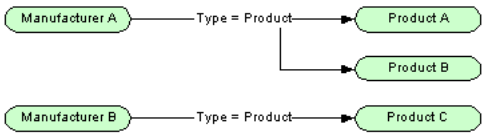
Each key can own a arbitrarily number of attributes. Attributes are additional informations for a key, e.g. a product-number for a product or an address for a customer. Each attribute has a name under which it can be addressed later in the queries.

Attributes are defined on a dimension basis, each key of a dimension can contain one or more values for any of the attributes defined for the dimension. But not every key must contain values for an attribute, therefore accessing an attribute of a key can also result to no values but won't raise an error in the type checker.

Because instantOLAP uses a strong type-system, attributes use also types. Each attribute is defined to be of a specific type and all values of this attributes may only contain values of this type, e.g. strings, integer-values or booleans.

A special case are attributes with a dimension-type, they may only contain keys from other dimensions as values. Because this attributes refer to elements from other dimensions, they act like "links" between dimensions. E.g. you could create a link between the two dimensions "Product" and "Manufacturer" with an attribute and use this link later in your queries.

Links between dimension keys



Link have directions, they point from one key to another. Also, bidirectional links are possible which then point from one key to another and backwards. In this case, each key has an own attribute (which can have different names) linking the other key.

Unique attributes

Attributes are generally used in two different ways: In queries, e.g. for displaying texts, or for the binding to cubes inside the configuration in order to assign the data from fact-tables to keys.

To bind keys with cubes often makes in neccesary to store the database-IDs within the keys, because in the cubes you'll have to map database-columns from fact-tables to

your dimension, respective keys. This columns often don't contain the name of a key (e.g. the product-name) but a technical ID defined inside the database.

Also, different databases or even different tables of a database can use different Ids, in this case your keys would need to contain multiple IDs, each for another table or database.

This IDs stored as attributes inside a key must by *unique*, otherwise the data from the fact-table couldn't be assigned to unique keys. Attributes are only unique if the creator of a configuration defines them as unqique attributes and if no value for this attribute is equal to the value of any other key inside the dimension.

Default text-attributes

Whenever Keys are used in queries, their IDs are displayed as text by default inside the table headers, selectors or any other place wherever they are used.

If the IDs are too technical (e.g. if you use database IDs or self generated IDs in order to make then unique inside the dimension), you can define a standard attribute from the dimension to be displayed instead of the IDs. This allows to display human-readable texts though the IDs are technical and unique. Also, this standard-text attribute does not have to be unique, e.g. two product could have the same text though they own different IDs.

This text attribute is called the "Default Text-Attribute" and can be defined for each dimension. If a key does not contain any value for the default text-attribute, its ID will be displayed in queries (like there was no default text-attribute).

Description and visibility

Beside the keys, dimension contain some general properties like the name, a description and their visibility. The Name of a dimension is important for the later usage of the dimension in cubes and queries, because dimension are always referred by their name.

Like all elements in instantOLAP, dimensions must have valid names with no digit at the beginning an no special-characters anywhere.

The description of a dimension is used for the administrator and power-users, who see the dimension in their configuration- or query-editor. The description is not mandatory and can be left empty.

The visibility of a dimension determines wheter it will be visible in the query-editor or any other end-user tool. Invisible dimension are hidden in this tools. Usually, technical or help-dimension are invisible, like the Line-Dimension which is contained in each model and used for loading and displaying detail-data from databases, but not visible for the end-user.

Loading dimensions

In instantOLAP, dimensions are build *once* at the first startup-time of a model and then used for the complete lifecycle of the model. After a dimension was loaded, it will be used for queries without beeing reloaded for each query excecution. This is mainly done for performance reasons, because building a complex dimension with all its keys

and attributes (which could also be assembled from multiple dimensions) can take some CPU time and should not be done each time a query is executed.

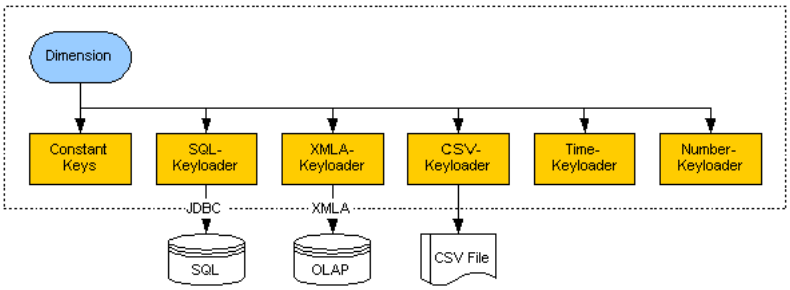
Key-Loaders

Dimensions are not simply mapped to columns of a single database - you can use and combine the loader modules for dimensions (the so-called "Key-Loaders") to build complex structures and hierarchies in your dimension and even combine data from multiple databases or other types of datasources in one single dimension.

There are several types of key-loaders which can be used to assemble a dimension:

- **Constant Key-Loaders,** which add a single constant key to your dimension and don't need any datasource,
- **SQL Key-Loaders,** which load keys from a JDBC-datasource,
- **CSV Key-Loaders,** which load keys from a CSV-datasource,
- **XMLA Key-Loader,** which allows to transfer keys from other OLAP-databases (which support the XMLA-standard as interface),
- **Time Key-Loaders,** which allow to build keys from the internal calendar of instantOLAP,
- **Number Key-Loaders,** with which you can generate keys from number-ranges.

Dimension use Key-Loaders to collect their keys from different datasources



All Key-Loaders offer a large number of properties which you can use to define how keys are loaded and which attributes they will contain.

Time dimensions

Time dimension are always a special case in OLAP-system. Most systems own a internal time dimension which can automatically be used in queries without having to build it before. This is different in instantOLAP and you'll have to define your own time-dimension. There is a special Key-Loader (the Time Key-Loader) which allows to access the internal calendar of instantOLAP and to generate keys from it in a very flexibel way.

The reason you'll have to define your own time dimension is because you're much more flexible in the definition of your time dimension and can also use other datasources than the calendar to build it. E.g you are able to import time dimensions from databases and only define keys for days with available data. An automatic time dimension would not able to do this for you.

Synchronisation of dimensions

Because the data inside the databases used for the dimensions can change, you can define an update process (a time pattern) for your dimensions which defines how often dimension will be reloaded. After the reload of a dimension, the new data will be available and visible the queries.

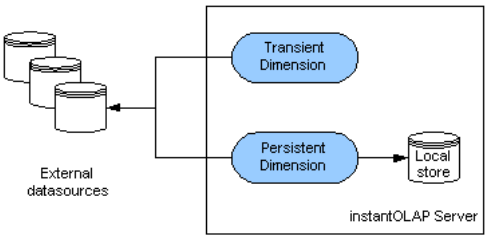
Depending on the type of data you store in your dimension, it will be neccesary to reload the data more or less often. A calendar would e.g not change more than once a day, so it would be enough to rebuild a time-dimension once every night. Other dimensions like products could change more often and should then be reloaded more frequently, e.g. once per hour.

Storage of dimensions

Persistent storage

When the keys of a dimension are created they will be stored locally on the hard disc of the instantOLAP server.

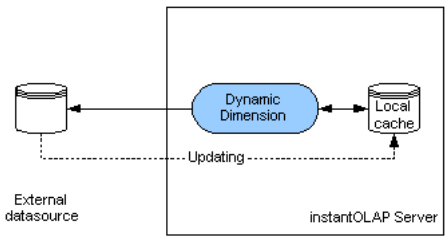
Dimension load their keys once and store the in memory or in a local store



Dynamic dimensions

The other storage-model for dimensions is the "dynamic" storage. In this case, keys will be queried from the datasource at the first time they are used and then added to an internal cache.

Dynamic dimension load and cache keys at the query-execution



Because this is like a "direct" mapping to a database, dynamic dimension can't be a complex as loaded dimensions. You can use only data from a single database in them and the hierarchy must be very simple.

Also (because the system always load the children of one key together), this hierarchy of the dimension should be rather deep than flat and no key should have more than thousand children.

You can use dynamic dimension to import data from databases with a very large number of keys without loading it in ahead.

Access control for dimensions

The access to dimensions can be limited for single users or user groups. instantOLAP contains a powerful access control, which allows to define exactly which user or group can access which keys from which dimension. Every user will only see the keys in the queries to which he has access. Also, all other elements of queries like selectors, expression editors etc. will only display this keys.

The access control for dimensions in instantOLAP is rule based. The administrator may define a set of rules (boolean expressions) for each dimension. These expressions are executed for each key a user wants to access. Only if one of the rules returns to "true" the user can view or edit the key. The rules allow both to control the viewing or editing of dimensions.

To "edit" a dimension means to edit a fact for a coordinate which contains a key of this dimension. The user can only edit the fact if he has a write access to *all* keys of the coordinate.

The expression language of instantOLAP contains a number of functions to access information about the current user executing a query. E.g. you can access the current user name or check his membership in a role.

If no access rules are defined for a dimension, then all users may see and edit all keys of this dimension.

Facts

What are facts?

In OLAP systems, Facts represent the Business Measures depending on the dimensions. Typical facts are "Turnaround" or "Price". All of these facts depend on at least two dimensions - e.g. the "Turnaround" would depend on the Period, the Product, the Region and so on.

Note that a fact, if it only depends on one dimension, it could also be treated as an attribute of that dimension. E.g. would the "Name" of a person be rather an attribute of the "Person" dimension than a fact. You should always consider to design such facts as an attribute.

Definition of facts

Before you can use facts in queries or create Cubes which deliver data for them, you must **define** the facts in your model.

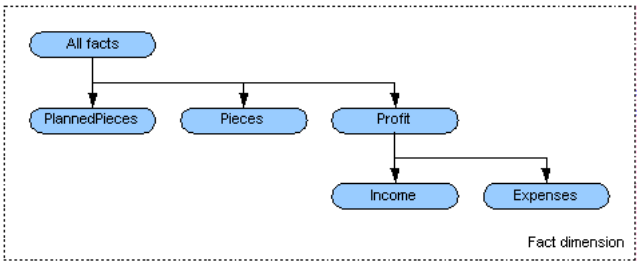
The fact-dimension

In instantOLAP all facts are *defined* in a special dimension, the so-called "Fact dimension" which must exists in every model. Each fact is defined as a key in this dimension, together with some addtional properties like the type of the fact, its format, its unit or user-defined attributes.

The name of the fact dimension can be freely defined. Usually its name is "Fact" and this is the default name for the dimension when creating a new Model with the Workbench.

Because facts are members of a dimension, you can define a hierarchy for them and also use this hierarchy in your queries. And because facts are simply keys defined in a dimension, you can make them selectable in selectors, use them in table headers, give them attributes and use them in your queries and so on.

A example fact-dimension



Types

The type of a fact determines which kind of data will be returned whenever a fact is used in expressions or queries. Usually fact return "Numbers" (Double or Integer values), but any other kind is also possible.

Formats

A fact can have a format which determines how numbers or dates will be formatted in the display of a query. If no format is specified, the standard format of Java will be used.

For other types than numbers and dates no format will be used and any defined format will be ignored.

Units

The unit of a fact is a small text beeing displayed in table headers to show the unit of a business fact. Typical units are "USD", "EUR", "Pieces" or "Days".

Accessing facts

After a fact was defined you can use it in your queries. There are two ways to address facts and to display their values in queries: You can place facts in table headers and let the system automatically display the values or you can use expressions to determine what to be displayed in tables (or at any other place).

In expressions, facts can be accessed by the special funtion CUBE, which allows to read to content of cubes. Or you can use the *Fact-Functions* generated by the model, which names are equal to the names of the facts (unless the name of a fact is not convertable to a well-defined function name!). Using this functions is simpler and easier to read than using the CUBE function.

Where the fact data comes from

- **Cubes** deliver and aggregate data which is available in datasources, e.g. in SQL-databases or in CSV-files. There are several types of Cubes, each for a

different type of datasource (SQL-Cubes, XMLA-Cubes, SQL-Cubes and CSV-Cubes).

- **Formulas** calculate facts which are not available in any regular cube (and therefore in no datasource) on basis of other coordinates (e.g the Amount for a quarter could be calculated on basis of three months or the Profit could be calculated on basis of the Income and Expenses).
- **Caches** store facts which are already loaded from a cube or calculated and make them available for following query-executions.
- **Stores** contain facts which are pre-loaded out of other cubes and then made available for queries. Stores are comparable to cubes of "classic" OLAP system, which usually do don't query datasources on the fly and only contain pre-loaded data.

Formulas, Caches and Stores also act like Cubes though they are treated different in the configuration editor. Read the following sections to learn more about the different types of cubes.

Cubes

What are cubes?

OLAP's main representation of facts is via what is known as a *Cube*. A Cube is a multidimensional representation of facts that can be accessed quickly to provide specific information for any combination of dimension-keys (a combination of keys from different dimensions is called a *coordinate* in instantOLAP).

For example a set of customers can be grouped by city, by district or by country. So with 50 cities, 8 districts and two countries there are three hierarchical levels with 60 members. These customers can be considered in relation to products. If there are 250 products with 20 categories, three families and three departments then there are 276 product members. With just these two dimensions there are 16.560 (276 * 60) possible coordinates for which a cube can deliver data.

As the data considered increases the number of aggregations can quickly total tens of millions or more.

How data is queried from cubes

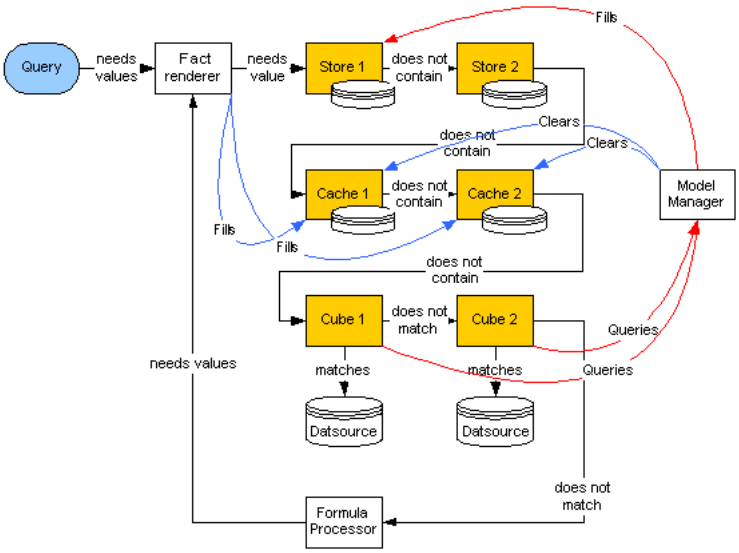
instantOLAP is a "Multicube OLAP" system, therefore more than one Cube is possible per model. Each Cube can deliver a different fact or the same fact for a different combination of dimensions or the same fact for the same dimensions but with a different filter. Therefore you can exactly determine which fact has to be loaded from which Cube(s).

In most OLAP systems, cubes are equivalent to facts because each cube delivers exactly one fact and each fact is delivered by one cube. This is different in instantOLAP because a single cube can deliver multiple facts or a fact can be delivered by multiple cubes.

The combination of Datasource Cubes in instantOLAP allows to gather and combine data from multiple datasources. Also, additional Caches and Stores (which are also Cubes) allow to reduce the response-time for queries dramatically.

Because you can define a large number of Cubes for an instantOLAP model you must understand how the system determines which fact is loaded from which Cubes. For each fact needed by a query, the following steps are performed when the query is executed:

Stores, Caches and Cubes are queried until the first returns a value for the desired coordinate



The "Fact Renderer" also adds the new values, which come from the Datasource Cubes, to the Caches of the Model. The Model Manager fills the Stores and cleans the Caches when they become too large or their data becomes too old.

Datasource Cube-Types

Where the data comes from depends on the type of the Cubes you define in your model configuration. There are three different types which can be used to connect data sources:

- **SQL-Cubes** query facts out of SQL-databases (using the JDBC interface),
- **XMLA-Cubes** query facts out of other OLAP-servers (using the XMLA-standard),
- **CSV-Cubes** read facts out of CSV-files.

All Cubes of this types query the facts *in the moment the query is executed*, e.g. a SQL-Cube would create SQL-statements and send them to a SQL-server, the CSV-cube would open and parse its CSV-file and so on.

SQL-Cubes (ROLAP)

A SQL-Cube is created from a star schema of tables of one database. At the centre is the fact table which lists the core facts which make up the query. Numerous dimension tables are linked to the fact tables. These tables indicate how the aggregations of relational data can be analysed. The number of possible aggregations is determined by every possible manner in which the original data can be hierarchically linked.

A fact table may contain one or more Facts. Usually you create one fact table per fact. For example if you want to analyze the sales numbers and also advertising spending, they are two separate facts. So you will create two separate fact tables, one for sales data and one for advertising cost data. On the other hand if you want to track the sales

tax in addition to the sales number, you simply create one more fact column in the Sales fact table called Tax.

Formulas

What are formulas?

Formulas are used to calculate facts which are not available in any other Cube. This can be e.g. complex calculations which cannot be perfomed by a database, facts which have to be computed with data from multiple databases etc.

Formulas are only used if no other Cube is able to deliver the fact, therefore formulas are the last step in the Cube chain described in the Cubes section of this documentation.

Definition of formulas

If you want to create a new formula, there a serveral things you must define for it:

- The **fact** for which the formula should apply
- The **expression** which will be used to compute the fact
- An optional **match expression** which defines for which coordinates a formula should apply (e.g. you may only want a formula to calculate values for years, but not for months, days etc.)
- A option **priority** which controls which formulas are calculated first

The calculated fact

The fact is the first thing you must define when creating a new formula. Only this fact is calculated by the formula. If you want to calculate other facts you must add new formulas (for the other facts) to your model.

Formula-Expressions

The expression describes how the fact should be calculated. Because instantOLAP is a multidimensional system, the expressions are also multidimensional and can apply to a large number of coordinates. In the expressions you can use other facts or navigate within the dimension, as an example you could calculate a "Year To Date" value as the sum of another fact from the past months in the current year.

The Match-Expression

The Match-Expression for a formula is optional. It defines, for which subset of the model the formula should be used to calculate a fact. If no Match-Expression is defined, the formula will apply to all coordinates of the model (for the given fact).

Match-Expression can be used to calculate different parts of the model with different formulas. Match expressions are boolean expressions. If the expression result to "true" for a coordinate to be calculated, the formula will apply. Otherwise the following formulas will be used to calculate the value (if any matches).

Match Expressions are usefull to perform complex calculations. As an example, a YearToDate of the turnaround could be calculated with the following two formulas:

```
YearToDate (for the first month in a year) = Turnaround of the
current month

YearToDate = YearToDate of the previous month + Turnaround of
the current month
```

Then the first formula had the Match Expression "For the first month in a year", or written in the instantOLAP syntax:

```
HASPOSITION( MONTH, 0 ).
```

Order of the expressions

Because you can use Match Expression to calculate different subsets of the model with different expressions, there can be more than one formula for a single fact in your model.

Therefore, the order of the formulas inside your model is important. Whenever the system looks for a formula in order to calculate a value, the *first* formula matching the current coordinate is used.

Formulas can also be recursive if a formula need other values to be computed. This other value could also be calculated with other formulas or even with the same formula. E.g. the second formula in the upper example would call itself until the first formula applies because of its Match-Expression.

If it gets to complex to control the order of the formulas, the option priority of a formula can be used to control the exact order of execution for your formula. The system will then calculate all formuals with the highest priority, then the formulas with the second highest and so on.

Caches

What Caches do

Caches are used to speed up the execution of queries by storing the results and re-using them for future queries as a whole ore only parts of it. Caches do not store whole results but the single values for each fact and coordinate which has been used by a prior query. If a new query only needs some of the coordiantes, it will then be delivered with values out of the Cache. If only parts of the coordinates needed by a query are available in Caches, the system will use this coordinates and query all others from the Cubes.

Caches store their values on the hard disc of the server and can become very large without using significant memory of the server.

Definition of Caches

When creating a new Cache you can define different properties for it:

- The unuique **name** of the Cache
- The **maximum age** of all values stored in the Cache
- An optional **Match Expression** which defines which coordinates are stored in the Cache

The maximum age of a cache

After values are stored in Caches they can change in the Cubes which they originally come from. Therefore it is very important to define a maximum age for the entries of the cache. Any entry beeing older than this maximum age will be deleted from Cache (and then re-queried with the next query execution) whenever the Cache is checked for outaged entries.

How of a Cache is checked of outaged entries can also be controlled. The systems allows to define a cron pattern which exactly describes how often a Cache should be cached. Together with the maximum age, this is a very flexible way to control the contents of your Caches. E.g. you could check the Cache every 5 minutes for entries beeing older than 2 days or check it every night at 00:00 for entries older than 0 seconds (and therefore always clear it at midnight).

The Match Expression of a cache

By default a cache will store all results of all queries. But with its Match-Expression you can define for a cache, which content has to be store and which not.

This allows to set up different stores (with a different maximum age, size or cron pattern) and store different contents in them. E.g. you could set up a cache without a maximum age for facts with never will change in the datasource and another with a short maximum age for facts which change often.

The Match-Expression must be a boolean expression and can use all functions of instantOLAP to check a coordinate (which should be stored) for compliance. E.g. an expression "`NOT( EXISTS( NEXT( Period ) ) )`" will cache all data execpt for the current year, month or day.

Stores

What stores are

Stores are Cubes which allow to collect and pre-aggregate the content of other Cubes in order to improve the performance of your model and the queries using the models.

Stores work like Cubes of "classical" OLAP tools because you can use them to use data from a local and multidimensional store instead of querying the source databases directly and with each execution of a query.

Stores are comparable to the Caches of instantOLAP. But Caches only store data which was used by a query and then will be available for the following executions of queries. Stores are filled and pre-aggregated in a ahead and contain much more data, even if no query will ever use the data.

Content of a store

Stores do not need to contain all data of all Cubes, they also can span only a subset of all dimensions. Then all data contained in this subset will be taken from the Store and all other data will still be queried from the other Cubes, Caches (or other Stores).

Loading a subset of the facts into Stores allows to keep Stores small and efficient. Often there is only a small amount of data which is frequently used by the users and causes performance problems, especially the highly aggregated data for the uppermost elements of the dimensions. All other data will be queried less frequently and will also be faster to query from the origin datasources, because they have a lower aggregation and therefore need less data from the datasources.

The subset of the dimensions defined for a Store is called a "Subcube". A Subcube defines a subset of keys of each dimension. There must be at least one key of each dimension in this subset. If a dimension should not be part of a Subcube, you can use the root-key of a dimension (which stands for the topmost aggregation like the "Complete period" or "All products).

Formulas and stores

When the data from all Cubes is used to build a Store you can decide whether you want the system to calculate all formulas which depend on the Store's content or not. Calculating formulas can be a time consuming task because, depending on the formulas in your model, a large number of formulas could be calculating for a single value in your store and the size of the Store could be much larger than without the calculated values.

If you don't want the store to the calculated values, the system will calculate the formulas whenever you execute a query. This results into a lower performance but saves time when building a store. Also, using Caches would avoid the multiple calculation of formulas even if they are not part of the Store.

Synchronization of stores

Because Stores contain snapshots of the data contained in the Cubes, they have to be updated frequently. This is done by the model synchronization which also rebuilds the Stores using a user defined time pattern.

Whenever you create a Store you'll have to decide how often it has to be rebuild. The cron-pattern of a store contains a Unix-like cron pattern which allows to freely defined this.

Until a Store is completely rebuild, the older version of the Store is available for the queries. If no older version is available, e.g. if this is the first time a Store was build or you deleted the older content manually, the Store will not be used until it is finished and all queries are using the Cubes or other Stores.

Importing other models

Model imports are used to make models easier to administrate because often used dimensions can be defined once in one or more master models and then be used in multiple other models. Therefore you need to define frequently used dimensions (like the time-dimension) only once and can use it in a large number of other models.

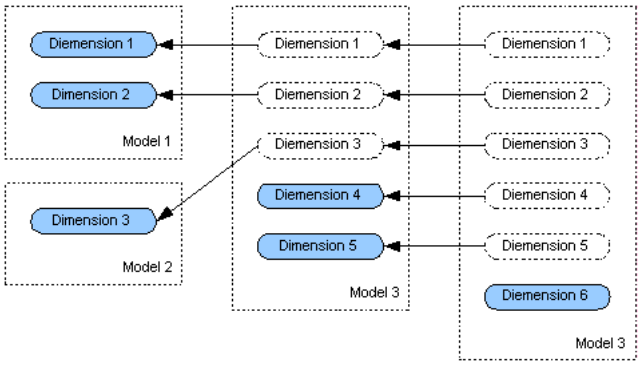
Also, using imported dimension can save system resources, because imported dimension allow to share the *physical* dimension between models. Imported dimension don't have to be synchronized by the importing models and they do not use memory or disc space in the importing model.

Imported dimensions

Each model can import an arbitrarily number of other models. From each imported model all dimensions except the fact dimension will imported and be available for Cubes and queries using the dimension.

If a imported model contains a dimension which is already defined in the importing model or a previously imported model, it will not be added to the importing model.

Imported dimensions



Importing models can be nested. If you import a model which already imports other models, your model will contain all dimensions being defined in the imported model or the models imported by this.

*Importing models will **only** import their dimensions together with their access rules. Facts, Cubes, Caches, Formulas and so on will not be transferred into the target models.*

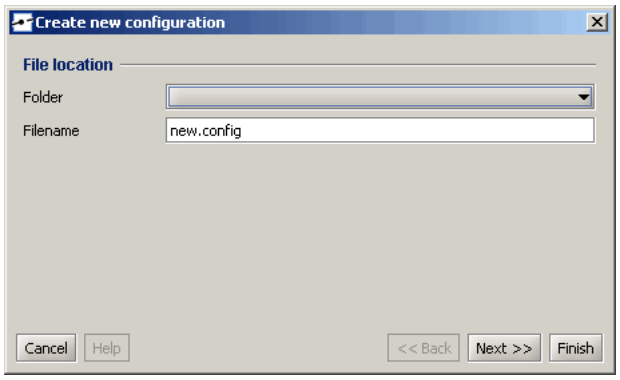
Starting the Config-Editor

Creating new configurations

There are two different ways create a new configuration within the Workbench: By using the main menu, the tool bar of the Workbench or by using the context menu of a directory within the Repository-Explorer.

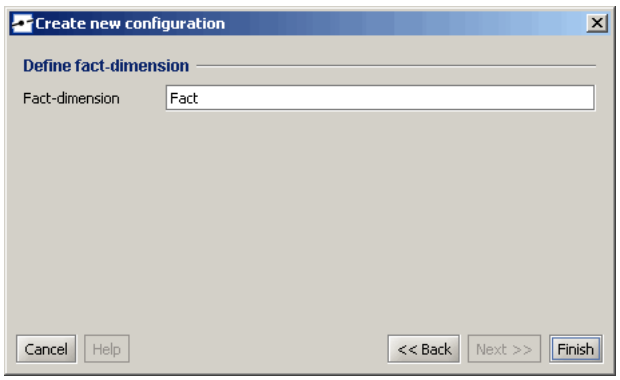
Using the main menu or the tool bar

Use the menu item "Configuration/New Configuration" to create a new configuration. Then a wizard opens where you can define serveral settings for your new configuration.



On the first page of the wizard you must define the filename of the new configuration and the folder where it will located. The filename of a configuration is very important because it will also determine the name of the model created by this configuration.

After you set the folder and the filename press the "Next >>" button to reach the next page of the wizard.



In the second and last page of the wizard you can define the name of the fact-dimension which will be automatically created by this wizard. Depending on your language there will be a default-name for this dimension but you can change the name if you want. The name of the fact-dimension must be a regular name and must not contains whitespaces or special characters.

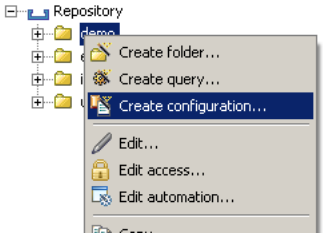
After you defined the name of the fact-dimension press the button "Finish" to create the new configuration. Then the configuration editor will open, showing an empty configuration file.

After you created a new configuration, it will not be written to its folder until you save it.

Using the context menu of a folder

Using the context-menu of a folder to create a configuration is similar to use the Workbench-Menu or toolbar but the folder in the wizard will be pre-filled with the folder on which you opened the context-menu.

Open the context-menu of a folder and use the menu item "Create configuration..." to open the configuration wizard.



Opening exisiting configurations

To open and edit an existing configuration you can double-click onto it within the Repository-Explorer or drag the file onto the empty space of your Workbench desktop.

Then the editor will open, showing the existing configuration with all its datasources, dimensions and other elements.