

Training Guide for the first steps

Apparo Fast Edit

for

Qlik Sense



Table of Contents

1	<i>Introduction</i>	8
1.1	What is Apparo Fast Edit	8
1.2	What were the driving forces behind the development?	8
1.3	Resulting cornerstones of Apparo Fast Edit	8
1.4	Target groups and goal of this tutorial	9
1.5	Technical requirements	9
1.6	Basic Handling	9
2	<i>Main Concepts</i>	10
2.1	Databases and Users	10
2.2	Usage of Apparo Fast Edit	11
2.3	Business Cases (BC)	11
2.4	Business Case types	11
3	<i>Clients</i>	13
3.1	Creating a new client	14
3.1.1	Client head	14
3.1.2	Client security groups	15
3.1.3	General settings	15
3.1.4	Languages	16
3.1.5	Default formats	16
3.1.6	Access rights	17
3.1.7	Automatic table/column creation	18
3.1.8	Excel export formats	19
3.1.9	Business Log	19
3.2	Standard colours of a client	20
3.3	Using own CSS definitions	22
3.3.1	Theme.css	22
3.3.2	Favicon.ico	23
4	<i>The Apparo Designer</i>	24
4.1	Introduction	24
4.2	Start screen with a list of all Business Cases	24
4.3	Description and normal mode	25
4.4	Managing of Business Cases	26
4.5	Designer Folders	26
4.6	Connections	27
4.6.1	Buttons	27
4.6.2	Creating a new database connection	28
5	<i>Creating a Table Business Case (TBC)</i>	29
5.1	Areas of a Table Business Case	29
5.2	Create a new Business Case	30
5.3	Business Case features	31

5.4	Features areas and features	32
5.5	Edit view of the Business Case	33
5.6	Business Case Settings	34
5.7	Widgets	35
5.7.1	Edit view	35
5.8	Widget types and areas	36
5.9	Widgets in the Edit Area	37
5.10	Special functions in the widget settings	38
5.11	Reading and Writing Expressions	38
5.12	Conditional options	39
5.13	Widget settings for the example ,Input field'	41
5.13.1	Mapping & Other	41
5.13.2	Features	43
5.13.3	Visual	48
5.13.4	Help texts	49
6	<i>Single Business Cases (SBC)</i>	50
6.1	Structure of the SBC	50
6.2	Arrangement of the widgets in the SBC	51
6.2.1	Columns - a users view	52
6.3	Visual	53
7	<i>Combining Business Case</i>	55
7.1	Business Case Set	55
7.2	Master-detail relationship	56
7.3	Primary Keys	57
8	<i>Linking Apparo Fast Edit and Qlik Sense together</i>	58
8.1	Usage of the Apparo Designer & Apparo BC Button extensions	58
8.2	Embedding Business Case into the Qlik Sense App	59
8.3	Business Case and the standard Qlik Sense Table	61
8.4	Using the Apparo Table extension	64
9	<i>Creating database table and columns with the Designer</i>	65
9.1	Activation of the feature in the client settings	65
9.2	Create the required database connections	65
9.3	Create a database connection to create tables / columns (DDL)	66
9.4	Creating and linking the DB connection type ,read / write only (DML) '	67
9.5	Create a new database table	67
10	<i>E-mail Business Cases (EBC)</i>	70
10.1	Creating an EBC	71
10.2	Header and Footer	71
10.3	E-mail properties	72

10.4	E-Mail body	73
10.5	E-mail attachments	74
10.6	Button titles	74
11	<i>E-Mail Import Business Case (EIBC)</i>	75
11.1	Creating a new Business Case of Type 'Email Import'	77
11.1.1	New Business Case - Main Settings	78
11.2	Overview of all possible settings	79
11.3	Main Settings.....	79
11.4	Importing Groups.....	80
11.5	Importing group settings	80
11.5.1	Main group settings	80
11.5.2	Business Cases.....	81
11.5.3	Add new attachment	81
11.5.4	Advanced Excel document data constraints	82
11.5.5	Email texts	83
11.5.6	Security.....	84
11.6	eMails	85
11.7	Logging.....	86
11.8	Variables	87
12	<i>Business Case Sets (Set)</i>	88
12.1	Selection and positioning of business cases in the set (Set)	88
12.2	Colors	89
12.3	Tab Widths	89
12.4	Global Set filters	90
12.5	Link into Portal (Standalone version only).....	91
13	<i>Linking BC as Master-Detail connection</i>	92
13.1	Master-Detail basics	92
14	<i>Optimizing Business Cases</i>	94
14.1	Securing data quality	94
14.1.1	Data output format.....	94
14.1.2	Data Quality check	97
14.1.3	Data row validation	98
14.2	Tracking of data changes.....	99
14.2.1	Auditing of data changes	99
14.2.2	Data History.....	101
15	<i>Calling external scripts / procedures / functions</i>	106
15.1	Own action buttons	106
15.2	Executable button.....	106
15.2.1	General Settings.....	106
15.2.2	Dialog visual.....	107
15.2.3	Information texts.....	107
15.2.4	Features.....	108
15.3	Database procedure button	109

15.4	URL buttons.....	111
15.5	E-mail Buttons	111
15.6	Pre/Post execution	112
16	Action Business Cases (ABC)	114
16.1	Example for calling a database function using AJAX	115
16.1.1	Create a new Action Business Case.....	115
16.1.2	Define a report variable	116
16.1.3	Define the actions	118
16.1.4	Calling the Action BC	119
17	Filter data output.....	120
17.1	Filter widgets.....	121
17.2	Filtering prompt page settings.....	121
17.3	Combine Widgets with AND/OR.....	121
18	Variables	122
18.1	Definition	122
18.2	Use of variables in the Designer	123
18.2.1	Variables in lookup definitions.....	123
18.2.2	Variables in labels, hint texts, the header and footer	123
18.2.3	Variables in filter definitions:.....	123
18.2.4	Variables used in data validations	124
18.2.5	Variables in variables	125
18.3	Internal Variables	126
18.4	Report Variables	127
18.5	SQL Variables.....	128
18.6	Script Variables.....	130
18.7	Widget reference variables	131
18.8	Environmental variables.....	132
18.9	Debugging variables.....	133
18.9.1	Definition.....	133
18.9.2	Variable output for debugging purposes.....	133
18.9.3	Debugging script variables	134
18.9.4	Debugging SQL variables.....	136
19	Data hierarchies	137
19.1	What do we need?	137
19.2	Expected result	138
19.3	Implementation.....	139
19.3.1	Creation of the Business Cases	139
19.3.2	Customizing of the Business Case.....	140
20	Excel Import	142
20.1	General Excel Import.....	143
20.2	Import strategy.....	144
20.3	Manual Import using copy & paste	146

20.4	Excel File Import	147
20.5	Automatic Excel file import	149
21	<i>Conditional formatting</i>	151
21.1	Background color	151
21.1.1	Widget background.....	151
21.2	Row background color	152
21.3	Conditional formatting using HTML & JavaScript.....	153
21.3.1	Colored text for Status Controlling.....	153
21.3.2	Arrows for the ,Trend‘ widget.....	154
22	<i>Movies</i>	155
23	<i>Support</i>	156

1 Introduction

1.1 What is Apparo Fast Edit

- Apparo Fast Edit is bridging the gap between the IT and the business departments, making it easy for managers, accountants etc. **to create forms, data entry masks and even complex workflows for quick data input** without special knowledge in a programming language.
- Apparo Fast Edit is a **high-performance data entry software**. With it, you can enter, edit, delete data and also easily import data from Excel files in different ways (via email, via copy&paste or as file import).
- With Apparo Fast Edit, you can
 - a) **build stand-alone data entry solutions**, or
 - b) **extend your BI reporting system (e.g. IBM Cognos BI) or your enterprise portal**

1.2 What were the driving forces behind the development?

- Reporting and Data Entry should be connected.
- Business requirements should be implemented very flexibly.
- Business departments want fast solutions, without waiting for IT development.
- System should be integrated completely in BI or enterprise portal systems.
- Software should be able to expand and grow depending on needs.
- Software should be easy to use, also by business departments.

1.3 Resulting cornerstones of Apparo Fast Edit

- Forms definable in the Apparo Designer, without programming knowledge.
- Powerful variables and scripts or database procedures to maximize the flexibility
- Security based on existing authorization system (e.g. LDAP, Active Directory)
- Data quality must be secured by possible input validation of data types and values.
- Excel and CSV import possible (manual or automatic), also handling of file attachments.

Due to its many features Apparo Fast Edit can be adapted to fit almost any need.
The online help in the Apparo Designer helps to start immediately by giving explanations and hints on all features and settings.

There are two sides of Apparo Fast Edit

- the **Backend** (Apparo Designer), on which forms, masks and Workflows are set up as so-called *Business Cases*

- and the **Frontend**, on which these forms are filled with data.

1.4 Target groups and goal of this tutorial

- The target groups of Apparo Fast Edit are persons who create and edit data entry forms, for example in business units. No programming knowledge is necessary!
- The focus of this tutorial is to explain the basic thoughts and models that are necessary to translate a problem into a Fast Edit Business Case, and to highlight important features.
- After completing this tutorial, the user should be able to create simple and also complex Business Cases so that data can be entered easily and quickly into reports, dashboards, mask and forms into any relational database.

This tutorial is intended to help you through the first steps with Fast Edit. It also touches many more advanced topics to give you an overview over the vast possibilities.

You can find extensive information about the Fast Edit features in the document “User Guide”, in the doc und <https://doc.apparo.solutions> and in the Training menu of the Apparo Designer (Movies & more).

1.5 Technical requirements

1. Apparo Fast Edit has to be installed and started.
2. A web browser has to be available, as Apparo Fast Edit is a web-based software.
3. The BI system has to be installed and ready.
4. As the language of the Fast Edit interface and the tutorial is English, basic English knowledge is necessary.
5. No BI Installation is needed for the standalone version.

1.6 Basic Handling

- *Fields with an *(asterisk) have to be filled in.*

Since Apparo Fast Edit works with databases and tables, buttons like “OK”, “Cancel” etc. have a defined effect in terms of database transactions.

- a) **“OK” (without a “Close” available):** All changes are stored in the database and the Business Case is closed.
- b) **“OK” (with a “Close” available):** All changes are stored in the database.
- c) **“Close”:** All changes are stored in the database and the Business Case is closed.
- d) **“Cancel”:** Rolls back database transaction(s). No changes are saved.

Note:

- *“OK” and “Close” are sending a COMMIT command.*
- *All button texts can be changed in the Apparo Designer on the “Button Titles” register, but this will not change their effect.*
- *If the feature ‘auto commit’ is enabled all changes are stored immediately.*

2 Main Concepts

In the following section, you will find a general overview of the main concepts and the technological background of Apparo Fast Edit.

For more technical information on the subject, please refer to the “System Architecture” document.

2.1 Databases and Users

Example: IBM Cognos BI & Apparo Fast Edit

Apparo Fast Edit bridges the gap between the database, the form designer and the user.

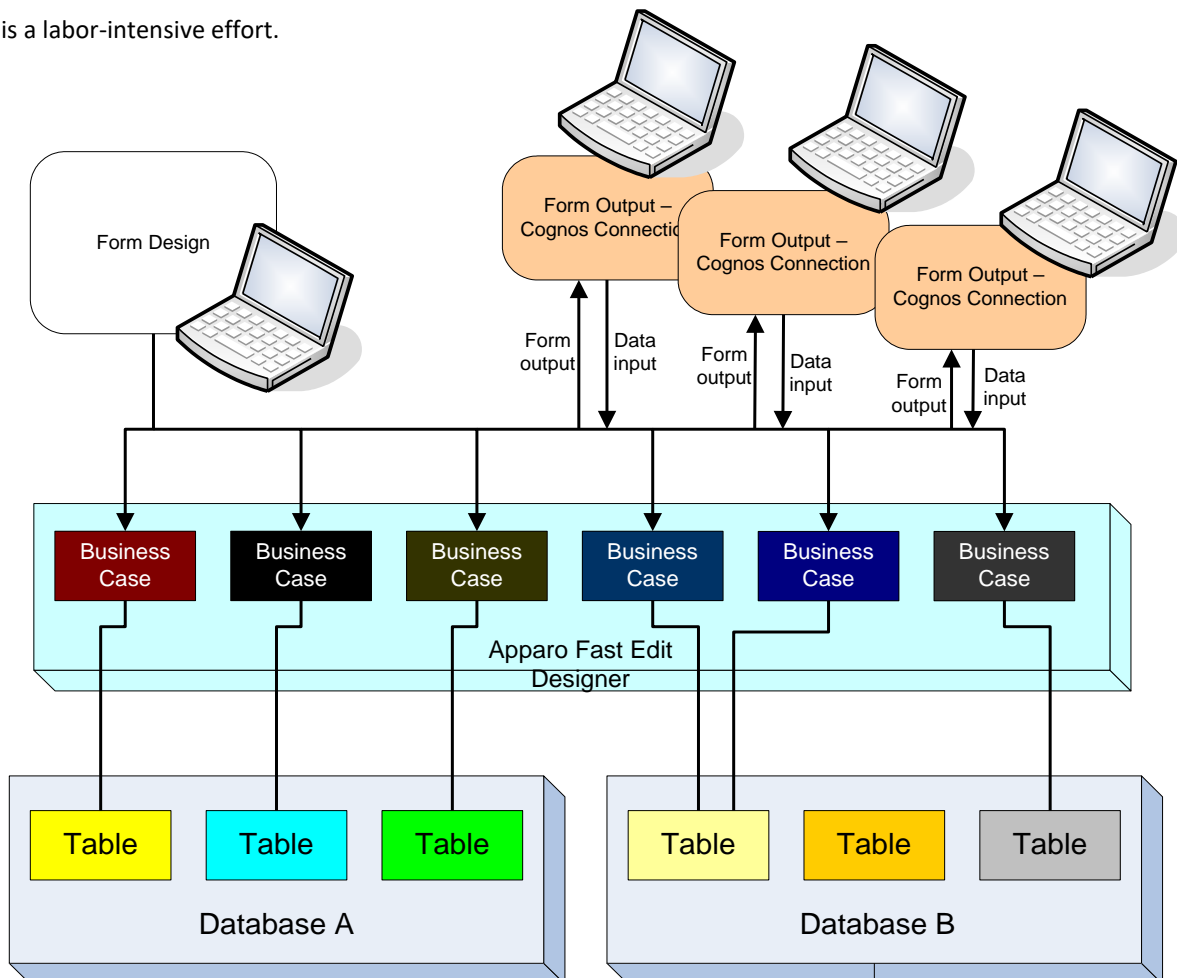
IBM Cognos BI is a software that is optimized for data output.

That means:

a) setting up the input forms and

b) adding the data

is a labor-intensive effort.



With Apparo Fast Edit, a power user can build forms and masks which refer to tables in the database without any programming knowledge.

Members of the same or other business departments can use these forms for quick data entry in stand-alone forms or within a Cognos Report.

2.2 Usage of Apparo Fast Edit

Apparo Fast Edit is intended for collecting any type of data, changing of data and for the output of data.

Typical use cases of Apparo Fast Edit are the maintenance of:

- Master data
- Addresses
- Personnel data
- Project times and status
- User designations, lists and codes
- Report distribution (bursting table)
- ...basically any kind of figures, measures, survey results, plan data etc.

In addition, you can comment on Control Reports and release the comments.

2.3 Business Cases (BC)

The cornerstones of Apparo Fast Edit are the so-called Business Cases.

Each Business Case is an application that can be started in a Business Intelligence System or stand-alone.

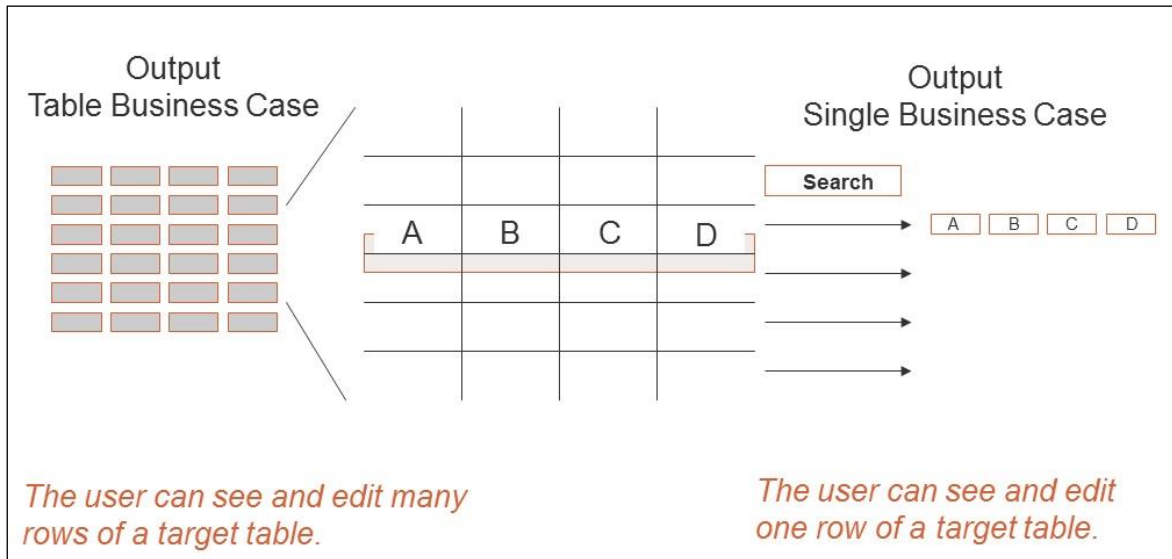
A Business Case contains input fields, prompts, logos, lookups, titles, buttons and functionality like deleting data, inserting, Excel-import and so on.

- While you can include more than one database and table in your Apparo Fast Edit set-up, each Business Case refers to only one table/view (lookups can refer to a different table).
- If you want to combine data of different tables, you need to use e.g. a master-detail relationship between Business Cases, as explained in the section “Linking Business Cases as Master-Detail connection”.
- You can filter the displayed data of a Business Case by another table (e.g. country > factory > personnel display). This is called a data hierarchy and explained in the section “Optimizing Business Cases”.

2.4 Business Case types

Business Cases come in 5 slightly different types:

- the Table Business Case (TBC) and
- the Single Business Case (SBC)
- the Email Business Case (EBC) to send emails
- the Email Import Business Case to import data sent by emails
- Action Business Cases to start scripts, executables, DB functions and so on



Similarities:

- Can be used with search fields, filters, buttons...

Differences:

Table Business Case (TBC)

- Shows data of a whole database table (one row at a time for full display).
- Fewer available widget types than for the single Business Case, due to restrictions in display space (one table per screen).

Single Business Case (SBC)

- Shows only the columns of one database table row. (Via Search next rows available.)
- More widget types than the table Business Case, due to less restriction in space (only one data set per screen).

Typical cases for Single Business Cases:

- Data entry in a single data record
- Display of detailed entries from a table Business Case.
- Data entry of product or staff details.

Typical cases for a Table Business Cases:

- Data entry
- Display of long data lists.
- Data entry from Excel sheets
- Data export into Excel

When you open a single Business Case report, you can transport primary key(s) and report variables to the Business Case.

3 Clients

Clients facilitate the **management of different business units**.

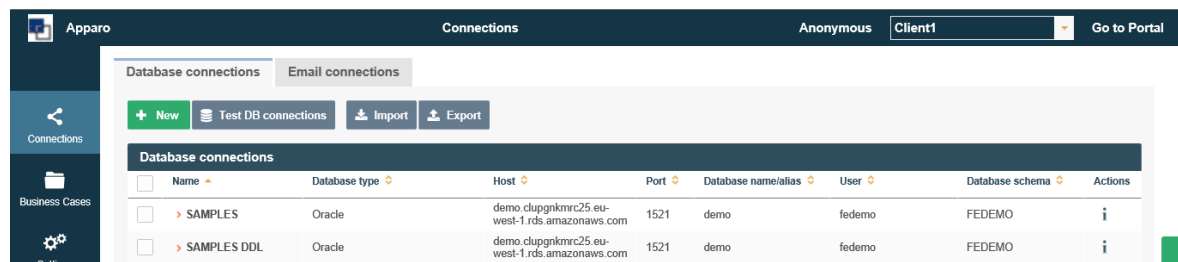
They enable the strict separation of the administration of the business cases in the designer and the easier separation of data from different departments.

Only super-users can create or edit clients, designer users don't have access to this feature.

Every user group can be assigned to one or more clients.

If a user is assigned to more than one client, he can select the client he wants to use in top right corner.

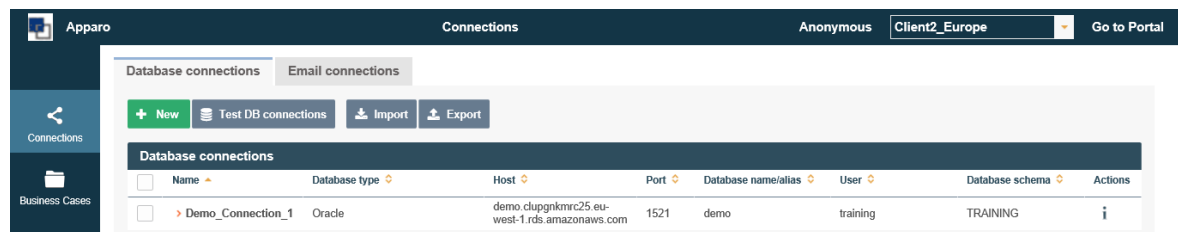
Each client represents a capsulated Apparo Fast Edit environment with own connections, Business Cases, language, settings and security:



The screenshot shows the 'Connections' page for 'Client1'. It features a sidebar with 'Connections', 'Business Cases', and 'Settings'. The main area has tabs for 'Database connections' and 'Email connections'. Below the tabs are buttons for '+ New', 'Test DB connections', 'Import', and 'Export'. A table lists database connections for 'Client1'.

Name	Database type	Host	Port	Database name/alias	User	Database schema	Actions
SAMPLES	Oracle	demo.clupgnkmrc25.eu-west-1.rds.amazonaws.com	1521	demo	fedemo	FEDEMO	i
SAMPLES DDL	Oracle	demo.clupgnkmrc25.eu-west-1.rds.amazonaws.com	1521	demo	fedemo	FEDEMO	i

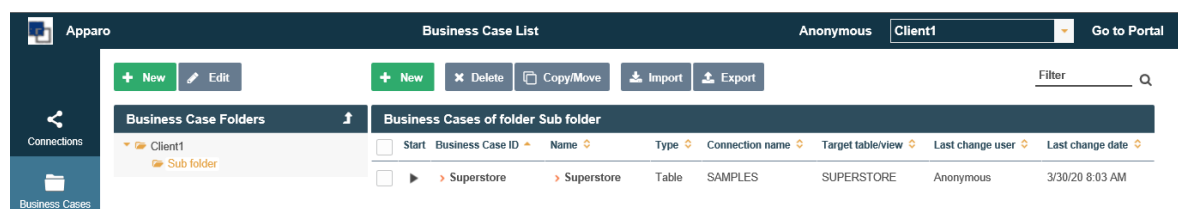
Database connections of 'Client 1'



The screenshot shows the 'Connections' page for 'Client2_Europe'. It features a sidebar with 'Connections', 'Business Cases', and 'Settings'. The main area has tabs for 'Database connections' and 'Email connections'. Below the tabs are buttons for '+ New', 'Test DB connections', 'Import', and 'Export'. A table lists database connections for 'Client2_Europe'.

Name	Database type	Host	Port	Database name/alias	User	Database schema	Actions
Demo_Connection_1	Oracle	demo.clupgnkmrc25.eu-west-1.rds.amazonaws.com	1521	demo	training	TRAINING	i

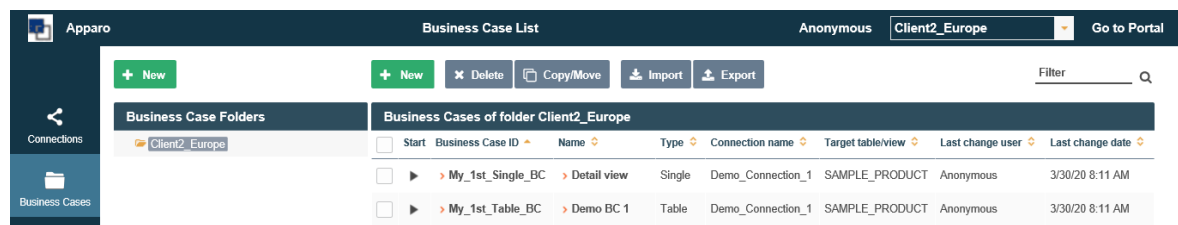
Database connections of 'Client2_Europe'



The screenshot shows the 'Business Case List' page for 'Client1'. It features a sidebar with 'Connections', 'Business Cases', and 'Settings'. The main area has buttons for '+ New', 'Edit', 'Delete', 'Copy/Move', 'Import', and 'Export'. A table lists business cases for 'Client1'.

Business Case ID	Name	Type	Connection name	Target table/view	Last change user	Last change date
Start	Business Case ID					
Superstore	Superstore	Table	SAMPLES	SUPERSTORE	Anonymous	3/30/20 8:03 AM

Business Case of 'Client 1'



The screenshot shows the 'Business Case List' page for 'Client2_Europe'. It features a sidebar with 'Connections', 'Business Cases', and 'Settings'. The main area has buttons for '+ New', 'Delete', 'Copy/Move', 'Import', and 'Export'. A table lists business cases for 'Client2_Europe'.

Business Case ID	Name	Type	Connection name	Target table/view	Last change user	Last change date
My_1st_Single_BC	Detail view	Single	Demo_Connection_1	SAMPLE_PRODUCT	Anonymous	3/30/20 8:11 AM
My_1st_Table_BC	Demo BC 1	Table	Demo_Connection_1	SAMPLE_PRODUCT	Anonymous	3/30/20 8:11 AM

Business Cases of 'Client2_Europe'

3.1 Creating a new client

The client tab can be opened with the button on the left menu. Click on 'New' to add a client:

The screenshot shows the 'Client management and settings' interface. On the left, there is a sidebar with icons for 'Connections', 'Business Cases', 'Settings', and 'Clients'. The main content area has a header with 'Apparo', 'Client management and settings', 'Anonymous', 'Client2_Europe', and 'Go to Portal'. Below the header, there is a table titled 'Installed clients' with columns: Client ID, Client name, Associated security groups, Order, and Actions. The table lists two clients: 'Client1' and 'Client2_Europe'. Above the table, there are buttons for 'New', 'Import', and 'Export'. A green button labeled 'Show description' is on the right side of the table.

Clients can be imported and exported

3.1.1 Client head

When creating a client you will be asked to fill these main settings:

- **Client identifier** = unique ID
- **Client name** = name, that will be shown
- **Client security groups** = assign security groups to this client
- **Client internal description** = description text for documentation purposes

The screenshot shows the 'Client attributes' form. It has a title bar with 'Client attributes' and a close button. The form contains the following fields and controls:

- Client identifier:** A text input field with the value 'Client3'. Below it is a description: 'A unique identifier for the client. This identifier will be used as a folder name on the server's filesystem name for client specific files. Client identifier may contain only basic letters a-z, A-Z, numbers 0-9, dashes or underscore characters.'
- Client name:** A text input field with the value 'Demo client'. Below it is a description: 'An unique name for the client.'
- Client is enabled and Business Cases can be used:** A checkbox that is checked.
- Client security groups (comma separated list):** A text input field with the value 'Administrators, Client3_users'. Below it is a description: 'Each user must be mapped to a client. If an user is member of at least one of these security groups (separated by comma) then the user is member of this client.'
- Client internal description:** A text input field with the value 'Internal demo client'.

At the bottom of the form, there are two buttons: 'OK' (green) and 'CANCEL' (orange).

Click 'OK' to move on

3.1.2 Client security groups

This security groups are used to assign and to check the authorization.

Authorization:

Apart from administrators are only users who are members of the entered security groups, entitled to run the Business Cases of this client.

Assignment:

When you open a Business Case without client ID, e.g. from a BI report or an e-mail, the security group of clients will be used to assign the appropriate Business Case.

Without security group or if the user is member of more than on client, the allocation of Business Cases with the same ID, existing in different clients, is based on the order of the clients in the list from top to bottom.

3.1.3 General settings

The general settings are containing these properties:

- **Application name** = This name is shown in the top left corner. Default is 'Apparo'
- **Application logo** = Optional. A logo can be shown next to the application name
- **Auto import** = Enables client-wide the automated file import (server background) feature
- **Import period** = When auto import is enabled, this is the interval in seconds to check the defined folders for new Excel files for the auto import feature
- **Export dependent connections** = if enabled then all Business Cases exports will contain the necessary database/email connection definitions too. If disabled then the connections get exported too but without valid settings (no server name, no logins etc.).

Apparo

Settings of client Demo client

Anonymous Client2_Europe

Client head

General

Languages

Default numeric & datetime formats

Access rights

Automatic table/column creation

Excel export formats

Business Log

Portal

Colours of portal and designer

Business case standard style

Application name

Apparo

The application name is always displayed at the top left corner.

Application logo

/pictures/logo.png

URL of image that will be used in header as application logo.

Examples:

- <http://example.com/clientLogo.png> - absolute URL address pointing to logo image.
- [/clientLogo.png](#) - if logo image is stored in client's root folder then relative URL address can be used.

Auto import

☐

If "Auto Import" is enabled:

1. Apparo looks into a file feeder-directory (definable for each Business Case) for a defined file name (with pattern, eg. AB*199071.data)
2. Move it to another working-directory (definable for each Business Case)
3. Import rows and move the file to a history-directory (for each Business Case definable too)

Automatic file import period [s]

300

Time period for automatic data file import in seconds. If one import is in progress, the check for next import will be done after the previous import is finished.

The import period can be delayed if one of the imports is running longer than import period time.

Export dependent connections

☒

If enabled then all Business Cases exports will contain the necessary database/email connection definitions too. If disabled then the connections get exported too but without valid settings (no server name, no logins etc.).

OK

CANCEL

Show description

3.1.4 Languages

When creating a client there are no languages installed.

You can import them from: [APPARO_HOME]/FastEdit/languages

When importing clients from older versions, you have to import the languages again, due to new language definitions.

Apparo is supporting many different languages.
If you want to use a language then you must import the language file first.
All language files are stored in the server file directory "[APPARO_HOME]/FastEdit/languages".
If you have no access to server file system then you must first copy the necessary language files from the server to your client computer. Then you can import the language using this web dialog.

The default language is automatically used if the user is using a language that is not installed for this client.
E.g. user language is Spanish but this language is not installed. In this case the default language will be used.

HINT: Don't delete a language if you have Business Cases that are using this language because if you delete a language then all entries of this language (for example labels) are deleted from the Business Cases too.

Warning: If you delete a language then all text entries of this language for all existing Business Case of this client will be deleted too. For example header texts, widget labels and button labels.

Code	Language	Default	Actions
en	English	<input type="checkbox"/>	X ↗
de	German	<input type="checkbox"/>	X ↗

Import language

Show description

3.1.5 Default formats

In 'Default numeric & date time formats' you can adapt the defaults of these date types

If there is no format specified for a widget/variable then it is using this 'standard output format' for the used data type.

Standard output format for number - Show thousand separator character or not, showing leading zeros and define number of decimal places. ?

Standard output format for date/time - Custom date pattern can be defined here. It can be overwritten by the widget or variable output format. ?

Language of the date format - Language that is used when formatting a date/time with the SHORT / MEDIUM / LONG / FULL format and number format. ?

Language	Standard output format for number	Standard output format for date/time	Language for SHORT/MEDIUM/LONG/FULL date formats and number format
English	###.00	MM.dd.yyyy	Use date/time format of this language
German	###.00	dd.MM.yyyy	English (en)

Show description

With the setting **Language for SHORT/MEDIUM/LONG/FULL date formats and number format** you can adapt the used number and date/time format for each language.

In the example we set German with English formats.

3.1.6 Access rights

Define restrictions for accessing secured parts of Apparo Fast Edit. Only users with listed security groups will be able to access these particular functions.

Apparo Client Administrator: has unrestricted access to all features of this client

If no security group is defined, all mapped users will have admin access, which means folder and BC-level access rights will have no impact.

Apparo Connection Administrator: can create, edit and delete database and email server connections

Apparo Designer: manages Business Cases - can create, delete and edit Business Cases and folders

Apparo Import & Export Administrator: can import and export Business Cases and connections

Enter comma separated list of security groups for each Apparo Fast Edit role:

Enter comma separated list of security groups for each Apparo role:

Apparo Client Administrator	Administrators
Apparo Connection Administrator	
Apparo Designer	
Apparo Import & Export Administrator	
Apparo Portal Administrator	

OK CANCEL

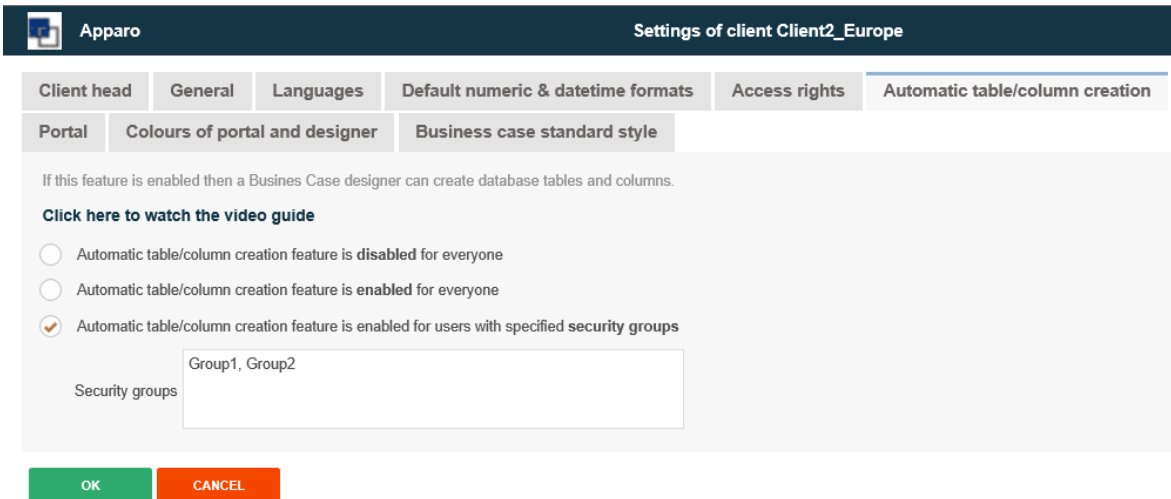
When left empty, no restrictions are set.

3.1.7 Automatic table/column creation

Defines the rights of designer users to use the automatic table creation feature.

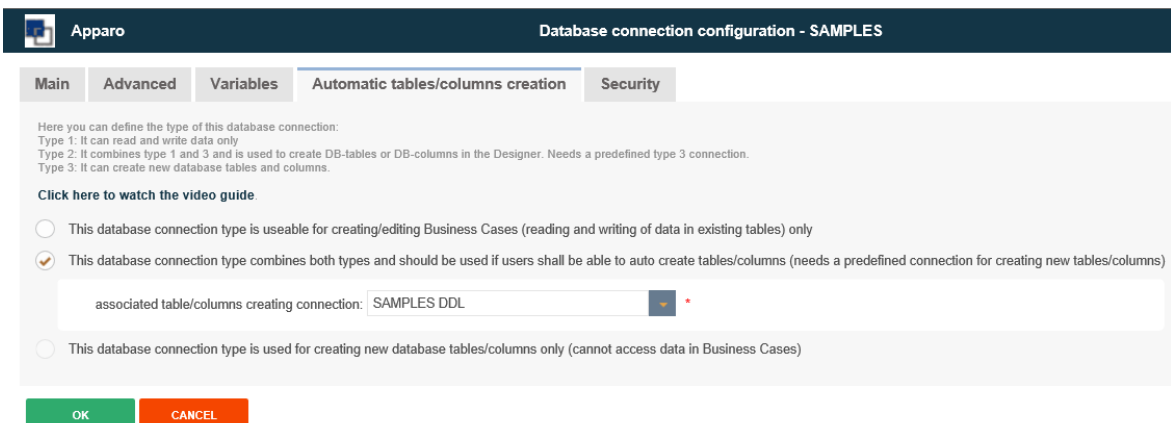
When enabled, Apparo Fast Edit can create automatically tables or columns using a defined database connection.

This feature is disabled by default and can be enabled for everyone or for specified user groups only.



The screenshot shows the 'Settings of client Client2_Europe' dialog box. The 'Automatic table/column creation' tab is selected. It contains a description: 'If this feature is enabled then a Business Case designer can create database tables and columns.' Below this is a link to a video guide. Three radio buttons are present: 'Automatic table/column creation feature is disabled for everyone' (unselected), 'Automatic table/column creation feature is enabled for everyone' (unselected), and 'Automatic table/column creation feature is enabled for users with specified security groups' (selected). A text box labeled 'Security groups' contains the text 'Group1, Group2'. At the bottom are 'OK' and 'CANCEL' buttons.

To use this feature you need a database connection of type DDL (This database connection type is used for creating new database tables/columns only) that will be mapped to database connection of type DML (This database connection type combines both types and should be used if users shall be able to auto create tables/columns)



The screenshot shows the 'Database connection configuration - SAMPLES' dialog box. The 'Automatic tables/columns creation' tab is selected. It contains a description: 'Here you can define the type of this database connection: Type 1: It can read and write data only. Type 2: It combines type 1 and 3 and is used to create DB-tables or DB-columns in the Designer. Needs a predefined type 3 connection. Type 3: It can create new database tables and columns.' Below this is a link to a video guide. Three radio buttons are present: 'This database connection type is useable for creating/editing Business Cases (reading and writing of data in existing tables) only' (unselected), 'This database connection type combines both types and should be used if users shall be able to auto create tables/columns (needs a predefined connection for creating new tables/columns)' (selected), and 'This database connection type is used for creating new database tables/columns only (cannot access data in Business Cases)' (unselected). A text box labeled 'associated table/columns creating connection:' contains the text 'SAMPLES DDL'. At the bottom are 'OK' and 'CANCEL' buttons.

This will be explained more detailed in a later chapter.

3.1.8 Excel export formats

To define the allowed excel export formats

Apparo

Settings of client Client2_Europe

Client head

General

Languages

Default numeric & datetime formats

Access rights

Automatic table/column creation

Excel export formats

Portal

Colours of portal and designer

Business case standard style

You can specify here which Excel file formats can be used for exporting of a Business Case data.

☒ CSV - text file
 ☐ XLS - Excel 2003 and older
 ☒ XLSX - Excel 2007 and newer

OK

CANCEL

3.1.9 Business Log

You can store all user activities (Business Cases, Designer) into a free definable database table.

Apparo

Settings of client Client2_Europe

Anonymous

Client

Client head

General

Languages

Default numeric & datetime formats

Access rights

Automatic table/column creation

Excel export formats

Business Log

Portal

Colours of portal and designer

Business case standard style

You can log all business related events into a custom database table.
The following definition can be used to create such a table.

IBM DB2:

```
CREATE TABLE APP_BUSINESS_LOG (
  LOG_SEQUENCE_NUMBER INTEGER,
  CLIENT_NAME VARCHAR(255),
  USER_NAME VARCHAR(255),
  EVENT_TIMESTAMP TIMESTAMP,
  BUSINESS_CASE_ID VARCHAR(500),
  MESSAGE_CODE INTEGER,
  MESSAGE_TEXT VARCHAR(4000),
  LOG_SEVERITY VARCHAR(255));
```

Oracle:

```
CREATE TABLE APP_BUSINESS_LOG (
  LOG_SEQUENCE_NUMBER INTEGER,
  CLIENT_NAME VARCHAR2(255),
  USER_NAME VARCHAR2(255),
  EVENT_TIMESTAMP TIMESTAMP,
  BUSINESS_CASE_ID VARCHAR2(500),
  MESSAGE_CODE INTEGER,
  MESSAGE_TEXT VARCHAR2(4000),
  LOG_SEVERITY VARCHAR2(255));
```

MS SQL Server:

```
CREATE TABLE APP_BUSINESS_LOG (
  LOG_SEQUENCE_NUMBER INTEGER,
  CLIENT_NAME VARCHAR(255),
  USER_NAME VARCHAR(255),
  EVENT_TIMESTAMP DATETIME,
  BUSINESS_CASE_ID VARCHAR(500),
  MESSAGE_CODE INTEGER,
  MESSAGE_TEXT VARCHAR(4000),
  LOG_SEVERITY VARCHAR(255));
```

☒ **Enable Business log**
 All business events will be logged into own database table.

Business Log database table settings

Database Connection

The database connection that is used for storing log records.

Please select a Database connection

Database schema

The database schema of the log table.

Please select a DB schema

Logging table

The table for the log records.

Please select a log table

3.2 Standard colours of a client

Here you can define the colours of the Apparo Designer and of the Business Cases of this client. Later you can overwrite colours in each Business Case if necessary.

Portal settings are available only in the Standalone Version

Apparo Settings of client Client2_Europe Anonymous Client1

Client head General Languages Default numeric & datetime formats Access rights Automatic table/column creation Excel export formats Business Log

Portal Colours of portal and designer Business case standard style

☒ Use own colour schema for Apparo Designer and Apparo Portal

- ▶ Designer basic colours
- ▶ Designer buttons
- ▶ Portal basic colours
- ▶ Portal buttons

Reset to default colours

OK CANCEL

Show description

Allows to change standard colours by code or colour picker.

▼ Designer basic colours

Application menu

Application menu background colour	#133546	
Application menu font colour	#FFFFFF	
Application menu selected item background	#3E7692	

Main panel

Window background colour	#FFFFFF	
Active tab colour	#F7F7F7	
Active tab marker	#95B3D7	
Inactive tab colour	#E4E4E4	
Title panel background colour	#2E4D5D	
Title panel font colour	#FFFFFF	
Help text colour	#909090	
Mandatory mark colour	#FF0000	

Here you can define default Business Case style settings:

Apparo

Settings of client Client2_Europe

Client head

General

Languages

Default numeric & datetime formats

Access rights

Automatic table/column creation

Portal

Colours of portal and designer

Business case standard style

Sections

Standard Buttons

Dialogs

Widgets

Others

OK

CANCEL

Contains font and style settings for dialogs, buttons, widgets...

Sections

Standard Buttons

Dialogs

Widgets

Widget font styles

Widget content font

Font face

Size

Style

Align

Colour

Arial

12

Normal

Left

#000000

Widget label style

Font face

Size

Style

Align

Colour

Arial

11

Normal

Left

#000000

Sample format text font

Font face

Size

Style

Align

Colour

Arial

10

Normal

Left

#000000

Widget background colours

Edit widget label background colour

rgba(0,0,0,0)

Background colour of edit widget label.

Read-only background

#CCCCCC

Background colour of read-only widgets

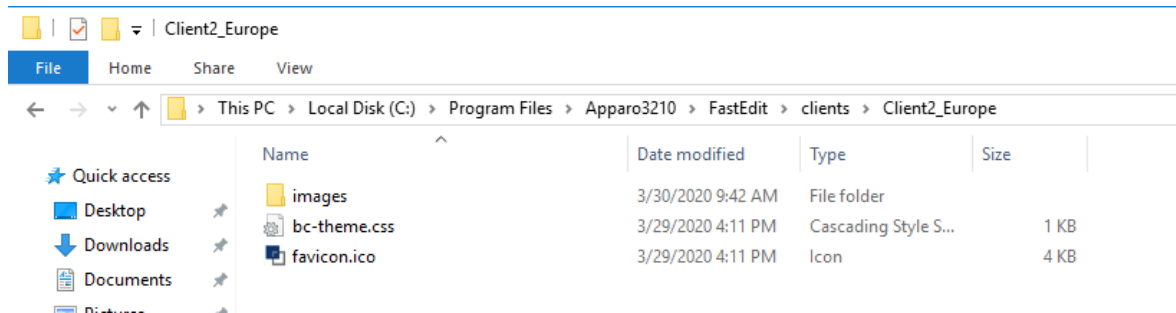
Error background

#FFA0A0

Background colour for widgets that have wrong data

3.3 Using own CSS definitions

In the client folder you will find different ways to create your own templates and to define a preset design.



The files can be found in the folders for each client.

[APPARO_HOME]\FastEdit\clients\<Client ID>

- Bc-theme.css
- Favicon.ico
- All pictures and backgrounds of the designer are stored in the subfolder "images"

3.3.1 Theme.css

The appearance of the designer and the business cases, you can change with CSS:

3.3.1.1 Styles for the Designer

The styles of the designer can be found in the accompanying commentary by theme.css.

Example

```
/* ----- */
/* ----- BUSINESS CASE SETTINGS LEFT MENU ----- */
/* ----- */

/* Menu items background */
div#designerForm\3A featureMenu,
div#settingsTab{
    border:none;
    background-color: #e0e0e0;
    background-image: url("images/menuBkg.png");
    background-repeat: repeat-y;
    background-clip: content-box;
    background-origin: content-box;
}
```

3.3.1.2 Styles for Business Cases

The colons used by the framework in the class designations have to be escaped in the CSS by \:
Unfortunately, this does not work in older Internet Explorer, so we use in the following examples, the hex code "\3A"

Examples for addressing labels in different business case areas in CSS

Hint: !important prevents overwriting of the setting (e.g. by the property files or default styles)

Filter area

```
#businessCaseUIForm\3A searcharea label {
color: #00ff00 !important;
font-size: 2em !important;
}
```

Edit area, Widget-Label

```
#businessCaseUIForm\3A editareaHeader label{
color: #ccc !important;
font-size: 1.5em !important;
}
```

Calculation area

```
#businessCaseUIForm\3A calcarea label{
color: #000 !important;
font-size: 3em !important;
}
```

Bulk update area

```
#businessCaseUIForm\3A bulkupdatearea label{
color: #eee !important;
font-size: 3em !important;
}
```

Header and Footer

```
#businessCaseUIForm\3A header label{
color: #eee !important;
font-size: 3em !important;
}
```

Header or footer can be so addressed only as a unit, but as you can use HTML in header and footer, as well as in almost all settings, e.g. the widget label, you can easily address any elements by Span tags: " TEXT \".

3.3.2 Favicon.ico

The favicon is a small graphic that is displayed by the browser usually in the address bar and / or in the tab:



If you replace this file, please remind the right format .ico

4 The Apparo Designer

4.1 Introduction

The Apparo Designer is the area in which the Business Cases or Business Case sets are created. Anything from the database connection to the data fields, access rights, archiving, corporate design, data import etc. is defined here.

Each Business Case is an own application that can be called separately.

Business Cases can be linked together, so that from the user's perspective, a Business Case can also consist of several masks.

4.2 Start screen with a list of all Business Cases

The screenshot shows the 'Business Case List' interface in the Apparo Designer. The top bar includes the Apparo logo, the title 'Business Case List', the user 'Anonymous', the selected folder 'Client2_Europe', and a 'Go to Portal' button. Below the top bar, there are action buttons: '+ New', '+ New', 'Delete', 'Copy/Move', 'Import', and 'Export'. A search bar labeled 'Filter' is also present. The main content area is divided into two sections: 'Business Case Folders' on the left, showing 'Client2_Europe', and 'Business Cases of folder Client2_Europe' on the right. The right section contains a table with the following data:

Start	Business Case ID	Name	Type	Connection name	Target table/view	Last change user	Last change date
<input type="checkbox"/>	▶	My_1st_Single_BC	Detail view	Single	Demo_Connection_1	SAMPLE_PRODUCT	Anonymous 3/30/20 8:11 AM
<input type="checkbox"/>	▶	My_1st_Table_BC	Demo BC 1	Table	Demo_Connection_1	SAMPLE_PRODUCT	Anonymous 3/30/20 8:11 AM

A 'Show description' button is located on the right side of the table.

At the start of the Apparo Designer, you will see a list of all business cases that are stored in the Apparo Repository. If the Apparo Repository does not contain any definition, this list will have no entries.

4.3 Description and normal mode

In the description mode most of the settings are explained briefly while the normal mode lacks these descriptions.

Example of the description mode:

Apparo Business Case List Anonymous Client1 Go to Portal

Here you see the list of all Business Cases of this system. A Business Case is a small program for displaying, filtering and editing of data, Excel import/export, file up-/download and other data output. If you want to create a new Business Case then you need a target table in your database and a database connection. You can import and export whole Business Cases to/from another system too. For a brief overview watch: [Video](#)

[New](#) [Edit](#) [New](#) [Delete](#) [Copy/Move](#) [Import](#) [Export](#) Filter

Business Case Folders

- Client1
 - Sub folder
 - Sub sub folder

Business Cases of folder Sub sub folder

<input type="checkbox"/>	Start	Business Case ID	Name	Type	Connection name	Target table/view	Last change user	Last change date
<input type="checkbox"/>	▶	> Superstore (1)	> Superstore	Table	SAMPLES	SUPERSTORE	Anonymous	3/30/20 11:41 AM

The same page in normal mode:

Apparo Business Case List Anonymous Client1 Go to Portal

[New](#) [Edit](#) [New](#) [Delete](#) [Copy/Move](#) [Import](#) [Export](#) Filter

Business Case Folders

- Client1
 - Sub folder
 - Sub sub folder

Business Cases of folder Sub sub folder

<input type="checkbox"/>	Start	Business Case ID	Name	Type	Connection name	Target table/view	Last change user	Last change date
<input type="checkbox"/>	▶	> Superstore (1)	> Superstore	Table	SAMPLES	SUPERSTORE	Anonymous	3/30/20 11:41 AM

The Designer is switching the mode by clicking the [Show description](#) button on the right side of the screen.

4.4 Managing of Business Cases



Filter

The following buttons are at your disposal:

- New - creates a new Business Case
- Delete - deletes all selected Business Cases
- Copy - copies all selected Business Cases
- Import - imports Business Cases from a file
- Export - exports selected Business Cases into a file
- Filter - filters all business cases from the input string by its ID

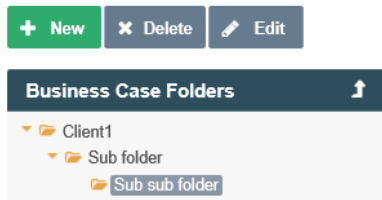
The sort can be changed by clicking the orange arrows:

☐ Start Business Case ID Name Type Connection name Target table/view Last change user Last change date

4.5 Designer Folders

Business Cases can be grouped in folders. Inside the folder you can create subfolders.

For the Folders pane, there are three buttons:



Depending on the given rights, the user can:

- Create new folders and subfolders
- Delete folder if empty (subfolders)
- Change the properties of the folder

Folder properties:

Settings of this Business Case folder Sub folder

Name of this Business Case folder	Sub folder
Description of this Business Case folder	<input type="text"/>
Necessary security group to see and open this folder	Opening_group
Necessary security group to edit/delete this folder	Edit_group
Necessary security group for running the included Business Cases of this folder and subfolders	User_group_BC_access
Necessary security groups with limited access to Business Cases of this folder and subfolders	User_group_read_only_mode

The following properties can be changed:

- Name of the folder
- The necessary security group to open the folder
- The necessary security group to edit the folder
- The necessary security group to execute containing Business Cases
- The necessary security group to execute containing Business Cases in limited mode (read only)

4.6 Connections

Apparo Fast Edit

Connections Administrator Demo

Database connections Email connections

+ New Test DB connections Import Export

	Name	Usage	Database type	Host	Port	Database name/alias	User	Database schema	Actions
<input type="checkbox"/>	> SAMPLES	reading/writing only (DML)	Oracle	demo.clupgnkmc25.eu-west-1.rds.amazonaws.com	1521	demo	fedemo	FEDEMO	i
<input type="checkbox"/>	> SAMPLES DDL	creating tables/columns (DDL)	Oracle	demo.clupgnkmc25.eu-west-1.rds.amazonaws.com	1521	demo	fedemo	FEDEMO	i

Show description


4.6.1 Buttons

The following buttons are at your disposal:



- New - creates a new database connection
- Test DB-Connection - is testing all selected DB connections
- Import - imports DB connections from a file
- Export - exports all selected DB connections into a file

4.6.2 Creating a new database connection

Click on the button 

Main
Advanced
Variables
Automatic tables/columns creation
Security

Connection name

The database connection name is the name that will be used in all Business Cases.

Database type

Exasol
IBM DB2
IBM DB2 Client
IBM DB2 I
IBM dashDB
IBM Netezza
Informix
MS SQL Server 2008-2019
Oracle
Oracle (using service name)
Oracle Client
PostgreSQL
SAP HANA
SAP Sybase ASE
SAP Sybase IQ
SAP Sybase SQL Anywhere
Teradata
MySQL

Database host

Host name or host_name\instance_name for SQL Server named instance.

TCP/IP port

Database host

Host name or host_name\instance_name for SQL Server named instance.

TCP/IP port

The TCP/IP port for communication:

- Oracle: default value 1521
- MS SQL Server: default 1433 (Please note: If you want to use named instances then please see at SQL Server Configuration Manager / SQL Server Network Configuration / Protocols for INSTANCE NAME / TCP/IP / IPAddresses for the correct port number or let this field empty if the port number is changing dynamically)
- IBM DB2: default: 6789 or 50000

Database name

The database instance name (SID for Oracle) or database name (for MS SQL Server):

Windows authentication
☐

In Windows authentication mode the current Windows user that is used for the Windows service of Apparo will be used for authentication for the MS SQL Server.
MS SQL Server, the optional BI system and Apparo must be in the same Active Directory domain.
In this mode, the database connection pooling will be automatically disabled.
Please see the description text of the ?-icon for detail information about Windows authentication.

Database user

Password

OK
CANCEL

Depending on the selected connection, you must configure the following settings:

- Connection name - Freely selectable unique identifier for the connection
- Database type - select from list your database type
- Database host - IP address or host name of the database
- TCP/IP Port - listening port of the database
- Database name - Name of the DB
- Database user - user name of the login
- Password - password of the login

5 Creating a Table Business Case (TBC)

In the following section, we will show the general steps to create a Business Case by using a table Business Case as example. All important points (creating, widgets, design, and output) are explained. Remember that each Business Case refers to a specific table in a database. If you need data from several tables, you need to work with e.g. master-detail relationships or **Lookup** widgets.

5.1 Areas of a Table Business Case

A Table Business Case consists of different (partially optional) areas

The screenshot displays the Apparo interface with the following components:

- Header area:** Includes a header title, header description, and a widget in the filter area with a search button and a reset filters button.
- Filter area:** Contains a widget in the filter area and another filter widget.
- Bulk update area:** Features a widget in the bulk update area and an update button.
- Edit area:** Shows a widget in the edit area and another edit widget, with a table of data including columns for product type (e.g., Jackets, Caps) and name (e.g., James Dean Tribute, The Sparrow).
- Calculation area:** Includes a widget in the calculation area and a page counter (Page: 1 / 2).
- Navigation area:** Contains buttons for OK, CANCEL, CLOSE, INSERT, EXPORT TO EXCEL, and EXCEL ROW-IMPORT.
- Footer area:** Includes a footer title and footer description.

Header area	- includes the title and description
Filter area	- for example, contains filter widgets to filter the data output
Bulk update area	- mass update panel
Edit area	- to modify existing data
Insert area	- for adding new records
Calculation area	- used to display information, such as text or calculations of variables
Navigation area	- includes page counter, navigation and buttons for resizing
Button area	- contains buttons
Footer area	- comparable to the header area

5.2 Create a new Business Case

Click on the button:



Now select the entry ,Table‘

Apparo

Anonymous Client2_Europe

Please select type of Business Case you want to create now

	Table	A table Business Case is showing many data rows on the same page. The user can filter the data, edit, import from Excel, export to Excel and so on.
	Single	A single Business Case is showing just one data row only.
	Set	A grouping of multiple Business Cases (table/single) for more comfortable usage. You can define global filters that are filtering all Business Cases automatically too.
	Email import	Importing Excel data directly by email - send Excel sheets using email attachments and Apparo will import the Excel data directly into the database including file uploads. No web browser is necessary, just an email.
	Email	An eMail Business Case is a definition of an email text including usage behavior and can be used in another Business Cases of type 'table' or 'single' only. In these Business Cases it is possible to define buttons that can use this eMail Business Case.
	Action	Purpose of Action Business Case is to execute scripts or database procedures that can be called from a report/HTML page. Usage of AJAX and Javascript for automatically executing in the background is possible too.

CANCEL

Show description

Business Case selection

Following, the general settings for the business case

Please provide a unique short name (ID), a name and select the target table.

The description is optional and can contain declarations, release notes, or other information.

If multiple database connections are set up, this selection is automatically extended by the points 'database connection' and 'database schema'.

Apparo

Main database Business Case settings

Identifier / Short name	Unique identifier	*
Business Case name	BC name	*
Target database table/view	SAMPLE_CARS	*
Notes		

NEXT

CANCEL

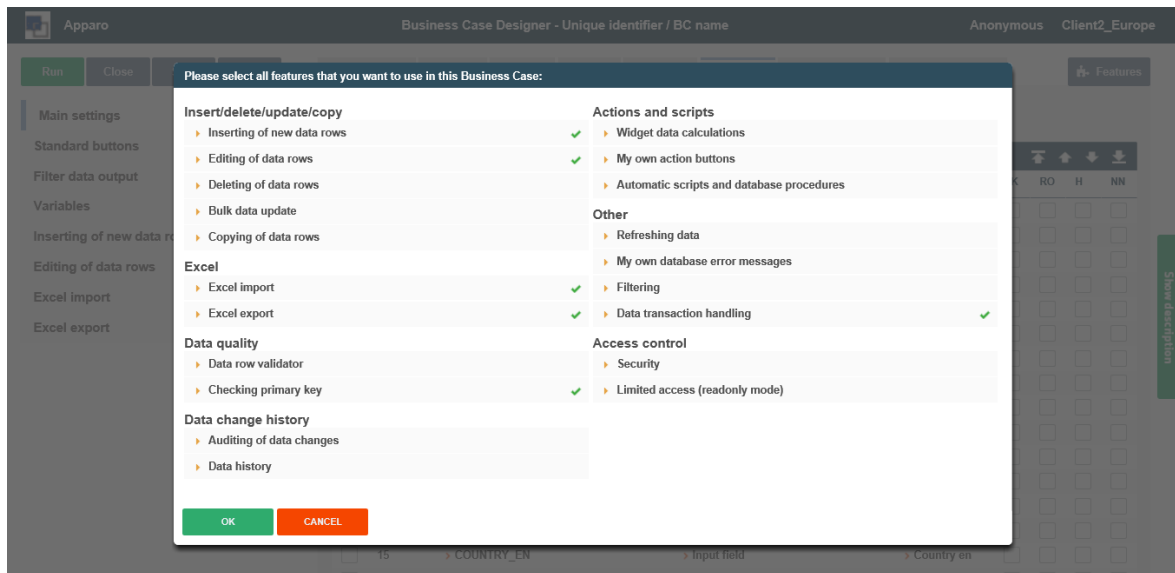
Main settings

5.3 Business Case features

The functions of a business case open automatically after creating a business case.

If the business case is opened later for editing again, you can open the feature selection with the button on

top right corner: 



The features are divided into seven sections. The various functions can be enabled or disabled as needed. If a function is activated, then the selection menu in the Business Case edit view will be extended accordingly.

The advantage of this activation is obvious, if the Business Case is opened for editing again after some time, then you can see with one look at the menu, which features are used in this business case.

5.4 Features areas and features

- **Insert/delete/update/copy**
 - Inserting of new data rows
 - Deleting of data rows
 - Bulk data update
 - Copying of data rows
- **Excel**
 - Excel Import
 - Excel Export
- **Data quality**
 - Data row validator
 - Checking primary key
- **Data change history**
 - Auditing of data changes
 - Data history
- **Actions and scripts**
 - Widget data calculations
 - My own action buttons
 - Automatic scripts and database procedures
- **Other**
 - Reloading data
 - My own database error messages
 - Filtering
 - Data transaction handling
- **Access control**
 - Security
 - Limited access

5.5 Edit view of the Business Case

The edit screen is divided into two areas:

Menu bar, the buttons on the controller and all activated functions as menu items contains.

Buttons:

- **Start** - saves all changes and starts the business case
- **Close** - saves all changes and closes the edit view
- **Save** - saves all changes
- **Cancel** - discards any unsaved changes and closes the Business Case

Settings area, contains settings for the various functions and optionally divided again into tabs.

Edit view in activation of all functions

Edit view in deactivation of all functions. Only the basic functions are displayed.

5.6 Business Case Settings

The main settings are divided into several tabs and are containing the settings of the data source and the optics of the business case. Above all, it contains the widget settings.

Widgets are the actual control and output elements of a business case. This can be a filter, input or selection fields, buttons and more.

Because of their importance are Widgets the first tab you see when you open the Main settings.

The screenshot shows the Apparo Business Case Designer interface. The top bar includes the Apparo logo, the title 'Business Case Designer - My_1st_Table_BC / Demo BC 1', and user information 'Anonymous' and 'Client2_Europe'. Below the top bar is a navigation menu with tabs: 'Run', 'Close', 'Save', 'Cancel', 'Target table', 'Header', 'Footer', 'Visual', 'Colours', 'Widgets' (selected), 'Row ordering', 'Link into Portal', and 'Features'. On the left side, there is a sidebar with a list of settings: 'Main settings' (selected), 'Standard buttons', 'Filter data output', 'Variables', 'Inserting of new data rows', 'Editing of data rows', 'Excel import', and 'Excel export'. The main area displays the 'Widgets' tab, which is divided into two sections: 'Filtering widgets' and 'Editing widgets'. The 'Filtering widgets' section has a table with columns: Row, Column, Column name, Widget type, Title, and H. It contains two rows: one for 'ACTIVE' with a 'Checkbox' widget type and a title 'Filter for active products', and another for 'PRODUCT_LINE_ID' with a 'Lookup multiselect (for all tables)' widget type and a title 'Filter for product lines'. The 'Editing widgets' section has a table with columns: Column, Column name, Widget type, Title, PK, RO, H, and NN. It contains three rows: one for 'PRODUCT_ID' with an 'Input field' widget type and a title 'Product id', one for 'PRODUCT_LINE_ID' with a 'Lookup dropdown (for all tables)' widget type and a title 'Product line', and one for 'PRODUCT NAME EN' with an 'Input field' widget type and a title 'Product'. A 'Show description' button is located on the right side of the 'Editing widgets' table.

Main settings, Widgets

5.7 Widgets

This chapter covers the central area of a Business Case.

Here you can have different widgets that are normally connected with the target table, positioned in different areas.

Each widget has its own individual settings.

5.7.1 Edit view

You can open the settings of an existing widget, by clicking on the column name or widget type:

Editing widgets									
<input type="checkbox"/>	Column	Column name	Widget type	Title	PK	RO	H	NN	
<input type="checkbox"/>	1	> PRODUCT_ID	> Input field	> Product id	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
<input type="checkbox"/>	2	> PRODUCT_LINE_ID	> Lookup dropdown (for all tables)	> Product line	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	3	> PRODUCT_NAME_EN	> Input field	> Product name en	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
<input type="checkbox"/>	4	> PRODUCT_COLOUR	> Input field	> Product colour	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	5	> PRODUCT_SIZE	> Input field	> Product size	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	6	> PRODUCT_MODEL	> Input field	> Product model	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	7	> PRODUCT_MANUF	> Input field	> Product manuf	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	8	> PRODUCT_DESCR	> Text area	> Product descr	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Depending on the type of widget has the edit view different options, divided into tabs.

Apparo

Business Case Designer - My_1st_Table_BC / Demo BC 1

Widget settings of database column PRODUCT_LINE_ID

Widget type

Mapping & Other

Features

Lookup & Dropdown & Multiselect

Visual

Help texts

Data output format

Column name

PRODUCT_LINE_ID

Default value

Constant value

Do not use constant value

Variable for using content in detail BC

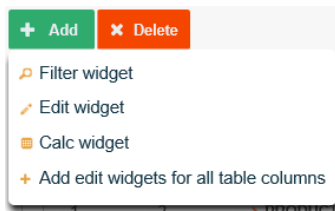
OK

CANCEL

Widget edit view for the type ,Lookup dropdown (for all tables)‘

5.8 Widget types and areas

When creating a new widget, you will first be asked for which area it is intended:



The option to add widgets for all database columns 'Add edit widgets for all table columns', is adding an 'input field' widget for each existing database column, if no widget is existing for this database column.

The corresponding areas for the widget types are:

- **Filter area**
- **Edit area**
- **Bulk update area**
- **Calculation area**

Based on the area for which the widget is thought, is there a selection of different widget types:

Input field - A standard entry field which allows the input of alphanumeric data

Text area - A multiline entry area that allows formatted text

Checkbox - Allows exactly two values, checked or not checked

Simple dropdown (for target table only) - Based on data in the target table

Lookup dropdown (for all tables) - Replaces numerical values with plain text from a 2nd table

Simple multiselect - Select multiple values

Lookup multiselect - Multiselect based on a lookup table

Label - Enables you to output text

Label with variables - Enables the output of text and values of variables

Spacer & Title - To set up void spaces between individual widgets

Business Case Link - To call e.g. detail BCs, data values are passed here

File Upload/Download - To attach files to data rows

Business Case Link and File Upload/Download can only be used in the edit area.

5.9 Widgets in the Edit Area

The edit area in a table business case (TBC) is mainly used for displaying data in list form and gives users the ability to edit the data.

The screenshot shows the APPARO Fast Edit interface with the 'Widgets' tab selected. The sidebar on the left contains navigation options: Run, Close, Save, Cancel, Main settings, Standard buttons, Filter data output, Variables, Inserting of new data rows, Editing of data rows, Excel import, and Excel export. The main area displays two tables: 'Filtering widgets' and 'Editing widgets'. The 'Filtering widgets' table has columns for Row, Column, Column name, Widget type, Title, and H. The 'Editing widgets' table has columns for Column, Column name, Widget type, Title, PK, RO, H, and NN. Both tables list various widgets like checkboxes, lookup multiselect, input fields, and text areas.

Widgets of an area are grouped
Example for edit widgets in the user view:

The screenshot shows the APPARO Fast Edit user view for 'Workflow Demo Step1 Data Input'. The interface includes a header with 'Demonstration Apparo Fast Edit' and 'Anonymous Demonstration'. Below the header, there are filters for 'Product line filter', 'My status' data filter, and 'Controlling status filter'. A table displays data for various products and plans, with a blue circle highlighting the 'My status' column and its dropdown menu.

In the edit area you can choose between these types of widgets:

- **Input field** - A standard entry field which allows the input of alphanumeric data
- **Text area** - A multiline entry area that allows formatted text
- **Checkbox** - Allows exactly two values, checked or not checked
- **Simple dropdown** (for target table only) - Based on data in the target table
- **Lookup dropdown** (for all tables) - Replaces numerical values with plain text from a 2nd table
- **Label** - Enables you to output text
- **Label with variables** - Enables the output of text and values of variables
- **Spacer & Title** - To set up void spaces between individual widgets
- **Business Case Link** - To call e.g. detail BCs, data values are passed here
- **File Upload/Download** - To attach files to data rows

5.10 Special functions in the widget settings

There are some special functions in the widget settings:

5.11 Reading and Writing Expressions

Reading and writing expressions allow the usage of SQL to manipulate data before it is shown to users or stored to the database.

Demonstration Apparo Fast Edit Business Case Designer - My_1st_Table_BC / Demo BC 1

Widget settings of database column **PRODUCT_ID**

Widget type Mapping & Other Features Visual Help texts Data output format

Column name PRODUCT_ID ☒ Enable expressions

Read expression PRODUCT_ID || PRODUCT_NAME

Write expression

☐ The used read and write expressions having an inverse behaviour

Variables are allowed here

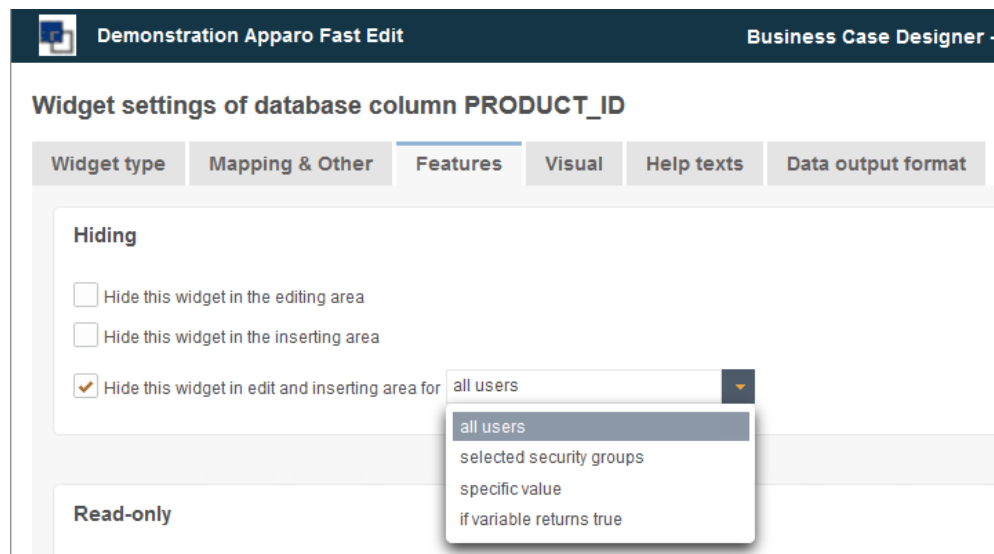
Common examples for expressions are:

- TRIM() - Removes spaces from strings
- UPPER() - Turns all letters into upper cases
- LOWER() - Turns all letters into lower cases

5.12 Conditional options

Many functions can be controlled with reference to conditions.

Thus, there are e.g. for the function 'Hidden', which hides a widget for the user when activated, several options.

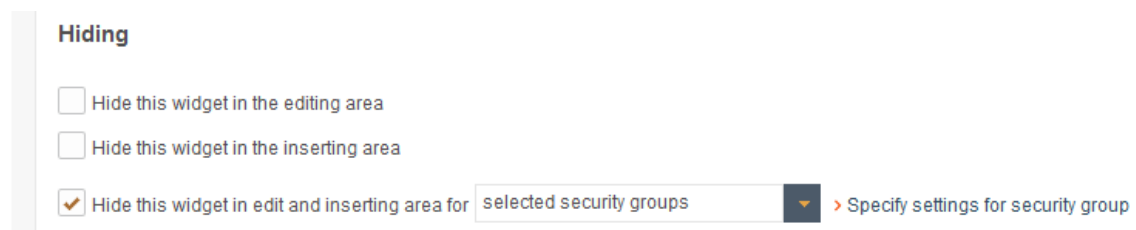


For all users

This option is set by default. It hides the widget for all users.

For selected security groups

This hides the widget, but only for users who are member of one of the entered user groups. Other users can see the widget.



Hide the widget for the selected user groups

Widget security groups definition - Hidden

Existing security groups

Security group

Group_A

Add new security group

Security group

Group_B

ADD SECURITY GROUP

BACK TO WIDGET EDITOR

Security group editor

For a specific value

The option applies here as soon as the value of one of the columns in the target table in the corresponding data row matches with the stored value.

In our example, the widget would be hidden once in a data row in the office ID column the value '3' appears.

Widget type

Mapping & Other

Features

Visual

Help texts

Data output format

Hiding

☐ Hide this widget in the editing area

☐ Hide this widget in the inserting area

☒ Hide this widget in edit and inserting area for

specific value

PRODUCT_ID

has value

3

The values can also be configured dynamically by using variables. <%VARIABLE%>

5.13 Widget settings for the example ,Input field'

Each widget type has its own settings. The following settings using the example of the widget type ,input field' can be found in almost all widget types. Distinctive features of each widget type are described in the following chapter ,More widget settings'

Widget settings of database column **PRODUCT_ID**

The screenshot shows the 'Widget type' tab selected. On the left, a list of widget types is displayed with radio buttons: Input field (selected), Text area, Checkbox, Simple dropdown (target table only), Lookup dropdown (for all tables), Multi select lookup, Label, Label with variables, Spacer & Title, Business Case link, and File upload/download. On the right, the description for the 'Input field' widget is shown: 'Input field: Simple input field with one line.' At the bottom, there are 'OK' and 'CANCEL' buttons.

5.13.1 Mapping & Other

Widget settings of database column **PRODUCT_ID**

The screenshot shows the 'Mapping & Other' tab selected. The 'Column name' is set to 'PRODUCT_ID'. There is a checkbox for 'Enable expressions'. The 'Default value' is set to '<%VARIABLE%>' for 'all users'. The 'Constant value' is set to 'Use constant value in insert case if variable returns 'true'' for 'all users'. The 'Variable name' is set to '<%TRUE_VAR%>'. The 'Variable for using content in detail BC' is empty. At the bottom, there are 'OK' and 'CANCEL' buttons.

Column name

Here you can select the database column that is connected to this widget. The widget is reading the content of the column and is writing to this column. It is possible to use a variable in this column name too.

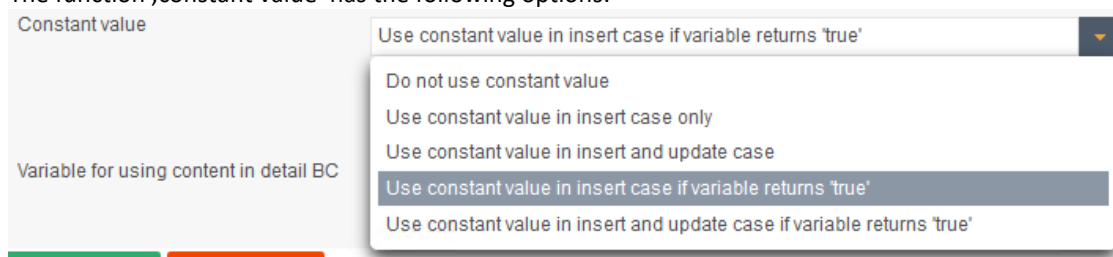
Default value

If you define a default value (using a variable is possible too) then the user will see this value in this input field in the **inserting** area. It is possible to define different default values for different user roles/groups. Use English format to define numeric or date values. Default value of lookup widget must be the lookup key value.

Constant value

A constant value is a value that will **be always used for this widget**. Even if the widget is hidden, read-only or if the user is inserting a value, then the constant value will be used. It is possible to define different constant values for different user roles/groups.

The function ,constant value' has the following options:



Variable for using content in detail BC

If this Business Case has a widget of type 'Business Case Link' for opening a detail Business Case, it is possible to define a variable that contains the value of the current widget. The detail Business Case can use this report variable with the current value of this widget for example for output.

Important: In the detail Business Case you must define this report variable in tab "Variables" too.

5.13.2 Features

In the tab Features you can control the behavior of the widget in detail.

Widget type	Mapping & Other	Features	Visual	Help texts	Data output format
Hiding <input type="checkbox"/> Hide this widget in the editing area <input type="checkbox"/> Hide this widget in the inserting area <input type="checkbox"/> Hide this widget in edit and inserting area for <input type="text" value="all users"/>					
Read-only <input type="checkbox"/> Read-only in edit and inserting area for <input type="text" value="all users"/> <input type="checkbox"/> Read-only in edit area for <input type="text" value="all users"/> <input type="checkbox"/> Read-only in inserting area <input type="text" value="all users"/>					
Other <input checked="" type="checkbox"/> Database column is the primary key or a part of it <input type="checkbox"/> Database column is computed by database (for example using a database trigger or auto-increment feature) <input type="checkbox"/> Show a small icon for easier deleting of the complete content of this widget <input checked="" type="checkbox"/> Value is mandatory (not null) <input type="checkbox"/> Remove all spaces at the begin and at the end automatically <input type="checkbox"/> Hide output in password style <input type="checkbox"/> Store value in upper case <input type="checkbox"/> Store value in lower case					
<div>OK</div> <div>CANCEL</div>					

Hiding-Group

Includes options for hiding widgets.

Hiding <input type="checkbox"/> Hide this widget in the inserting area <input checked="" type="checkbox"/> Hide this widget in edit and inserting area for <input type="text" value="all users"/>
--

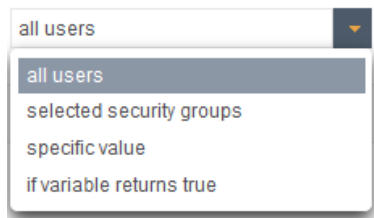
Hide this widget in the inserting area

If enabled, the user will not see this widget in the inserting area. If you use a constant value then it will be used no matter if the widget is hidden or not.

Hide this widget in edit and inserting area for

The data field is to be used, but not shown in insert and editing area, optional security group based. That means this widget can be hidden for certain user groups only.

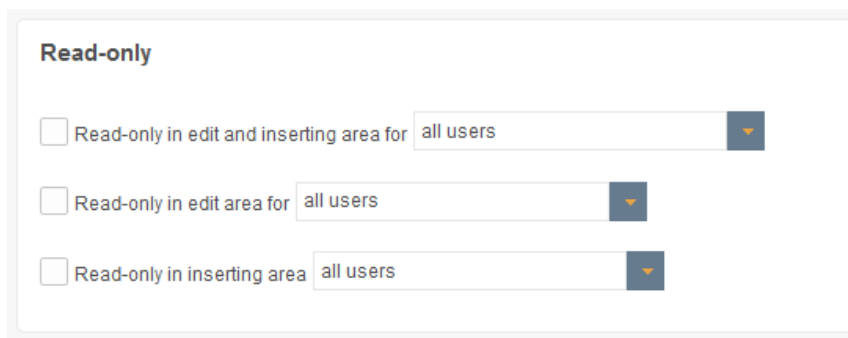
Options:



A dropdown menu with the following options: all users, all users, selected security groups, specific value, and if variable returns true. The first 'all users' option is highlighted.

Read only group

Includes options to disable the entering or changing of values in widgets



A panel titled 'Read-only' containing three checkboxes and dropdown menus:

- ☐ Read-only in edit and inserting area for all users
- ☐ Read-only in edit area for all users
- ☐ Read-only in inserting area all users

Read-only in edit and inserting area for

The data field cannot be altered in editing and inserting area but it is still visible with another background color, optional security group based.

Options:

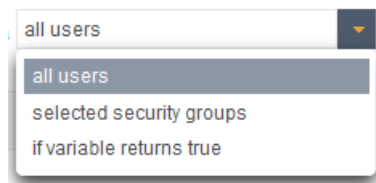


A dropdown menu with the following options: all users, all users, selected security groups, and if variable returns true. The first 'all users' option is highlighted.

Read-only in edit area for

The data field cannot be altered in editing area, optional security group based. Read-only widgets have an own background color.

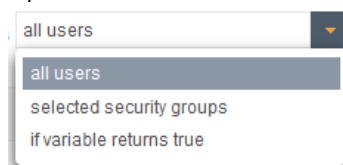
Options:



Read-only in inserting area

The data field cannot be altered in inserting area, optional security group based. Read-only widgets have an own background colour.

Options:



Other-Group

Contains all other settings

Other

- ☒ Database column is the primary key or a part of it
- ☐ Database column is computed by database (for example using a database trigger or auto-increment feature)
- ☐ Show a small icon for easier deleting of the complete content of this widget
- ☒ Value is mandatory (not null)
- ☐ Remove all spaces at the begin and at the end automatically
- ☐ Hide output in password style
- ☐ Store value in upper case
- ☐ Store value in lower case

Database column is the primary key or a part of it

The widget is the primary key of the underlying table or is an part of the key (with combined keys). This definition is independent of the primary key definition in the database and at least one column must be defined as primary key. A primary key is identifying an unique data row of the target table/view.

Database column is computed by database

(for example using a database trigger or auto-increment feature)

The database table column value is filled automatically by the database (e.g. with triggers, auto-increment field). Apparo Fast Edit is not changing this value in the target table.

Show a small icon for easier deleting of the complete content of this widget

Showing a small delete icon for deleting the widget content.

Value is mandatory (not null)

If a widget value is mandatory then the user must input a value into this widget (or using default or constant value). The definition of this behavior is independent from the definition of the target table column in the database.

If a filtering widget is mandatory it is a good idea to define a default value for him too. You will avoid some error messages at the Business Case startup.

Remove all spaces at the begin and at the end automatically

If enabled then all spaces at the begin and end are removed automatically before storing into database table

Hide output in password style

The input is hidden on screen

Store value in upper case

If enabled then all characters are changed to upper case before storing into database table

Store value in lower case

If enabled then all characters are changed to lower case before storing into database table

5.13.3 Visual

In the tab 'Visual' you will find the header (column heading), and settings for the layout, and settings to limit the maximum allowed input length in this widget.

By default the maximum entry length is defined by the database column definition, for example Varchar(20) allows a maximum of 20 alphanumeric characters. This can be further limited by the input of an own value

Widget settings of database column PRODUCT_ID

Widget type	Mapping & Other	Features	Visual	Help texts	Data output format
Column label					
Language		Column label			
German		<input type="text" value=""/>			
English		Product id			
Visual settings					
Label style	Font face	Size	Style	Align	Colour
	Arial	11	Normal	Left	#000000
Background colour for label	No background colour is defined				
	<input type="button" value="ADD"/>				
Widget align	Right				
Column width (px)	80				
Border colour	#D0D0D0				
Background colour	No background colour is defined				
	<input type="button" value="ADD"/>				
Font	Font face	Size	Style	Align	Colour
	Arial	12	Normal	Right	#000000
Maximum input length	Defined by database column definition				
<input type="button" value="OK"/> <input type="button" value="CANCEL"/>					

The layout can also be controlled global using the client settings or by CSS.

5.13.4 Help texts

Contains the settings for description and hint text

Widget type	Mapping & Other	Features	Visual	Help texts	Data output format						
Description text <table border="1"> <thead> <tr> <th>Language</th> <th>Description text</th> </tr> </thead> <tbody> <tr> <td>German</td> <td><input type="text"/></td> </tr> <tr> <td>English</td> <td><input type="text"/></td> </tr> </tbody> </table>						Language	Description text	German	<input type="text"/>	English	<input type="text"/>
Language	Description text										
German	<input type="text"/>										
English	<input type="text"/>										
Hint text <table border="1"> <thead> <tr> <th>Language</th> <th>Hint text</th> </tr> </thead> <tbody> <tr> <td>German</td> <td><input type="text"/></td> </tr> <tr> <td>English</td> <td><input type="text"/></td> </tr> </tbody> </table>						Language	Hint text	German	<input type="text"/>	English	<input type="text"/>
Language	Hint text										
German	<input type="text"/>										
English	<input type="text"/>										
<div>OK CANCEL</div>											

Description text

This text can describe the widget and can be helpful for the user. You can add a more detail description text for each installed language.

The user is seeing this text if he is pointing to the label of this widget.

Hint text

The hint text is displayed only if the widget has no value.

Is displayed in the input area in gray text, e.g. 'Enter date in the format: dd.MM.yy'

6 Single Business Cases (SBC)

A single business case (SBC) is used to represent a single data set (database row). A typical application is a data entry screen or a detailed view.

The functions and settings of the SBC are substantially identical to those of Table business cases. This chapter focuses on the features and the settings that apply only to the single business case.

Demonstration Apparo Fast Edit Anonymous Demonstration

Product details

Product line: Jackets
 Product ID: 170
 Product name: Madox
 Colour: white
 Size: M
 Model: Tower
 Manufacturer: Escada
 Start date: 15.01.2009
 Price history > Price details

Product description

This is a product description.
 Test

OK EXPORT TO EXCEL EXCEL ROW-IMPORT

User view of a SBC, the widgets are visually divided into 2 columns.

6.1 Structure of the SBC

- **Header area** - with the title and description
- **Data area** - where the widgets are arranged in columns
- **Navigation pane** - used to navigate between records and the switch button for the data input mode
- **Button area** - contains the default and user-defined buttons
- **Optional footer area** - for info and graphics

6.2 Arrangement of the widgets in the SBC

The widgets in the SBC can be output with multiple columns.

Editing widgets

Drop here (widgets below will be shift down).		Drop here (widgets below will be shift down).	
<div>+ Drag here</div>		<div>+ Drag here</div>	Drop here to place the widget.
> Spacer & Title	H <input type="checkbox"/>	> PRODUCT_DESCR > Text area > Product description	PK <input type="checkbox"/> RO <input type="checkbox"/> H <input type="checkbox"/> NN <input type="checkbox"/>
Drop here (widgets below will be shift down).			
<div>+ Drag here</div>		<div>+ Drag here</div>	Drop here to place the widget.
> PRODUCT_LINE_ID > Lookup dropdown (for all tables) > Product line	PK <input type="checkbox"/> RO <input checked="" type="checkbox"/> H <input type="checkbox"/> NN <input type="checkbox"/>		
Drop here (widgets below will be shift down).			
<div>+ Drag here</div>		<div>+ Drag here</div>	Drop here to place the widget.
> PRODUCT_ID > Input field > Product ID	PK <input checked="" type="checkbox"/> RO <input checked="" type="checkbox"/> H <input type="checkbox"/> NN <input type="checkbox"/>		
Drop here (widgets below will be shift down).			
<div>+ Drag here</div>		<div>+ Drag here</div>	Drop here to place the widget.
> PRODUCT_NAME FN			

Designer view: The arrangement of the widget by columns and rows

Widgets can be placed easily by drag and drop.

6.2.1 Columns - a users view

Demonstration Apparo Fast Edit Anonymous Demonstration

Product details

	Column 1	Column 2	Column 3
Row 1	Product line: Jackets	Product description: a product description.	
Row 3	Product ID: 170		
Row 5	Product name: Madox		
Row 7	Colour: white		
	Size: M		
	Model: Tower		
	Manufacturer: Escada		
	Start date: 15.01.2000		
	Price history > Price details		

OK EXPORT TO EXCEL EXCEL ROW-IMPORT

User view: Column 2 contains only one text area widget

The terms 'column' and 'line' refer only to the visual presentation and may not be consistent with database columns or rows.

6.3 Visual

Here you define the general optical settings for the single business case.
These settings are different from those in Table Business Case

Target table	Header	Footer	Visual	Colours	Widgets	Row ordering	Link into Portal							
			Label width (px)	120 *										
			Widget width (px)	500 *										
			Visual column label widths	80,,80										
			Visual column widget widths	150,400										
			Gap width between data rows(px)	2 *										
			Hide application header	<input type="checkbox"/>										
			Enable a general button bar for rich text widgets	<input type="checkbox"/>										
			Enable dialog window with error message in case of error	<input checked="" type="checkbox"/>										
			Window background image URL	<input type="text"/>										
			Show just the first data row only	<input type="checkbox"/>										
			No data to display message	<table border="1"> <thead> <tr> <th>Language</th> <th>No data to display message</th> </tr> </thead> <tbody> <tr> <td>German</td> <td>Keine Daten zum Anzeigen</td> </tr> <tr> <td>English</td> <td>No data to display</td> </tr> </tbody> </table>					Language	No data to display message	German	Keine Daten zum Anzeigen	English	No data to display
Language	No data to display message													
German	Keine Daten zum Anzeigen													
English	No data to display													

Options

Label width (px)

Width of the label in pixels

Widget width (px) *

Width of the input area of the widget

Visual column label widths

Defines the visual column label widths. If no value is defined for visual column label width then 'Label Width' property is used as default value; e.g.

100,150,200	3 visual columns with label widths 100 (px), 150 (px) and 200 (px)
100,,200	3 visual columns with label widths 100 (px), 'Label Width' (px) and 200 (px)
,,200	3 visual columns with label widths 'Label Width' (px), 'Label Width' (px) and 200 (px)

IMPORTANT: Negative numbers are not valid.

Visual column widget widths

Defines the visual column widget widths. If no value is defined for visual column widget width then 'Widget Width' property is used as default value; e.g.

100,150,200	3 visual columns with widget widths 100 (px), 150 (px) and 200 (px)
100,,200	3 visual columns with widget widths 100 (px), 'Widget Width' (px) and 200 (px)
,,200	3 visual columns with widget widths 'Widget Width' (px), 'Widget Width' (px) and 200 (px)

IMPORTANT: Negative numbers are not valid

Gap width between data rows(px)

The optical gap between data rows in pixels, default is 2

Hide application header

Hides the header with application name, BC name and user name

Enable a general button bar for rich text widgets

If this feature is enabled then the user is seeing only one button bar with bold, italic, underline etc. buttons for changing the text style in rich text widgets.

This button bar is necessary if widgets of type text area with rich text functionality are used (using bold, italic, underline, different colours). The general button bar is visible like in Microsoft Word. If disabled then each text area with rich text feature has an own button bar.

Enable dialog window with error message in case of error

If this setting is enabled then a pop-up dialog window will be displayed after each error.

Window background image URL

It is possible to use own background picture using an URL.

Show just the first data row only

When enabled only first data row will be displayed; otherwise additional buttons "<<" and ">>" will be displayed to show previous respectively next data row.

No data to display message

Message presented to user when there are no data to display


7 Combining Business Case

There are two main ways to combine Business Cases.

- As Business Case Set
- As master-detail relationship

7.1 Business Case Set

Table and Single Business Cases can be combined in a **Business Case Set**. Single or Table Cases are displayed together so that navigating is made easier. The Business Cases are not inherently connected.


Demonstration Apparo Fast Edit

SAMPLES - product lines
SAMPLES - product list
SAMPLES - product details

Product lines

Simple output of a list.

☐ Product Line *

☒ Bags
☒ Bikinis
☒ Caps
☒ Jackets
☒ Pol
☒ T-Shirts
☒ Trousers
☒ Underwear

SEARCH
RESET FILTERS

Product line ID *	Product line english	Count of products
<input type="checkbox"/> 1	Trousers	1
<input type="checkbox"/> 2	T-Shirts	7
<input type="checkbox"/> 3	Pol	2
<input type="checkbox"/> 4	Jackets	8
<input type="checkbox"/> 14		0

Count of product lines: 8

OK
CANCEL
CLOSE & SAVE
DELETE
INSERT
COPY
EXPORT TO EXCEL
EXCEL FILE IMPORT

7.2 Master-detail relationship

Business Cases can also be combined in a **master-detail relationship**. In this case, the contents of two tables are handled together. The database transactions are performed on both tables at the same time.

Example for a Master/Detail relationship
One dealer has many products.

Dealer Name	City	Product list
France	Bordeaux	Product list
Germany	Berlin	Product list
Germany	Hamburg	Product list
France	Paris	Product list

Page: 1/1 OK Cancel Close

ProductList - Windows Internet Explorer
ProductList
Administrator

Example for a Master/Detail relationship
One dealer has many products.

All products of dealer France

City	Detail number	Product
Bordeaux	11.00	Erwin Nathan
Bordeaux	12.00	Erwin Nathan
Bordeaux	30.00	Fast Info
Bordeaux	15.00	Fast SSB
Bordeaux	21.00	Erwin Document
Bordeaux	26.00	Erwin planning
Bordeaux	110.00	Fast SSB
Bordeaux	110.00	Fast Security
Bordeaux	45.00	Erwin Nathan
Hamburg		
Paris		

Page: 1/1 OK Insert Excel Row-Import Delete Cancel Close Copy

Done Local intranet 100%

Both combinations are explained more detailed later in this tutorial.

7.3 Primary Keys

- A primary key identifies a unique row in a table.

A typical example is a unique product ID:

01			
02			
03			
04			

- It can either be a single primary key (one attribute) or a compound key consisting of several primary keys (more than one attribute).

An example for a compound key would be a supplier-product relationship. Each supplier offers one or more products, so the supplier ID alone is not enough to identify the row:

01		a	
01		b	
01		c	
02		m	

- When creating a Business Case, the number of primary keys has to be set to the minimum necessary to identify every unique row.
- The primary keys in a Fast Edit Business Case do not depend on the underlying database table design. However, the primary keys of the table are usually also used for the Business Cases.

Each column (also many columns at the same time) can be used as the primary key(s).

In Fast Edit you can define a completely different primary key than in the underlying table (the same applies to any null/not null-definitions). In the Insert and Update cases, only the primary key definition from Fast Edit will be taken into consideration.

However, in about 90% of the Business Cases, the same primary keys as in the underlying table are used.

The primary key of the database table column is important for:

- The definition of a lookup table for the **Lookup Multi-select** widget (field "Lookup table key column").
- The widget **Business Case Link**, with which the two tables of the two Business Cases can be edited in one database transaction.

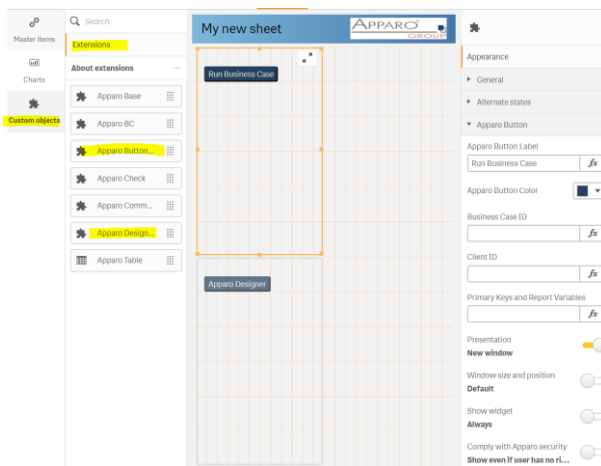
8 Linking Apparo Fast Edit and Qlik Sense together

8.1 Usage of the Apparo Designer & Apparo BC Button extensions

Both buttons can be drawn from the left bar with the mouse into the spreadsheet.

The Apparo Designer button has no further options and serves to open the Apparo Designer only, if the respective user has appropriate access rights.

The Apparo BC button on the other hand is used to open a specific Business Case and requires the setting of other options on the right side under “Appearance”.



Business Case ID

Here you add the ID of the Business Case you want to open

Client ID

Optionally, enter the ID of the client here, if the same Business Case ID exists in several clients

Additional Parameters

Here additional parameters can be transferred.

&p1= is the first primary key. The Business Case would be filtered accordingly in this case.

&FE_Variable_name= would not filter, but pass a value to a report variable of this name.

Apparo Button Label

Here you can enter an identifier for the button

Presentation

With the options open in a new window or open in a tab

Window size and position

Sets the size and position of the browser opened window

8.2 Embedding Business Case into the Qlik Sense App

You can use the “Apparo Business Case Extension” for that:

Target table | Header | Footer | Visual | Colours | Widgets | Row ordering | **Linking to Qlik Sense App**

Using the Apparo Business Case Extension

1. Drag Apparo BC extension from the Custom objects panel to the sheet.
2. Appearance settings will be displayed on the right side. Please expand nested Apparo Business Case tab and fill following parameters in:

Business Case ID:	SAMPL WF WORKFLOW1
Client ID:	Demo
Primary Keys and Report Variables:	<pre>=&p1=' & [REPLACE WITH PK 1] &'&p2=' & [REPLACE WITH PK 2] &'&p3=' & [REPLACE WITH PK 3] &'&p4=' & [REPLACE WITH PK 4] &'&p5=' & [REPLACE WITH PK 5] &'&p6=' & [REPLACE WITH PK 6] &'&FE_SelectedProductLines=' & GetFieldSelections([REPLACE WITH VALUE FOR THE APPARO VARIABLE SelectedProductLines], ' ', 999)</pre>

[Click here to see a movie about Qlik Sense filters and Business Cases](#)

Add the extension “Apparo BC” into your sheet:

Master items | Search | Extensions | About extensions | ...

Apparo Base

Apparo BC

Apparo Button...

Apparo Check

Apparo Comm...

Apparo Design...

Apparo Table

My new sheet

Create

Drag items from the library panel to the sheet

Search | Extensions | About extensions | ...

Apparo Base

Apparo BC

Apparo Button...

Apparo Check

Apparo Comm...

Apparo Design...

Apparo Table

My new sheet

APPARO GROUP

No Business Case ID was provided.

Appearance

- General
- Alternate states
- Apparo Business Case
 - Attention: Textarea widget with HTML is not supported.
 - Business Case ID: fx
 - Client ID: fx
 - Primary Keys and Report Variables: fx
 - Refresh extension on resize: ☐

Hide properties

8.3 Business Case and the standard Qlik Sense Table

The integration of Apparo Fast Edit Business Cases in Qlik Sense Table widget is achieved via hyperlinks. In the tab "Linking to Qlik Sense App", you will find the entry for the URL for calling the Business Case.

Copy the appropriate entry.

Target table	Header	Footer	Visual	Colours	Widgets	Row ordering	Features
Linking to Qlik Sense App							
<ul style="list-style-type: none"> Using the Apparo Business Case Extension Using standard Qlik Sense table 							
<p>Expression for opening this Business Case from a standard Qlik Sense Table:</p> <pre>= "https://qsdemonstration/extensions/ApparoBase/apparoBusinessCase.html?bc=SAMPL%20WF%20WORKFLOW1&clientid=Demo&qsapp=' & DocumentName() & 'p1=' & [REPLACE WITH PRIMARY KEY 1] & 'p2=' & [REPLACE WITH PRIMARY KEY 2] & 'p3=' & [REPLACE WITH PRIMARY KEY 3] & 'p4=' & [REPLACE WITH PRIMARY KEY 4] & 'p5=' & [REPLACE WITH PRIMARY KEY 5] & 'p6=' & [REPLACE WITH PRIMARY KEY 6] & 'FE_SelectedProductLines=' & [REPLACE WITH VALUE FOR THE APPARO VARIABLE SelectedProductLines]"</pre>							
<ul style="list-style-type: none"> Using the Apparo Table Extension Using the Apparo Business Case Button Extension Running/refreshing this Business Case from another Business Case using a HTML hyperlink 							

Create or edit a Qlik Sense app and drag with your mouse the item "Table" in a worksheet of your choice. Then add the desired dimensions.

Add a dimension and **unlink the entry**:

Then select in the Representation option **"Link"**.

In the opened formula editor of setting **"Field"** and add the URL that you copied from the Designer.

Example:

The generated URL:

```
= 'https://qsdemonstration/extensions/ApparoBase/apparoBusinessCase.html?bc=SAMPL%20WF%20WORKFLOW1&clientid=Demo&qsap=' &
DocumentName() &
'&p1=' & [REPLACE WITH PRIMARY KEY 1] &
'&p2=' & [REPLACE WITH PRIMARY KEY 2] &
'&p3=' & [REPLACE WITH PRIMARY KEY 3] &
'&p4=' & [REPLACE WITH PRIMARY KEY 4] &
'&p5=' & [REPLACE WITH PRIMARY KEY 5] &
'&p6=' & [REPLACE WITH PRIMARY KEY 6] &
'&FE_SelectedProductLines=' & [REPLACE WITH VALUE FOR THE APPARO VARIABLE SelectedProductLines]
```

Now you can remove unnecessary primary key entries or putting the values into the URL.

```
1 = 'https://qsdemonstration/extensions/ApparoBase/apparoBusinessCase.html?bc=SAMPL%20WF%20WORKFLOW1&clientid=Demo&qsap=' &
2 & '&p1=' & [SAMPLE_PRODUCT.PRODUCT_ID]
3 & '&FE_SelectedProductLines=' & PRODUCT_LINE_NAME
```

Then click on "Apply"

Add the right output expressions into the settings **"Label"** and **"Link label"** too.

If you click now in the final worksheet the hyperlink, the associated Business Case opens in a new tab and displays the entry with the appropriate ID

My new sheet

ID_CAR	COLOUR_ID	Link to Apparo Fast Edit
200	2	Open the Business Case
300	3	Open the Business Case
301	4	Open the Business Case

Apparo Fast Edit Demo
testuser1
QA

Car testing center

Who tested what car?
Please enter test run id, name of the car, driver and the date of the test run
You are working with BC: Demo Variables

Filter for driver

SEARCH
RESET FILTERS

ID	Car	Driver	Tested when?	Change who	Change when
<input type="checkbox"/> 301.00	Demo-Auto1	testuser1	Apr 27, 2016	DRIVER: testuser1	testuser1

301.00
testuser1
Apr 27, 2016
DRIVER: testuser1

Row count: 4
Row count filtered: 1

Page: 1 / 1

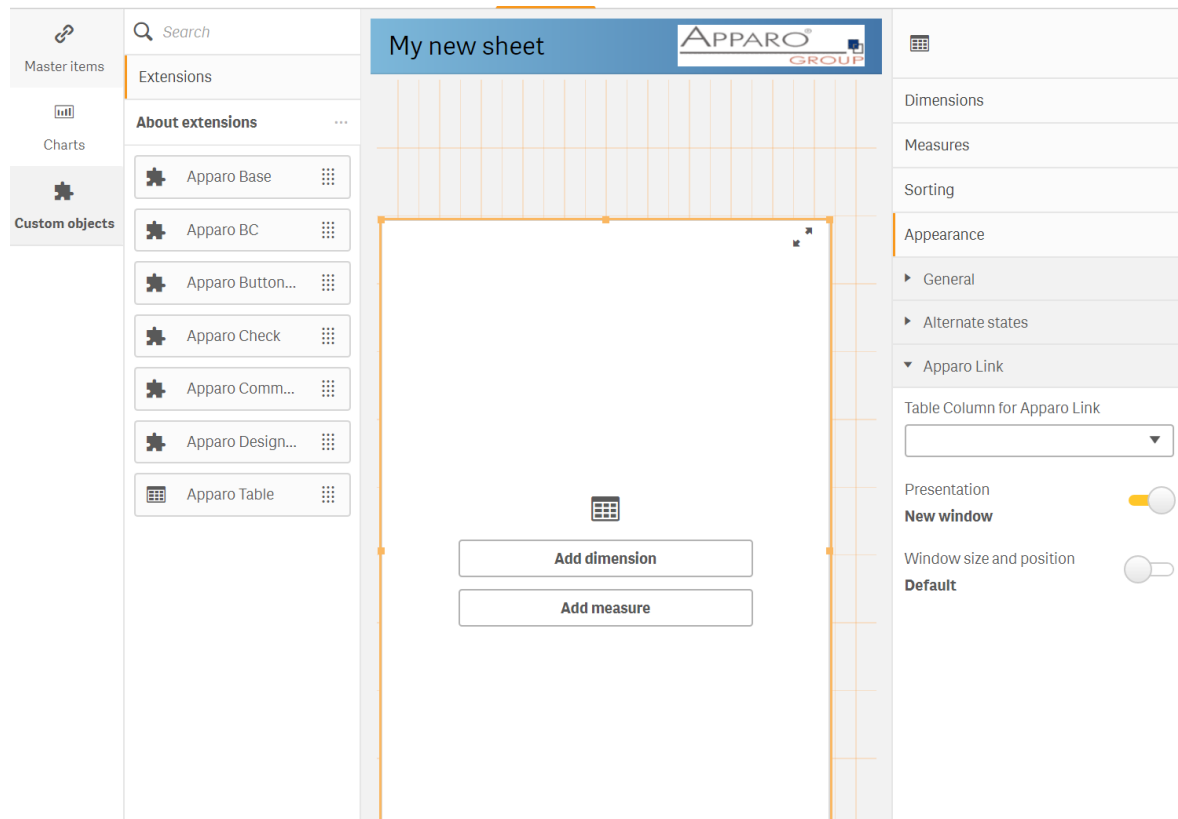
OK
CANCEL
CLOSE
DELETE
INSERT
RELOAD
EXPORT TO EXCEL
EXCEL ROW-IMPORT

8.4 Using the Apparo Table extension

The Apparo Table is used the same way as the standard Qlik Sense table.

The advantage of this extension is the ability to open the Business Case in a separate window.

The disadvantage is, you can use only one column as a primary key column.



The Hyperlink column is defined on the left side under "Appearance":

Apparo Link Column

Defines the column whose value is transferred at "Additional Parameters"

Business Case ID

The ID of the Business Case

Additional Parameters

„&p1=“ is passing the value from "Apparo Link Column" to the first primary key Widget

Apparo Link Text

Naming the Hyperlink

9 Creating database table and columns with the Designer

You can allow your users Designer to create new database tables with the designer or to add columns to existing tables.

This function allows only simple database tables and the following basic data types:

- Number
- Text
- Date

9.1 Activation of the feature in the client settings

This feature is disabled by default and must be enabled in the client. Optionally, you can restrict this function to certain security groups.

Demonstration Apparo Fast Edit Settings of client Client2_Europe Anonymous Demonstration

Client head General Languages Default numeric & datetime formats Access rights Automatic table/column creation Excel export formats Business Log

Portal Colours of portal and designer Business case standard style

If this feature is enabled then a Business Case designer can create database tables and columns.

[Click here to watch the video guide](#)

☒ Automatic table/column creation feature is disabled for everyone

☐ Automatic table/column creation feature is enabled for everyone

☐ Automatic table/column creation feature is enabled for users with specified security groups

OK CANCEL

Show desc

9.2 Create the required database connections

As mentioned in Chapter, 'Clients', you will need two different database connections.

Database connections									
Database connections									
<input type="checkbox"/>	Name	Usage	Database type	Host	Port	Database name/alias	User	Database schema	Actions
<input type="checkbox"/>	Erste	reading/writing only (DML)	Oracle	demo.clupgnkmc25.eu-west-1.rds.amazonaws.com	1521	demo	erste	ERSTE	i
<input type="checkbox"/>	Erste DDL	creating tables/columns (DDL)	Oracle	demo.clupgnkmc25.eu-west-1.rds.amazonaws.com	1521	demo	erste	ERSTE	i

9.3 Create a database connection to create tables / columns (DDL)

This database connection type also requires, in addition to read and write, the right to create tables.

First enter the required connection data:

The screenshot shows the 'Main' tab of a configuration window. It has five tabs: 'Main', 'Advanced', 'Variables', 'Automatic tables/columns creation', and 'Security'. The 'Connection name' field contains 'Erste DDL'. Below it, a note states: 'The database connection name is the name that will be used in all Business Cases.' The 'Database type' dropdown menu is open, displaying a list of database types: Exasol, IBM DB2, IBM DB2 Client, IBM DB2 i, IBM dashDB, IBM Netezza, Informix, MS SQL Server 2008-2019, Oracle (highlighted), Oracle (using service name), Oracle Client, PostgreSQL, SAP HANA, SAP Sybase ASE, SAP Sybase IQ, SAP Sybase SQL Anywhere, Teradata, and MySQL. The 'Database host' field contains 'demo.clupgnkmrc25.eu-west-1.rds.amazonaws.com'.

Then select in the tab “Automatic table / columns creation” the third radio button:

This database connection type is used for creating new database tables/columns only (cannot access data in Business Cases)

The screenshot shows the 'Automatic tables/columns creation' tab. It contains a heading: 'Here you can define the type of this database connection:'. Below this are three types: 'Type 1: It can read and write data only', 'Type 2: It combines type 1 and 3 and is used to create DB-tables or DB-columns in the Designer. Needs a predefined type 3 connection.', and 'Type 3: It can create new database tables and columns.' There is a link 'Click here to watch the video guide'. Three radio buttons are present: the first is unselected, the second is unselected, and the third is selected. Below the radio buttons are two text areas for templates. The first is labeled 'Template for create table statement' and contains the text: 'CREATE TABLE <%SCHEMA_NAME%>.<%TABLE_NAME%> (<%COLUMN_DEFINITIONS%>)'. The second is labeled 'Template for create index statement' and contains the text: 'CREATE INDEX ix_<%TABLE_NAME%>_<%COLUMN_NAME%> ON <%SCHEMA_NAME%>.<%TABLE_NAME%> (<%COLUMN_NAME%>)'. At the bottom are 'OK' and 'CANCEL' buttons.

It is not possible to create with this DB connection Business Cases.

9.4 Creating and linking the DB connection type ,read / write only (DML) '

After you have added the necessary connection information, select the in tab ,Automatic table / column creation' the option ,This database connection type combines both types and should be used if users shall be able to auto create tables/columns (needs a predefined connection for creating new tables/columns)'

Then, assign the previously created DDL connection here.

Here you can define the type of this database connection:
 Type 1: It can read and write data only
 Type 2: It combines type 1 and 3 and is used to create DB-tables or DB-columns in the Designer. Needs a predefined type 3 connection.
 Type 3: It can create new database tables and columns.

[Click here to watch the video guide](#)

☐ This database connection type is useable for creating/editing Business Cases (reading and writing of data in existing tables) only

☒ This database connection type combines both types and should be used if users shall be able to auto create tables/columns (needs a predefined connection for creating new tables/columns)

associated table/columns creating connection: Erste DDL

☐ This database connection type is used for creating new database tables/columns only (cannot access data in Business Cases)

OK CANCEL

This completes the setup of the database connection.

9.5 Create a new database table

To create a new table when creating a new Business Case, select <Create new data storing table> in the setting ,Target database table/view'

Main database Business Case settings

Identifier / Short name ID

Business Case name BC NAME

Target database table/view <Create new data storing table>

Notes

APPROVAL_STATUS
 APPROVER
 Lieferant
 PROCUREMENT
 PROCUREMENT_OFFER
 SAMLE_OFFICE_WORKERS
 SAMPLE_TASKS

NEXT CANCEL

Afterwards please click on NEXT.

Then you will see a blank widget overview page.

The table will not appear until you create the first Edit widget

First, select the widget type

And then open the tab ,Mapping and Other' and select in the setting column name '<Create new data storing column>'

Widget settings of database column

Widget settings of database column

Widget type Mapping & Other Features Visual Help texts Data output format

Column name

Default value

Constant value

Variable for using content in detail BC

OK CANCEL

Then a window will appear, in which you specify the table name and the column names and types.

Defining the name of new data storing table and data storing column

Data Storing Table Name max. 18 characters

Data storing column name max. 18 characters

Data storing column type

Max. allowed amount of characters

Text
Long text
Number
Date
Date & Time

CREATE CANCEL

When creating further edit widgets then you will only be asked for the name and type for the new database column.

10 E-mail Business Cases (EBC)

An e-mail Business Case is used to send e-mails.

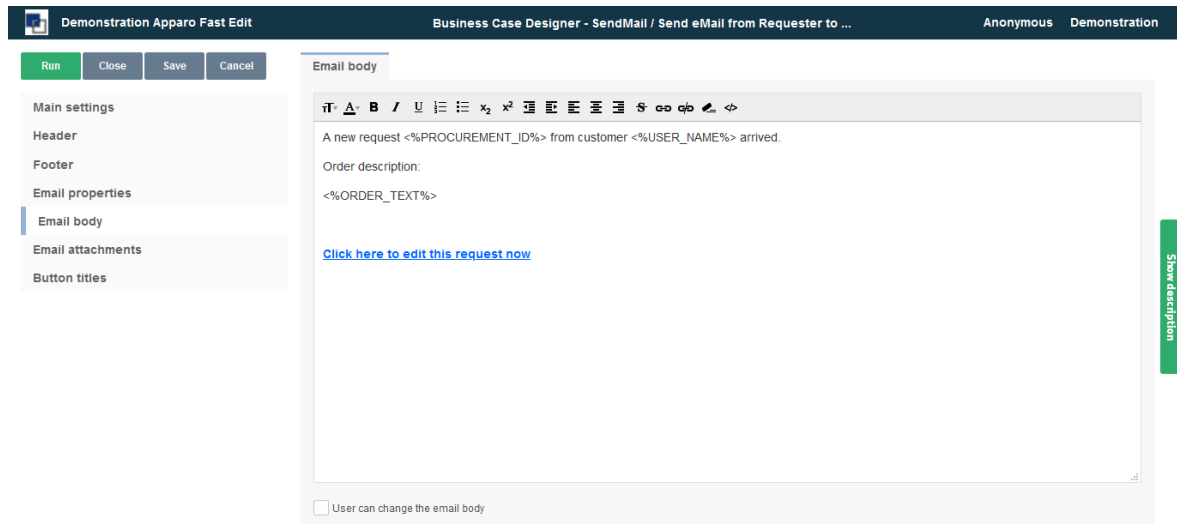
It contains the definitions, such as subject and body.

Content, recipient, etc. can be made dynamic with variables.

An e-mail business case is called usually by button from Single or Table business cases.

An e-mail Business Case can access all the widget reference variables of the current line.

All other variables can also be used.



EBC with backlink to a Business Case

10.1 Creating an EBC

When you create an e-mail business cases you have to fill first, like all other types of business case, the general settings.

The email connection is used only to send and can also be used in other e-mail business cases.

The optional security group ensures that only authorized users can send e-mails.

A number of security groups are to be entered separated by a comma.

Demonstration Apparo Fast Edit Business Case Designer - SendMail / Send eMail from Requester to ...

Run Close Save Cancel

Main settings

Header

Footer

Email properties

Email body

Email attachments

Button titles

Main settings

Identifier / Short name SendMail

Business Case name Send eMail from Requester to Approver

Email connection EMail

Business Case security group

Notes Send eMail from Requester to Approver

10.2 Header and Footer

In the header or footer, you can define captions and descriptions, specify fonts and styles and insert logos. In the title, the description and the logo URL variables can be used.

Demonstration Apparo Fast Edit Business Case Designer - SendMail / Send eMail from Requester to ... Anonymous Demonstration

Run Close Save Cancel

Main settings

Header

Footer

Email properties

Email body

Email attachments

Button titles

Header

Title & Description

Language	Title	Description
German		
English		

Title style

Font face	Size	Style	Align	Colour
Arial	14	Bold	Left	#000000

Description style

Font face	Size	Style	Align	Colour
Arial	12	Normal	Left	#000000

Background colour #BEE5FA

Left logo URL

Right logo URL

Show description

10.3 E-mail properties

Here you can define the sender e-mail, the recipient list and their settings.

Sender & Recipients

Sender email address

Does the indicated in the e-mail sender address this need not match the e-mail sender from the e-mail link. Variables can be used.

Options:

- Try to use automatically the email address of the user if the email is stored in the security system
- Users can change the sender address

Recipient(s)

Contains all recipients, separated by commas.
Variables can be used.

Optionally, the user can modify the list.

Subject

Contains the subject of the e-mail, variables can be used.
Optional user may change the subject.

Settings

Defines the size of the text area for the e-mail text (visible if the user is allowed to change the e-mail text).

- Widget Width (px)
- Label Width (px)

10.4 E-Mail body

Contains the 'E-mail Body', also known as e-mail text.
If you use formatted text, the email HTML format is used.
You can use all the variables of the calling business cases.

Email body

A new request <%PROCUREMENT_ID%> from customer <%USER_NAME%> arrived.

Order description:

<%ORDER_TEXT%>

[Click here to edit this request now](#)

☐ User can change the email body

10.5 E-mail attachments

Here you can define email attachments. You have 3 possibilities:

If your file is stored on server-side file system you have to click on 'Add new file attachment' button and choose option 'Using file stored on the server file system'.

Absolute paths must be used, e.g. 'C:\Files\<%USER_NAME%\report1.pdf' or '<%CALCULATED_FILENAME%>'.

Note: If the Email Business Case is started from Table Business Case (by email button) then you can use variables from that Table Business Case too.

If the Email Business Case is run from Table Business Case then File Widget reference variable can be used too: click on 'Add new file attachment' button and choose option 'Using file stored in the file-widget'. e.g. <%WIDGET_EMPLOYEE_CV%>. This type of attachment is not supported if the Email Business Case is run using script function afe.runEmailBc.

By option, the user can attach any file.

Email attachments

Here you can define email attachments. You have 2 possibilities:

- If your file is stored on server-side file system you have to click on 'Add new file attachment' button and choose option 'Using file stored on the server file system'. Absolute paths must be used, e.g. 'C:\Files\<%USER_NAME%\report1.pdf' or '<%CALCULATED_FILENAME%>'. note: If the Email Business Case is started from Table Business Case (by email button) then you can use variables from that Table Business Case too.
- If the Email Business Case is run from Table Business Case then File Widget reference variable can be used too: click on 'Add new file attachment' button and choose option 'Using file stored in the file-widget'. e.g. <%WIDGET_EMPLOYEE_CV%>. This type of attachment is not supported if the Email Business Case is run using script function afe.runEmailBc.

Order	Name	Actions
No e-mail attachments defined		

ADD NEW FILE ATTACHMENT

☐ User can add own attachments when Email Business Case is running

10.6 Button titles

Contains the label of the buttons in all installed languages

Button titles			
Language	Add email attachment button	Send email button	Don't send email button
German	Datei anhängen	E-mail senden <input type="checkbox"/>	Abbrechen <input type="checkbox"/>
English	Attach file	Send eMail <input type="checkbox"/>	Cancel <input type="checkbox"/>

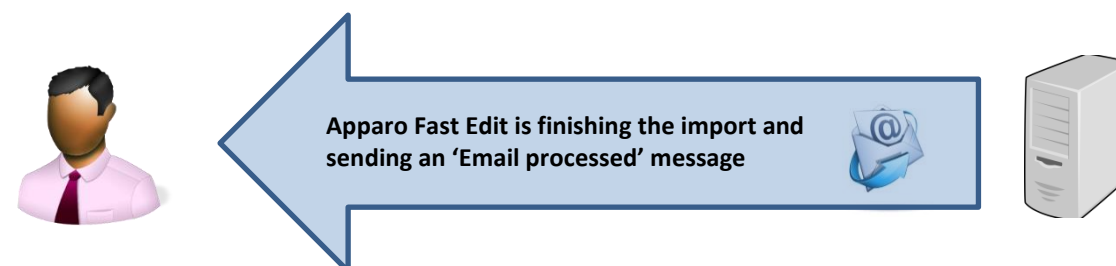
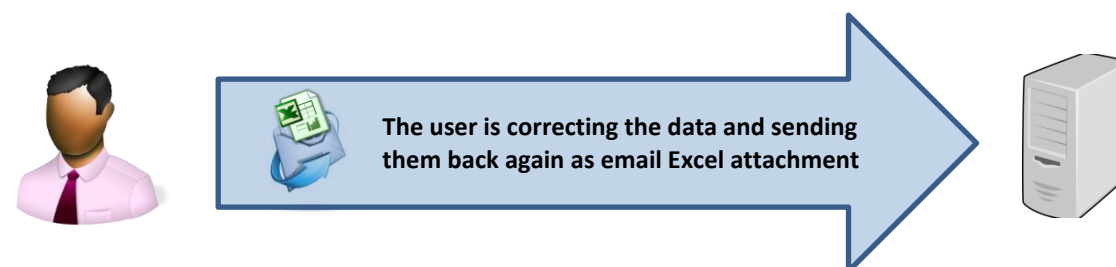
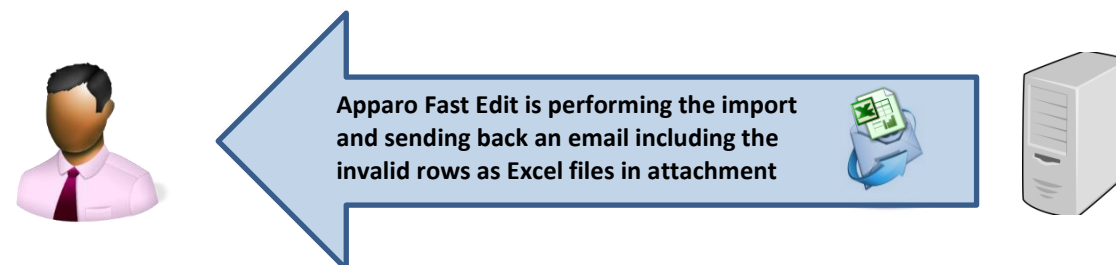
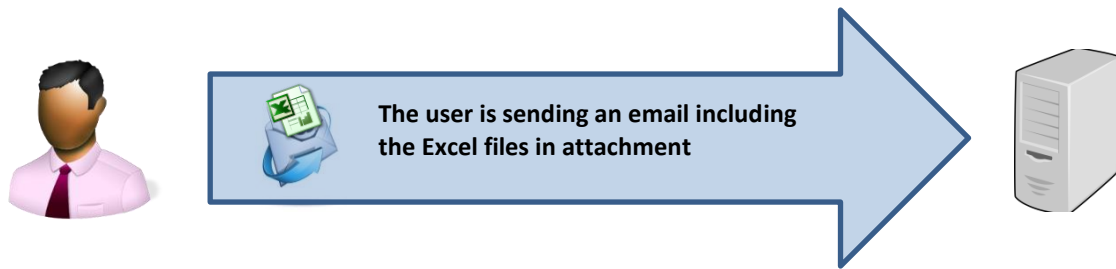
11 E-Mail Import Business Case (EIBC)

The Excel Email Import feature enables you to import data that is stored in Excel files (CSV, XLS and XLSX) from email attachments.

That means the user can send an email with **Excel files in attachment and the data of these Excel files will be imported automatically in your relational databases.**


The user is getting automatically answer-emails that are informing the user about the progress or data quality issues.

All activities can be logged in database table, the emails and attachments can be stored physically on the server.









11.1 Creating a new Business Case of Type 'Email Import'

When clicking on New Business Case in the Business Case list the following selection list will appear:


Demonstration Apparo Fast Edit
Anon

Please select type of Business Case you want to create now

	Table	A table Business Case is showing many data rows on the same page. The user can filter the data, edit, import from Excel, export to Excel and so on.
	Single	A single Business Case is showing just one data row only.
	Set	A grouping of multiple Business Cases (table/single) for more comfortable usage. You can define global filters that are filtering all Business Cases automatically too.
	Email import	Importing Excel data directly by email - send Excel sheets using email attachments and Apparo will import the Excel data directly into the database including file uploads. No web browser is necessary, just an email.
	Email	An eMail Business Case is a definition of an email text including usage behavior and can be used in another Business Cases of type 'table' or 'single' only. In these Business Cases it is possible to define buttons that can use this eMail Business Case.
	Action	Purpose of Action Business Case is to execute scripts or database procedures that can be called from a report/HTML page. Usage of AJAX and Javascript for automatically executing in the background is possible too.

CANCEL

Click on Email Import to create a new 'Email Import Business Case'

For the setup we will need a pre-defined email connection and at least one database connection, used for the Business Case that will perform the import.

These Business Cases are also containing all definitions for securing the data quality.

The Business Cases must have activated the Excel file import feature.

11.1.1 New Business Case - Main Settings

The main settings require the following settings:

- **Identifier:** The short name of the Business Case (must be unique)
- **Business Case Name:** This name will appear as name when we link the Business Case to the portal
- **Email connection:** The email connection for sending and receiving emails
- **Internal description:** Optional. For documentation purposes only.



Demonstration Apparo Fast Edit

Email Import Business Case (EIBC) - Main settings

Identifier / Short name	<input type="text"/>	*
Business Case name	<input type="text"/>	*
Email connection	<input type="text" value="Select email connection"/>	▼ *
Enabled	<input checked="" type="checkbox"/>	
Notes	<input type="text"/>	

NEXT
CANCEL

Fill all necessary fields and click 'Next' to create the Business Case

11.2 Overview of all possible settings

Once the Business Case is created we will see the following overview.

Here you can save and close the Business Case and click through the tabs of the settings:

- **Introduction:** Contains usage examples and explanations
- **Main Settings:** Contains the main settings and the server path for storing emails and attachments
- **Importing Groups:** Contains the import groups, the detailed settings how shall be imported
- **eMails:** Here you can define the text of a failure email, for the case that no import group is matching
- **Logging:** Contains the logging settings, details like user name can be mapped here to a database column
- **Variables:** Contains predefined variables and you can create own JavaScript variables

The screenshot shows the 'Business Case Designer' interface. The title bar reads 'Demonstration Apparo Fast Edit' and 'Business Case Designer - Import of Excel data using eMail / Impor...'. Below the title bar are 'Close', 'Save', and 'Cancel' buttons. On the left is a sidebar with tabs: 'Introduction', 'Main Settings' (selected), 'Import Groups', 'eMails', 'Logging', and 'Variables'. The main area has two sub-tabs: 'Main' and 'Email file system storage'. Under 'Email file system storage', there are fields for 'Identifier / Short name' (value: 'Import of Excel data using eMail'), 'Business Case name' (value: 'Import of Excel data using eMail'), 'Email connection' (value: 'EMail'), 'Enabled' (checked checkbox), and 'Notes' (text area with 'showing example').

11.3 Main Settings

This tab contains, beside from the main settings, the path for storing emails and attachments physically on the server.

The screenshot shows the 'Business Case Designer' interface. The title bar reads 'Demonstration Apparo Fast Edit' and 'Business Case Designer - Import of Excel data using eMail / Impo...'. Below the title bar are 'Close', 'Save', and 'Cancel' buttons. On the left is a sidebar with tabs: 'Introduction', 'Main Settings' (selected), and 'Import Groups'. The main area has two sub-tabs: 'Main' and 'Email file system storage'. Under 'Email file system storage', there is a field for 'Path for storing the email content' with the value '<%AFE_HOME_DIR%>\emails'.

email file system storage

11.4 Importing Groups

An import group contains the definitions what attachments are expected and what Business Case shall perform the import. It also contains the security settings, log settings and the response email texts.

Adding a new import group

11.5 Importing group settings

11.5.1 Main group settings

It contains the import group name (unique) and an optional description text. You can enable or disable the import group here.

Email import group settings

Main group settings
Business Cases
Email texts
Security

Import group name

Excel import *

Description

Enabled

☒

Error handling strategy

Rollback import if there are invalid rows ▼

Error list file format

XLSX ▼

Language

English ▼ *

OK

CANCEL

11.5.2 Business Cases

Contains the mapping to all Business Cases that are defined to perform the import. When an email from a valid sender arrives, Apparo Fast Edit automatically analyses the structure of the attachments and comparing the structure with the defined import groups.

Demonstration Apparo Fast Edit Business Case Designer - Import of Excel data using eMail / Import... Anonymous Demonstration

Email import group settings

Main group settings Business Cases Email texts Security

[ADD NEW BUSINESS CASE](#)

Please notice, that an email, in order to be accepted by this import group, all its attachments (excluding images) must be mapped to the defined table Business Case files. For example if the email contains a text file attachment that stays unused, the email will not be accepted by the import group.

Table Business Cases to be executed in following order					
Order number	Change order	Table Business Case identifier	Notes	Excel file attachments	Actions
1	⬇ ⬆ ⬇	> excel data mapping		> *.excel	✕

[OK](#) [CANCEL](#)

11.5.3 Add new attachment

You need to define at least one email attachment for every defined importing Business Case.

Demonstration Apparo Fast Edit Business Case Designer - Import of Excel data using eMail / Import... Anonymous Demonstration

Email import settings for table Business Case

Table Business Case

Available

- Advanced auditing logging
- Angebote einsehen
- Angebote zu Anforderung hinzufügen
- Approval Status
- Business log of a client
- Data hierarchy example
- HANDS ON
- List of my procurement requests
- Liste aller Bestellanforderungen
- My_1st_Table_BC
- Overview
- PLAN_ART

Selected

excel data mapping

☐ Show just Business Cases of current folder only
☐ Show folder path

Notes

[ADD NEW FILE ATTACHMENT](#)

Email attachment file list to be read in following order				
Order number	Change order	File name pattern	Description	Actions
1	⬇ ⬆ ⬇	> *.excel	2 columns	✕

Creating a new file attachment

Email file attachment settings - Excel file

Basic settings **Advanced Excel document data constraints**

Please note, that only the first sheet of an excel file will be imported.

File name pattern: *.excel *

Description:

Min. and max. occurrences: 1 - 1 *

Language: English ▼

The following properties are expected:

- **File name pattern:** Defines the allowed file extension (*.excel allows all Excel files: .xls, .xlsx, .csv)
- **Description:** For the internal documentation
- **Starting row:** For the case the contains a header in row 1, we start the import in row 2
- **Min and Max occurrences:** The minimum should be at least 1 – the user gets an error email if the attachment contains less attachments than expected
- **Language:** Important for language sensitive data types e.g. date

11.5.4 Advanced Excel document data constraints

This feature is optional:

Here you can define the expected data column types, this feature allows Apparo Fast Edit to better distinguish similar Excel file attachments.

Email file attachment settings - Excel file

Basic settings **Advanced Excel document data constraints**

Starting row: 2 *

Expected row and column counts

Number of data columns:

Minimum row count:

Maximum row count:

Expected data column types

This functionality is not supported for CSV files.

Column name (eg. A, B or AA): A

Column type: Text ▼

Short description:

ADD COLUMN TYPE

Column name (eg. A, B or AA)	Column type	Short description	Actions
No column definitions found			

Expected data column types

11.5.5 Email texts

Contain the bodies of different auto response emails.
Optional. When empty, no email will be sent.

There different kinds of response emails:

- **'Matching email import group found'**: Sent when email received
- **'Confirmation email'**: Sent when confirmation by user is necessary
- **'Security constraints not met'**: Sender does have the required rights for the import
- **'Email processing cancelled because of error'**: Sent in case of data errors and the import is set to 'Cancel the import in case of errors'
- **'Errors occurred, but import was performed'**: Sent when the import is finished with errors
- **'Email successfully imported'**
- **'Limited access prevented email processing'**: The feature 'limited access' is activated and prevents the import
- **'Error list'**: Email with file attachment containing all erroneous rows

Email import group settings

Main group settings	Business Cases	Email texts	Security
Matching email import group found			
Email subject	[Data Import] <%ORIG_EMAIL_SUBJECT%> / Ticket <%IMPORT_TICKET_ID%> / Info - email received V		
Email body	<p>This is an automatically generated email by Apparo Fast Edit. Ticket number: <%IMPORT_TICKET_ID%></p> <p>Data structure of your email attachments is correct. They are going to be imported now.</p> <p>You will receive additional emails informing you about the import progress.</p>		
Email successfully imported			
Email subject	[Data Import] <%ORIG_EMAIL_SUBJECT%> / Ticket <%IMPORT_TICKET_ID%> / Success - all files from your email have been V		
Email body	<p>This is an automatically generated email by Apparo Fast Edit. Ticket number: <%IMPORT_TICKET_ID%></p> <p>Your email attachments were imported completely successfully.</p> <p>The import is finished now.</p>		
Security constraints not met			
Email subject	[Data Import] <%ORIG_EMAIL_SUBJECT%> / Ticket <%IMPORT_TICKET_ID%> / Error - access denied V		
Email body	This is an automatically generated email by Apparo Fast Edit. V		

Auto response email texts

11.5.6 Security


The email import can be secured:

- by limiting the allowed email senders (as list of email addresses)
- by limiting the email senders based on a security group:
 - the user account including email address must be stored in an MS Active Directory system
- by using a text keyword that must be delivered in the subject or body of the email
- by enabling a confirmation email (an automated email is returned to the sender, which has to be confirmed within a defined timeframe)
- by a list of trusted email servers (only emails of listed servers are accepted)

All emails can be encrypted using SSL

The general access can be restricted by using the limited access feature in the tab 'Security':

- **No limitations:** Default value, no restrictions
- **Limited for all:** Nobody can use this import group
- **Limited for variable value:** Not useable if a variable return 'true' – e.g. a variable returns true during the time period when the database is performing maintenance tasks


 Demonstration Apparo Fast Edit

Business Case Designer - Import of Excel data using eMail / Impor...

Email import group settings

Main group settings

Business Cases

Email texts

Security

Allowed email sender addresses

whitelist@company.com

Security keywords

dsfWERT%\$"

Email confirmation required

☒

Confirmation reply must come within

15 minutes

Check if the email address of the sender is defined in the local security system

☒

Authorized security groups

Business Case limited access

☒ No limitation (default)

☐ Limited for all

☐ Limited for variable value

OK


CANCEL

11.6 eMails

It contains the general error message for the case that no matching import group could be found to perform the import.

This can have different causes:

- Erroneous setup of import groups
- Erroneous attachments (e.g. file does not match the file import template)
- The import group can be temporary disabled by the administrator
- disabled by a variable (e.g. a time controlled variable to avoid issues during a maintenance period)


Demonstration Apparo Fast Edit

Business Case Designer - Import of Excel data using eMail / Impor...

Close

Save

Cancel

Introduction

Main Settings

Import Groups

eMails

Logging

Variables

eMails

No matching import group found for email

Email subject [Data Import] <%ORIG_EMAIL_SUBJECT%> / Ticket <%IMPORT_TICKET_ID%> / Error - no matching profile found

Email body

This is an automatically generated email by Apparo Fast Edit.
Ticket number: <%IMPORT_TICKET_ID%>

You have sent an email to the Apparo Fast Edit system.

The system analyzed your email attachments but the structure of your Excel sheets or the number of attachments are not assignable to the defined import possibilities.

Therefore the system imported nothing and the import stopped.

Please inform your Apparo designer who is responsible for the email Excel import functionality.

Internal error occurred and processing stopped

Email subject [Data Import] <%ORIG_EMAIL_SUBJECT%> / Ticket <%IMPORT_TICKET_ID%> / Error - Internal error occurred

Email body

This is an automatically generated email by Apparo Fast Edit.
Ticket number: <%IMPORT_TICKET_ID%>

You have sent an email to the Apparo Fast Edit system.

An internal error has occurred. Please inform your Apparo Fast Edit administrator.

General error messages

11.7 Logging

All events can be logged into an own database table.

In order to log all possible values the table will need the following columns:

- **Column for client name:** What client was used for the import
- **Column for sender address:** What sender address tried to import
- **Column for event timestamp:** Timestamp
- **Column for ticket ID:** Ticket ID, unique ID for the import event
- **Column for storage path:** Where is the email and attachment stored
- **Column for Business Case ID:** What Business Case performed the import
- **Column for importing group name:** What import group performed the import
- **Column for the import message:** Plain text with error message
- **Column for the log severity:** Can be warning, error, info or debug
- **Column for the message code:** A number representing the message

Demonstration Apparo Fast Edit Business Case Designer - Import of Excel data using eMail / Impor...

Close Save Cancel

Introduction
Main Settings
Import Groups
eMails
Logging
Variables

Logging

☒ Write a log to a database table

Email import log settings

Database Connection	Demo_Connection_1
Logging table	AFE_BUSINESS_LOG
Column for the log sequence number	
Column for client name	LOG_SEQUENCE_NUMBER
Column for sender address	MESSAGE_CODE
Column for the event timestamp	
Column for the ticket ID	
Column for storage path	
Column for the Business Case ID	
Column for the import group name	
Column for the import message	
Column for the log severity	
Column for the message code	

Mapping of the database table based log

11.8 Variables

Allows to create own JavaScript variables.
Hint: JavaScript variables can perform SQL too.

There is a list with pre-defined variables, ready to use.

Demonstration Apparo Fast Edit Business Case Designer - Import of Excel data using eMail / Import... Anonymous Demonstratio

Close Save Cancel

Introduction
Main Settings
Import Groups
eMails
Logging
Variables

Variables

+ Add - Delete

User defined variables

Variable name	Variable type
<%Var1%>	Script variable

Internal variables

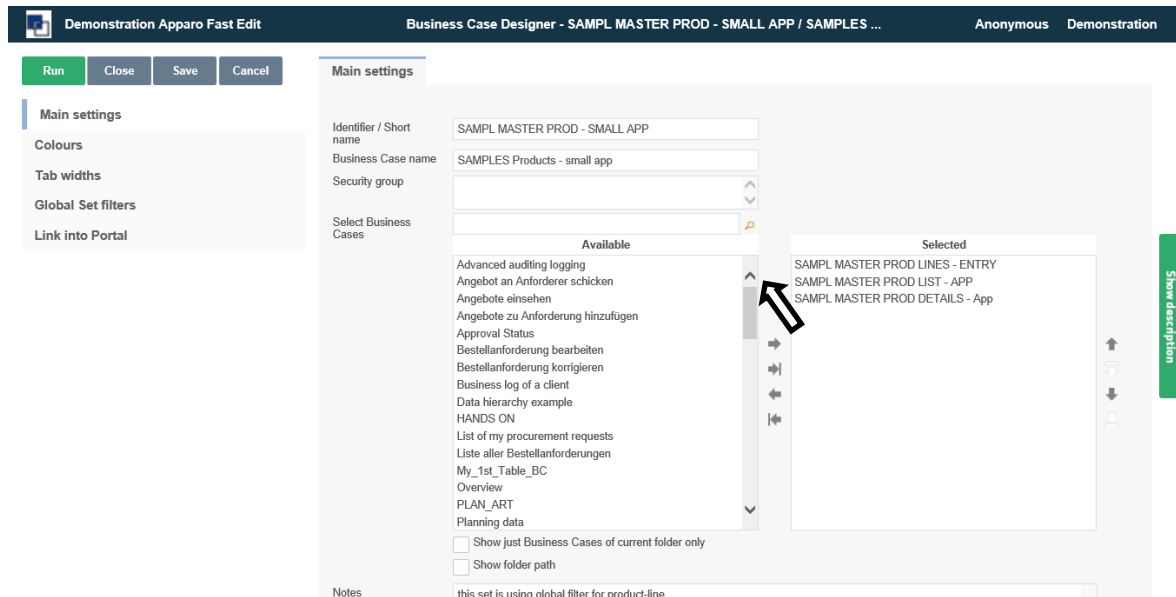
Internal variables ready for use

Variable name	Variable description
<%AFE_HOME_DIR%>	Folder on the server which contains Apparo settings
<%AFE_BC_NAME%>	Name of currently opened Business Case
<%SERVER_NAME%>	Name of application server where Apparo is running
<%NEW_UNIQUE_VALUE%>	Unique value (everytime variable is resolved, its value will be unique)
<%CURRENT_DATE%>	Current date and time
<%DATE%>	Current date
<%TIMESTAMP%>	Current date and time
<%TIME_MS%>	The number of milliseconds since 1.1.1970 (UNIX timestamp)
<%ORIG_EMAIL_SUBJECT%>	Subject of the user's original email when importing data using Email Import Business Case
<%IMPORT_TICKET_ID%>	Ticket number of the current import process when importing data using Email Import Business Case
<%USER_EMAIL%>	Email address (in upper case) of currently logged user
<%USER_NAME%>	Name of currently logged user
<%USER_LOGIN%>	Unique login name of currently logged user
<%MINUTES_FOR_CONFIRMATION%>	Maximum time to wait for the confirmation email
<%ATTACHMENT_NAME%>	Attachment name
<%ERROR_ROW_NUMBER%>	Error row number
<%COLUMN_NAMES%>	Column names
<%ERROR_ROW_DATA%>	Error row data

12 Business Case Sets (Set)

Sets group multiple business cases in a tab view. The business cases can be accessed with tabs and edited comfortable.

12.1 Selection and positioning of business cases in the set (Set)



In 'Available' you find all existing business cases.
By double-clicking or using the arrow keys, these are assigned to the set.







The positioning within the set is also done via arrow keys or the mouse.

By holding down the Ctrl key you select multiple business cases and move it to the desired position.

12.2 Colors

In colors you can set the color of the tabs (tab):

Colours

Inactive tab background colour	#CCCCCC		 *
Inactive tab text colour	#000000		 *
Active tab background colour	#FFFFFF		 *
Active tab text colour	#000000		 *

12.3 Tab Widths

In Table width you define the width of the tabs

Tab widths

Business Case name	Tab width
SAMPLES - product lines	<input type="text" value="200"/>
SAMPLES - product list	<input type="text" value="200"/>
SAMPLES - product details	<input type="text" value="200"/>

12.4 Global Set filters

A global filter is a connection between different filter widgets of different Business Cases of a Business Case Set. That is helpful if some Business Cases of this Set must be filtered in the same way when if the user is jumping to another Business Case.

Example: All Business Cases must filter for the same product and the user is selecting the product just once. It is possible to use many different global filters parallel, e.g. for product and for product-line.

Global Set filters

+ Add

✕ Delete

Global filters

☐ Global filter Name

☐ > prod lines

All existing filter widgets of the Business Cases in the Set are listed here.

To create a global Set filter, move all related filters to ‚Selected filter widgets’ and hit OK.

Global filter

Global filter name

prod lines

Select widgets

Available filter widgets

Selected filter widgets

SAMPL MASTER PROD LIST - APP.product line

SAMPL MASTER PROD DETAILS - App.Product line id

SAMPL MASTER PROD LINES - ENTRY.Product Line

➡

➡

⬅

⬅

OK

CANCEL

12.5 Link into Portal (Standalone version only)

Pick the folder and click OK

Link into Portal

Select portal folder

Demonstration

- 01. Basics
- 02. Workflow
- 03. Logging and auditing
- 04. Data quality checks
- 05. Conditional formatting
- 06. Security
- 07. Excel Import&Export
- 08. Variables
- 09. ActionBC with output
- 10. Small data driven apps
- 11. (Automatic) script & procedure calls
- 12. Creating DB tables and columns
- 13. Dynamic Forms
- 14. EIBC
- 15. Procurement demo
- 16. Smartphone

Main

Advanced

Tiles

Name

B I U T+ tT- T

SAMPLES Products - small app

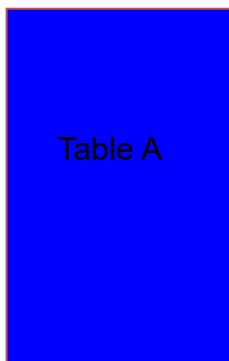
13 Linking BC as Master-Detail connection

In the context of relational databases, a master-detail relationship is the same construction as using a foreign key. (See also the Wikipedia entry http://en.wikipedia.org/wiki/Foreign_key.)

13.1 Master-Detail basics

- Usually, each Business Case refers to one table/view.
- The master-detail-relationship is a way to combine the tables of the Business Cases. It uses the “foreign key” concept of relational databases.
- All user changes in the connected Business Cases are transmitted in one database transaction.

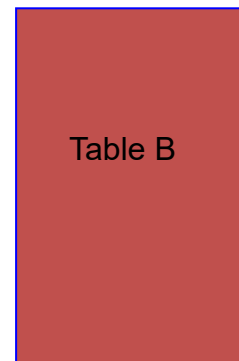
Master BC



Business Case Link widget

*At least one primary key of both tables
needs to be connected.*

Details BC



Example:

- The referenced master table includes an invoice header with a unique **InvoiceNumber**.
- The referencing detail table includes each **InvoicePosition** (including the referred *InvoiceNumber* as foreign key).

Invoice (**InvoiceNumber**, Date...)

InvoicePositions (**PositionNumber**, *InvoiceNumber*, ...)

The advantage of using a master-detail relationship in Fast Edit as compared to normal database programming is that the logic behind the connection is handled by Fast Edit. You don't have to deal with the construction within the detail table itself, as the mapping is all the information necessary for Fast Edit.

A master table may have many referencing detail tables. However, in the context of Fast Edit, a new Business Case must be created for each master-detail relationship.

The general procedure is as follows:

- The master Business Case needs a **Business Case Link** widget. This widget is the connection to the detail Business Case, by mapping the primary key(s). Note that usually only one of the primary keys needs to be mapped, as this is enough to connect the two Business Cases.
- The detail Business Case needs no special widget. It is only connected to the master Business Case by the mapping.

Once the two Business Cases are connected, entering data in the detail table results in one database transaction for master and detail table.

It is recommended that for the connected column(s), you assign the same column names in both tables. You can only map to widgets with “primary key” tag in the linked Business Case.

The optical appearance and title text of the Business Case link is defined on the “Visual” register of the widget.

14 Optimizing Business Cases

14.1 Securing data quality

14.1.1 Data output format

Under data output format you will find several options for testing the validity of the data. The default setting 'Use type of output column data' and provides no further adjustments. With this option, the definition in the database of the associated database column determines which data type is used.

Widget settings of database column **PRODUCT_LINE_ID**

Widget type	Mapping & Other	Features	Visual	Help texts	Data output format
<div> <div>Output type</div> <div> Number Currency Percentage Date / Time Text Use type of output column </div> </div> <div> <div>OK</div> <div>CANCEL</div> </div>					

Output types:

- **Number** - Requires a number
- **Currency** - Shows number values with currency symbol
- **Percentage** - Percentages, e.g. 12,34%
- **Date / Time** - Requires a date / time
- **Text** - To enter text, as a special validation option, there are regular expressions

Widget settings of database column PRODUCT_LINE_ID

Widget type	Mapping & Other	Features	Visual	Help texts	Data output format
Output type	<div> <div>Number</div> <div>Currency</div> <div>Percentage</div> <div>Date / Time</div> <div>Text</div> <div>Use type of output column</div> </div>				
Decimal places	<input type="text" value="0"/>				
Show separate groups	<input type="checkbox"/>				
How to show negative number	<div> <div>with minus sign</div> <div>with minus sign and in red colour</div> </div>				
Data quality check					
Custom validator Java 8 class	<input type="text" value=""/>				
Interval of old value (%)	<div> <div>Minimum allowed:</div> <input type="text"/> <div>V</div> </div>				
	<div> <div>Maximum allowed:</div> <input type="text"/> <div>V</div> </div>				
Interval	<div> <div>Minimum allowed:</div> <input type="text" value="0"/> <div>V</div> </div>				
	<div> <div>Maximum allowed:</div> <input type="text" value="<%maxProductlineID%>"/> <div>V</div> </div>				
<div> <div>OK</div> <div>CANCEL</div> </div>					

Decimal places

Show separate groups

How to show negative number

- You can set the number of decimal places displayed
- Serves for better readability of large numbers e.g. 1,000,000,000
- Negative numbers can only be viewed by a minus or colored red

Output type ,Currency'

Is identical to the output type ,number', but contains as a further option the setting for a currency symbol

Currency symbol	€
-----------------	---

Output type ,Date and Time'

Widget settings of database column PRODUCT_LINE_ID

Widget type	Mapping & Other	Features	Visual	Help texts	Data output format
Output type					<div> Number Currency Percentage Date / Time Text Use type of output column </div>
Show date picker					<input checked="" type="checkbox"/>
Date and time part					<div> Date Time Date with time </div>
Format					<div> Short Medium Long Full Custom </div>

With ,Show date picker' (default) users can easily pick a date

April 2, 2020 9:00

The date pickers prevents entry mistakes

14.1.2 Data Quality check

Custom validator Java 8 class

Data quality check

Custom validator Java 8 class

Optional. A Java 8 class that is testing the input value. The file directory of this file is defined in the Apparo Configuration Manager. This class is called automatically before Apparo Fast Edit is updating or inserting this row.

Interval of old value (%)

Interval of old value (%)	Minimum allowed:	<input type="text" value="50"/>
	Maximum allowed:	<input type="text" value="100"/>

Hereby you limit the validity of the entered values based on the existing values.

Example: In the widget, the value is 100. In this case, users may only enter values between 50% and 100% of the original value, so values between 50 and 100. Otherwise, the user receives an error message.

Interval

Interval	Minimum allowed:	<input type="text" value="1000"/>
	Maximum allowed:	<input type="text" value="2000"/>

Limits the validity of entries based on an absolute interval. Permissible are values only from 1000 to 2000. Interval limits can be set dynamically with variables.

Regular Expression (Only for type ,Text')

Regular expression for data quality

Using a regular expression is helpful to define more complex input rules. For example you can define that the first character must be an 'A' and then just numbers are allowed. Click the '?' icon to see the detail instructions.

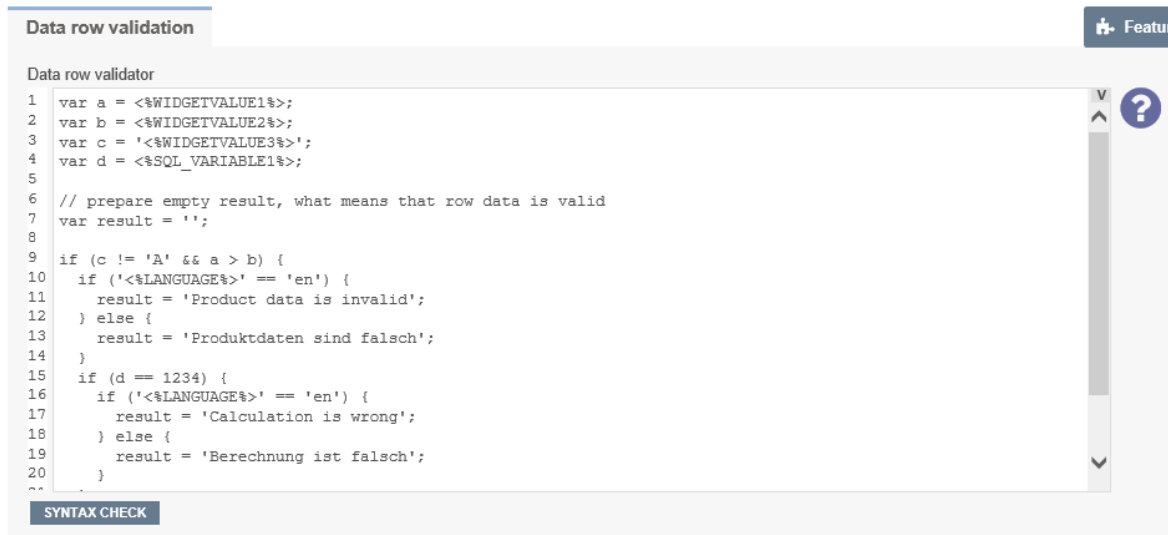
Characters		
Character	Description	Example
Any character except <code>[\\^\$. ?*+()]</code>	All characters except the listed special characters match a single instance of themselves.	<code>a</code> matches <code>a</code>
<code>\</code> (backslash) followed by any of <code>[\\^\$. ?*+()]</code>	A backslash escapes special characters to suppress their special meaning.	<code>\+</code> matches <code>+</code>
<code>\xFF</code> where FF are 2 hexadecimal digits	Matches the character with the specified ASCII/ANSI value, which depends on the code page used. Can be used in character classes.	<code>\xA9</code> matches © when using the Latin-1 code page.
<code>\n</code> , <code>\r</code> , and <code>\t</code>	Match an LF character, CR character and a tab character respectively. Can be	<code>\n\n</code> matches a

14.1.3 Data row validation

Enables validation of input when inserting or updating data

You can:

- Access all widget content via widget reference variables
- Use SQL variables
- Define own error texts, which are output automatically



Technical:

You define a JavaScript routine that can access widget reference variables or SQL variables.

An example can be obtained by clicking the question mark icon.

If the data row contains an error, an appropriate error message can be defined and the user is seeing exactly this error message.

14.2 Tracking of data changes

14.2.1 Auditing of data changes

The audit function you can use to document all data changes.

There are 2 different types of audit:

14.2.1.1 Simple Auditing

To save the audit information into the target table.

Auditing of data changes
Features

Simple auditing settings		
	Inserting a new row case	Updating or deleting row case
User name column	<input type="text"/>	<input type="text"/>
Date column	<input type="text"/>	<input type="text"/>
State (U,I,D) column	<input type="text"/>	
Row edit type column	<input type="text"/>	
In delete case delete data row physically	<input type="checkbox"/>	

It is possible to save the user name, date and time and the type of change for each row in the target table.

There are 2 different types of changes possible:

- The user adds a new row
- The user deletes or modifies a row.

The following states are possible: U = Update, I = Insert (paste), D = Delete (Delete).

Options

User name column

Stores the name of the user

Date column

Date column for storing update or insert date and time

State (U, I, D) column

The database column in that the state(U=Update,I=Insert,D=Delete) will be stored.

Row edit type column

In this auditing column the row edit type can be stored. The row edit type (type of string) is describing the way of editing.

In delete case delete data row physically

Physically delete row(s) with 'D' flag from table. If disabled then all deleted rows get the state 'D' and are not physically deleted.

14.2.1.2 Detailed Auditing

Storing detailed audit information in a separate audit database table helpful if every small change (eg a column) with name, timestamp, etc. should be documented.

Detailed auditing settings		
Auditing database table	SAMPLES_ADV_AUDITING	
Auditing column for user name	USERNAME	
Date column	CHANGE_DATE	
State (U,I,D) column	STATE_TYPE	
Row edit type column	ROW_EDIT_TYPE	
Custom value column		
Target table name column		
Business Case identifier column		
SQL statement column		
Summary change column		

Widget mapping		
Editing widget	Auditing column for the old value	Auditing column for the new value
PRODUCT_LINE_ID (NUMBER)	OLD_NUM_1	NEW_NUM_1
PRODUCT_NAME_EN (TEXT)		
PRODUCT_ID (NUMBER)		
PRODUCT_COLOR		

Options

Database schema

The database schema in that the auditing table is already stored.

Auditing database table

The database table for the auditing data.

Auditing column for user name

The database column of the auditing table in that the user name who has changed data will be stored.

Date column

The auditing column for storing the date/time of the data change.

State (U, I, D) column

The auditing table column in that the state must be stored (U=update, I=insert, D=delete).

Row edit type column

In this auditing column the row edit type can be stored.

Custom value column

In this auditing column a custom value with variables can be stored that is stored in the auditing table only.

Target table name column

In this auditing column the name of the target table of this Business Case can be stored.

Business Case identifier column

In this auditing column the Business Case ID (short name) can be stored.

SQL statement column

In this auditing column the SQL statement can be stored. Be sure that this column can store a long text.

Summary change column

This text contains all data changes in one string like oldValue=1, newValue=2,..The column names are defined in the widget list beneath.

14.2.2 Data History

Apparo Fast Edit can historicize a record (slowly changing dimension type 1 and 2). Information about "**Slowly changing dimension**", see:

http://en.wikipedia.org/wiki/Slowly_Changing_Dimensions

Note: For a historicizing the database must be able to perform "save points".

Since the Sybase / Informix / Teradata JDBC driver does not support this feature, the historicization of records within a Sybase or Informix or Teradata database is not possible.

This function automatically copies data rows when they are modified. It automatically manages the current record and makes it possible to either overwrite or historicize records within time frame definitions.

The user usually sees only the current line and not the data changes or deletions (if only virtually), the new rows are simply copies of the original lines.

14.2.2.1 Tab Main

Main
Advanced
Features

Using history functionality means that Apparo is automatically copying a data row if the user is changing a value into it and it is managing the 'date from'/'date to' columns automatically. Depending on the decision strategy (time resolution) it is updating the row or copying including time frame update. The row group normally contains all used primary key widgets. It must not contain date_from & date_to columns. For auto-handling the date from widget, it requires e.g. the <%CURRENT_DATE%> variable set as constant value in insert case or as default value. Please set the output type of this widget to at least date with time (medium). For milliseconds use the custom output type e.g. 'dd.MM.yyyy:hh:mm:ss.SSS'. Please read this [document](#) for more details.

Use columns as row group

VALID_FROM
VALID_TO
PLAN_YEAR
PLAN_MONTH
OFFICE_ID
PRODUCT_LINE_ID
PRODUCT_ID
SALES
STATUS_ID
STATE_REVISION_ID
FORECAST
FORECAST2
CONTACTS

The history functionality is helpful if the user is updating data values or creating new data rows and the Business Case must create automatically a copy of this complete data row to be able to see the history of the data changes.

The Business Case is managing automatically 'date from', 'date to' and 'current' columns of the target table. With these date columns it is possible to see the time dependencies of the changes. Apparo is combining data rows together to a 'row group'. A 'row group' are data rows that are storing detail information about an entity, for example a 'product entity' has many different prices over the time.

Depending on the decision strategy (time resolution) it is definable when the Business Case is updating the current row or making a copy of this data row. If for example the decision strategy is 'same hour' and there are 2 changes in the same hour then the second change is just updating the data row. If the second change is for example 5 hours later then the Business Case is creating a new data row including update of 'date from', 'date to' and 'current' columns. Please don't use widgets(columns) for the history feature that contain read/write expressions.

Example:
The Business Case is managing the price of a product. The 'row group' is the column 'product_id'.
The primary key is 'product_id' and 'date_from'.
'product_id' 'date_from' 'date_to' 'price'
5 20.12.2010 31.12.2999 100

Now the user is changing the price to '105', editing date: 05.01.2011'
5 20.12.2010 04.01.2010 100
5 05.01.2010 31.12.2999 105

If there are 2 or more data changes into a row in the same time frame then Apparo will update the row only. If the next change is outside of the time frame then Apparo is copying automatically this row and changing the 'date from' and 'date to' columns automatically too.
Special case: if simple auditing is used with specified 'state (U,LD)' column then deleting a data row will insert a new data row with flag 'D' instead of setting the flag to the same data row. (= 'Slowly changing dimension' strategy)

'Valid from' date column

VALID_FROM
☒ Validate that the new 'valid from' value precede the latest existing 'valid from' value of this row group.

The 'valid from' database column of the target table is used for storing the begin of the time frame for a row.
HINT: this column must be a part of the primary key

'Valid to' date column

VALID_TO

The 'last active' database column of the target table is used to mark the last record of a row group. In most cases, this is the current entry, but it can also be an entry that lies in the future. This setting is optional. It is automatically managed by the Business Case.

'Last active' flag column

CURRENT_FLAG

The 'last active' database column of the target table is used for marking the last active row of a group. This setting is optional. It is automatically managed by the Business Case.

The Business Case is managing automatically 'date from', 'date to' and "current" columns of the target table.

With these date columns it is possible to see the time dependencies of the changes.

Background

Apparo Fast Edit is combining data rows together to a 'row group'. A 'row group' are data rows that are storing detail information about an entity, for example a "product entity" has many different prices over the time.

Please don't use widgets (columns) for the history feature that contain read/write expressions.

Time resolution

If there are 2 or more data changes into a row in the same time frame then Apparo Fast Edit will update the row only. If the next change is outside of the time frame then Apparo Fast Edit is copying automatically this row and changing the 'date from' and 'date to' columns automatically too.

'Valid from' date column

The 'valid from' database column of the target table is used for storing the begin of the time frame for a row.

Validate that the new 'valid from' value precede the latest existing 'valid from' value of this row group.

Useful for manually entered valid from values

HINT: this column must be a part of the primary key

'Valid to' date column

The 'valid to' database column of the target table is used for storing the end of the time frame for a row. This setting is optional. It is automatically managed by the Business Case.

Use for current flag

The 'current' database column of the target table is used for marking the current row of a group. This setting is optional. It is automatically managed by the Business Case.

Advanced settings for the history feature:

Date for infinity

The infinity date is used in the 'date to' column for the current data row. The current row is usually valid to this date.

Value for ,is current' lines

Value for current line (for example, 1)

Value of non-current line (for example, 0)

14.2.2.2 Tab Advanced

It contains the settings for the infinity date, the current flag and other settings.

Main	Advanced
Date for infinity	<input type="text" value="12.31.2999 0:0:0"/> The infinity date is used in the 'date to' column for the current data row. The date format is MM.dd.yyyy H:m:s where <ul style="list-style-type: none"> d - day of month (1-31) M - month of year (1-12) yyyy - year H - hour m - minute s - second e.g. 1.1.2999 1:45:45 or 12.31.2999 11:11:11
History flag for current row	<input type="text" value="1"/> * Here you can define the value for the current column for the data row with the current values Specify number in format "###.###.###" or for text you can use any character
History flag for not current row	<input type="text" value="0"/> * Here you can define the value for the current column for all non-current data rows Specify number in format "###.###.###" or for text you can use any character
Set historical entries to read-only	<input type="checkbox"/> If enabled then data records with a valid-to date that lie in the past cannot be deleted or changed.
Don't delete entries physically	<input type="checkbox"/> If enabled and the user deletes a record with valid-to date in the future, the entry is not physically deleted, but the valid-to date is set to the current date and will be historicized.
Users can enter custom valid-to dates	<input type="checkbox"/> If enabled, the user can set custom valid-to dates for the current row. By default, the valid-to date is taken from the feature settings (infinity date) and used for the current row.
Use advanced constant value settings for 'valid-from' widget	<input checked="" type="checkbox"/> If enabled, the use of constant values of the widget 'valid-from' can be defined in a more detailed way.

Advanced constant value settings for 'valid-from' widget

The constant value for inserting a new history group entry	<input type="text" value="<%TIMESTAMP%>"/> V Use format 'yyyy-MM-dd HH:mm:ss.SSS' e.g. 2019-09-25 14:59:59.000
The constant value for editing an existing group entry	<input type="text" value="<%TIMESTAMP%>"/> V Use format 'yyyy-MM-dd HH:mm:ss.SSS' e.g. 2019-09-25 14:59:59.000
If activated, the user can manually overwrite the set constant values	<input type="checkbox"/>

If left empty, no constant values are used.
Set constant values are valid for all security groups if using security group-dependent constant values.

14.2.2.2.1 Date for infinity

The infinity date is used in the 'date to' column for the current data row.

The date format is: MM.dd.yyyy H:m:s

Where

- d - day of month (1-31)
- M - month of year (1-12)
- yyyy - year
- H - hour
- m - minute
- s - second

e.g. 1.1.2999 1:45:45 or 12.31.2999 11:11:11

14.2.2.2.2 History flag for current row

Here you can define the value for the current column for the data row with the current values
Specify number in format "###,###.###" or for text you can use any character.

In most cases the value '1' is used here.

14.2.2.2.3 History flag for not current row

Here you can define the value for the current column for all non-current data rows
Specify number in format "###,###.###" or for text you can use any character

In most cases the value '0' is used here.

14.2.2.2.4 Set historical entries to read-only

If enabled then data records with a valid-to date that lie in the past turns to read-only and cannot be deleted or changed.

14.2.2.2.5 Don't delete entries physically

If enabled and the user deletes a record with valid-to date in the future,
the entry is not physically deleted. In this case the valid-to date is set to the current date instead and will be historicized.

14.2.2.2.6 Users can enter custom valid-to dates

If enabled, the user can set manually custom valid-to dates for the current row.
By default is the valid-to date taken from the feature settings (infinity date) and used for the current row automatically.

14.2.2.2.7 Use advanced constant value settings for 'valid-from' widget

If enabled, the use of constant values of the widget 'valid-from' can be defined in a more detailed way. If this feature is activated, but the constant values for inserting and/or editing were left empty, then no constant values are used.

Set constant values are valid for all security groups if using security group-dependent constant values.

The advanced constant value settings for 'valid-from' widget offers three options:

1. *The constant value for inserting a new history group entry*

You can use custom or predefined variables here (e.g. <%TIMESTAMP%> or <%CURRENT_DATE%>) or a fix date value with the format 'yyyy-MM-dd HH:mm:ss.SSS'

e.g. 2019-09-25 14:59:59.000

2. *The constant value for editing an existing group entry*

You can use custom or predefined variables here (e.g. <%TIMESTAMP%> or <%CURRENT_DATE%>) or a fix date value with the format 'yyyy-MM-dd HH:mm:ss.SSS'

e.g. 2019-09-25 14:59:59.000

3. *If activated, the user can manually overwrite the set constant values*

Users can enter manually custom values instead of using the given constant values.

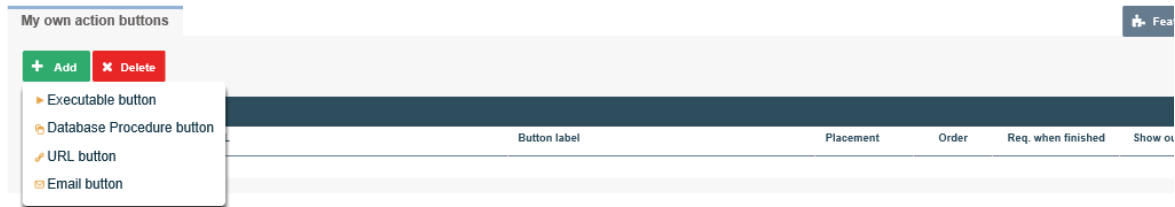
Hint: Further details can be taken from the History document (<https://doc.apparo.solutions>)

15 Calling external scripts / procedures / functions

15.1 Own action buttons

Action buttons can call executables files or scripts, database procedures, URLs and eMail Business Cases.

It is possible to specify different behavior patterns. For example, single call or a call for each selected row of data, etc.



15.2 Executable button

With Apparo Fast Edit you have several options for further processing of the data. With "Executable Button" you can add a button for processing (like .bat, .cmd, .sh, .sql). All files that are to be called up must be stored in the script file directory which was defined in the Apparo Configuration Manager. Using the Apparo Configuration Manager it is possible to change the used file directory.

15.2.1 General Settings

My own action button

General settings	Dialog visual	Information texts	Features
Database connection	SAMPLES		
DB procedure call expression	demo_change_status(<%city%>)		
Return value variable	<%RETURN_VALUE%>		
Placement	Right		
Gap (left)	4		
<div>OK</div> <div>CANCEL</div>			

Executable filename

Here you can select the batch processing file or sql file, which will be executed by this button.

Arguments

Optional you can use arguments (variables are allowed too) that will be delivered to the script or database procedure.

Return value variable

In this variable is the return value of the function/script stored.

Placement

Arrangement of buttons on the screen.

Gap

Space to the next button in pixel

15.2.2 Dialog visual

Here you will find settings for the layout and behavior of the message window.

My own action button

General settings	Dialog visual	Information texts	Features
"Please wait" font	Font face: Arial Size: 14 Style: Bold Align: Left Colour: #000000		
Output message font	Font face: Arial Size: 14 Style: Bold Align: Left Colour: #000000		
Finish message font	Font face: Arial Size: 14 Style: Bold Align: Left Colour: #000000		
Background colour	#FFFFFF		
Logo URL			
Dialog window size	Width: 300 Height: 150		
Automatically close dialog window	<input checked="" type="checkbox"/>		

OK CANCEL

15.2.3 Information texts

At this point you can define a label for your button and individual texts for waiting and finishing.

My own action button

General settings	Dialog visual	Information texts	Features
Language	Button label	"Please wait" message	Finish message
German	 Schalte alle Daten frei für Cor		
English	Approve all data		

OK CANCEL

15.2.4 Features

Here you can define the behavior of the button.

My own action button

General settings	Dialog visual	Information texts	Features
Refresh Business Case data after finish	<input checked="" type="checkbox"/>		
Show process output	<input type="checkbox"/>		
Hidden for	> Specify settings for security group		
Execution mode	Execute the script/procedure/email once		
Send eMail after execution	<input type="checkbox"/>		

OK CANCEL

Refresh Business Case data after finish

If enabled, the Business Case is reloading the database data again after execution of the script/procedure. This is helpful if your script/procedure is changing data that must be displayed in the Business Case too.

Show process output

If enabled, the user will see the script output in a small window.

Execution mode

Execute the script/procedure/email once

Here you can define the exact behavior of this button. Your script/procedure can be called for each row, selected row or just once.

Execute the script/procedure/email once

- Execute the script/procedure/email once
- Execute the script/procedure/email for all rows of the current page only
- Execute the script/procedure/email for each row of all pages
- Execute the script/procedure/email for each already selected row only

Send eMail after execution

After execution of a script or database procedure it is possible to send automatically an eMail. This eMail Business Case has access to all widget values of this Business Case. That means that the eMail body can contain values of this current Business Case.

15.3 Database procedure button

My own action button

Database connection

Here you can select the database connection on which the button will be proceeded.

DB procedure call expression

How to call a database function or procedure:

[Calling convention] procedure/function_name (argument1, argument2, ..., argumentN)

Please use same character cases for schema and procedure/function like defined in your database.

If the database connection of this procedure/function is same as the one for Business Case than the procedure/function is executed within the same database transaction.

The procedure must not commit or rollback the existing transaction, but is allowed to start its own inner (named) transaction (if supported by database) or use savepoint's.

For character or string argument use ' character to enclose argument.
Use at least one space between [Calling convention] and procedure name.

Your parameters may contain Apparo Fast Edit variables, for example: <%USER_NAME%>
Do not enclose Apparo Fast Edit variables with apostrophes or quotes.

Oracle or IBM DB2 database:

return - For calling a stored function that returns a value

MS SQL Server database:

Calling functions on SQL Server is not supported. It is possible to have a return value from procedure but [Calling convention] must be empty in this case.

Please use in your SQL Server procedure at the begin "SET NOCOUNT ON;" Then it is possible to use SQL commands in your procedure without having impact to the return value.

Sybase database:

select - For calling a stored function that returns a value

Teradata database:

return macro - For calling a Teradata macro that returns a value

macro - For calling Teradata macro that does not return a value

return - For calling a stored functions that returns a value

Return value variable

In this variable is the return of the function/script stored.

Placement

Here you can decide about the arrangement of buttons on the screen.

Gap

Space to the next button in pixel

15.4 URL buttons

With these buttons you can call any URL:

- Web Sites & Portals
- Reports & Dashboards
- Business Cases

My own action button

General settings	Dialog visual	Information texts	Features
URL	<input type="text" value="https://google.com"/>		
Placement	<input type="text" value="Left"/>		
Gap (right)	<input type="text" value="4"/>		
<input type="button" value="OK"/> <input type="button" value="CANCEL"/>			

15.5 E-mail Buttons

With these buttons you can send e-mails.

My own action button

General settings	Dialog visual	Information texts	Features
Email settings	<input type="text" value="SendMail (1)"/>		
Placement	<input type="text" value="Left"/>		
Gap (right)	<input type="text" value="4"/>		
<input type="button" value="OK"/> <input type="button" value="CANCEL"/>			

The settings for the e-mail you make in the selected e-mail business case.
All variables of the calling business case can be used.

15.6 Pre/Post execution

With Pre/Post-Execution it is possible to run automatically a script or a database procedure/function at certain moments.

It is possible to start a shell-script, database function/procedure or SQL-script before Business Case/server side file import starts, and/or after it is finished before forced Excel row import starts, and/or after it is finished after the user has inserted or updated data row

This behavior can be defined for all or for users that are members of a specified group. If the current user is member of a specified group then just the shell-script, database function/procedure, SQL-script of this group is executed only.

In all other cases the default script/function/procedure is called only. For now Apparo Fast Edit is supporting Oracle, Microsoft SQL Server, IBM DB/2, Sybase ASE/IQ (chained mode only) and Teradata databases.

A SQL-script is a text file with file name extension .sql that contains SQL-statements like INSERT, UPDATE, DELETE.

The commands are executed using the same database session like the Business Case and are separated by a semicolon.

Pre Business Case execution

(allows to run automatically a script or database function/procedure if the user is starting the Business Case)

Post Business Case execution in success case

(allows to run automatically a script or database function/procedure after the user has closed the Business Case with 'Ok' or 'Close' button)

Post Business Case execution in failure case

(allows to run automatically a script or database function/procedure after the user has closed the Business Case with 'CANCEL' or 'X' button)

Fast Edit is checking the browser state by default every 180 seconds, it may take up to 3 minutes after closing the BC with 'X' before the script/procedure is executed.

Post insert execution

(allows to run automatically a script or database function/procedure after a new row was inserted)

This insert can be done:

- From inserting area (Table Business Case)
- From insert mode (Single Business Case)
- From Excel file import
- From Excel row import using copy and paste
- From automatic server import
- From automatic import of email data-file attachments
- By copying row/s in the same window

The procedure or script will **NOT be executed after modifying a row in edit area.**

Post update execution

(allows to run automatically a script or database function/procedure after a row was updated)

This update can be done:

- From inserting area (Table Business Case)
- From insert mode (Single Business Case)
- From Excel file import
- From Excel row import using copy and paste
- From automatic server import
- From automatic import of email data-file attachments
- By copying row/s in the same window
- After modifying a row in edit area

Optionally, a query window to activate that appears when the user updates a row of data from the input area.

Post Excel import execution

(allows to run automatically a script or database function/procedure after any kind of Excel import has finished)

All Apparo Fast Edit variables can be used here, including:


- <%IMPORTED_ROWS%> count of imported rows
- <%INSERTED_ROWS%> count of inserted rows
- <%UPDATED_ROWS%> count of updated rows
- <%IMPORTED_FILE_NAME%> file name of the imported file (if applicable)
- <%EXCEL_IMPORT_ID%> An unique ID of type string

16 Action Business Cases (ABC)







Action BC is helpful for adding database action like changing data, calling scripts server side without user interactivity. It is possible to call an Action BC using AJAX.

An Action BC can contain own web output too, even buttons like yes/no are possible.

An Action BC can call database procedures or scripts automatically.


Demonstration Apparo Fast Edit

Please select type of Business Case you want to create now


	Table	A table Business Case is showing many data rows on the same page. The user can filter the data, edit, import from Excel, export to Excel and so on.
	Single	A single Business Case is showing just one data row only.
	Set	A grouping of multiple Business Cases (table/single) for more comfortable usage. You can define global filters that are filtering all Business Cases automatically too.
	Email import	Importing Excel data directly by email - send Excel sheets using email attachments and Apparo will import the Excel data directly into the database including file uploads. No web browser is necessary, just an email.
	Email	An eMail Business Case is a definition of an email text including usage behavior and can be used in another Business Cases of type 'table' or 'single' only. In these Business Cases it is possible to define buttons that can use this eMail Business Case.
	Action	Purpose of Action Business Case is to execute scripts or database procedures that can be called from a report/HTML page. Usage of AJAX and Javascript for automatically executing in the background is possible too.

CANCEL







You can find an execution example for AJAX and HTML in tab “Calling Business Case using http/Ajax” in the detail description. This example is using jquery as framework on the client side.

16.1 Example for calling a database function using AJAX

16.1.1 Create a new Action Business Case

 Demonstration Apparo Fast Edit

Please select type of Business Case you want to create now

	Table	A table Business Case is showing many data rows on the same page. The user can filter the data, edit, import from Excel, export to Excel and so on.
	Single	A single Business Case is showing just one data row only.
	Set	A grouping of multiple Business Cases (table/single) for more comfortable usage. You can define global filters that are filtering all Business Cases automatically too.
	Email import	Importing Excel data directly by email - send Excel sheets using email attachments and Apparo will import the Excel data directly into the database including file uploads. No web browser is necessary, just an email.
	Email	An eMail Business Case is a definition of an email text including usage behavior and can be used in another Business Cases of type 'table' or 'single' only. In these Business Cases it is possible to define buttons that can use this eMail Business Case.
	Action	Purpose of Action Business Case is to execute scripts or database procedures that can be called from a report/HTML page. Usage of AJAX and Javascript for automatically executing in the background is possible too.

CANCEL

Select "Action" and define the main settings:

Please use a database connection too because later it is necessary to define a database procedure that must be called automatically.

Main Action Business Case settings

Identifier / Short name	ID *
Business Case name	NAME *
Database connection	<div> <div></div> <div> Demo_Connection_1 dt7dwhf_ap696 Procurement SAMPLES </div> </div>
Show output	
Business Case security group	
Notes	

NEXT **CANCEL**

Demonstration Apparo Fast Edit Business Case Designer - ID / NAME

Run Close Save Cancel

Main settings

Actions

Variables

Link into Portal

Main settings

Identifier / Short name

Business Case name

Database connection

Show output

Business Case security group

Notes

16.1.2 Define a report variable

Please click to tab “Variables” and add a new variable.

Demonstration Apparo Fast Edit Business Case Designer - ID / NAME Anonymous Demonstration

Run Close Save Cancel

Main settings

Actions

Variables

Link into Portal


User defined variables Internal variables


+ Add X Delete


User defined variables

Variable name	Variable type

Select the type of new variable

 Script variable

 **Report variable**

 SQL variable (for all tables)

You can use JavaScript to compute advanced calculations and the result can be used in Apparo as any other variable. The execution is done server side only.

Report variables: They are used to deliver content to a Business Case using the URL e.g. from a report or to deliver the content of a widget from one Business Case to another one.

SQL variable to execute commands on all tables. Every time the SQL variable is used then the defined SQL is executed. The variable content is the first column of the first row of the executed SQL. You can use the variable (e.g. <%current_year%> in many input fields of the Business Case definitions, e.g. in header text, default value, constant value and so on.

CANCEL

Select “Report variable”.

The report variable is helpful for delivering a value from a HTML page to this Action BC. It will be used later as a parameter in the call of the database function.

Now the name and output settings:

Variable for Business Case

Variable name *

Variable description

Variable value	Data output format
Default value	<input type="text" value="test"/>

OK CANCEL

Change the output format to numeric:

Variable for Business Case

Variable name *

Variable description

Variable value	Data output format
<p>Defines the type of the variable, the data output format and the expected data input format.</p> <p>If you want to define the value of this variable at Business Case start time then you must care about the expected data input format. Expected format for date is MM.DD.YYYY (for example: 12.31.2018) or YYYY-MM-DD (for example: 2018-12-31), for time 'HH:mm:ss' (for example: 23:59:59) and for date with time 'MM.DD.YYYY HH:mm:ss' (for example: 2018-12-31 23:59:59). You can also use ISO 8601 date format (for example: 2018-12-31T23:59:59.999-0700).</p> <p>If you want to define own date/time format just for this variable use additional parameter FE_name_FORMAT, where name is name of this variable. Example: &FE_name_FORMAT=dd.MM.yyyy HH:mm (or using different format).</p> <p>Or you can use additional parameter dateFormat to define your own date/time format. Parameter dateFormat is global so it is used for all date/time report variables and primary keys. Example: &dateFormat=dd.MM.yyyy HH:mm:ss (or using different format).</p> <p>Output type</p> <div> <div>Text</div> <div>Number</div> <div>Date / Time</div> </div>	

OK CANCEL

Now the report variable is defined:

Run

Close

Save

Cancel

User defined variables

Internal variables

+ Add

✖ Delete

User defined variables

Variable name	Variable type
<%My_Report_Variable%>	Report variable

16.1.3 Define the actions

In the „Actions“ tab you can define the business logic that must be called if the Action BC is running.

In this example it must call the Oracle function oracle_function with parameter <%My_Report_Variable%>.

Please be sure that you use right character case of the function/procedure name.

☒ Enable Pre Business Case action execution

Pre Business Case execution (allows to run automatically a script or database function/procedure if the user is starting the Business Case)

Automatic execution of

Database procedure

for

all users

Name

demo_proc(<%My_Report_Variable%>)

Return value variable

<%RETURN_VALUE%>

☐ Enable Post Business Case action execution in success case
☐ Enable Post Business Case action execution in failure case
☐ Enable Exit Business Case action execution

16.1.4 Calling the Action BC

There are several ways to call the Action BC.
The easiest way is simply by calling the URL.

In the standalone version you will find it under Advanced in Link into Portal

Link into Portal

- 09. Automatic main output
- 10. Small data driven apps
- 11. (Automatic) script & procedure calls
- 12. Creating DB tables and columns
- 13. Dynamic Forms
- 14. EIBC
- 15. Procurement demo
- 16. Smartphone

Main Advanced Tiles

Open in

☐ Same window
☐ Own small window
☒ New tab

Here you can select if the entry is opening in a new tab, the same window or in a custom sized new window.

Additional Parameters

Space for URL parameters: e.g. &insertMode=true or country_id=en
If you want to fill a report variable of type date then the expected format is english, e.g. 12.31.2018 or 123.45

Security groups

Comma separated list of required security groups to see this portal entry. Attention: It is possible to run a Business Case using an URL. Therefore it is necessary to secure the Business Case too using security setting in the Business Case itself.

URL

<http://afe32winstandalone.sad.com/apparo/pages/businessCases/userInterface/businessCase.xhtml?bc=ID&clientId=Demo&backLink=http%3A%2F%2Fafe32winstandalone.sad.com%2Fapparo%2Fportal%2Fportal.xhtml%3Ffaces-redirect%3Dtrue>

You can use this URL for running this Business Case directly without using the portal first. You can change the backLink parameter to define another website that must be called after this Business Case is closed pressing close/cancel button.

Other ways, like calling via Ajax can be found in the user guide.

17 Filter data output

The function filter data output represents the global filter of the business case. Additional filters can be added through filter widgets.

Filter data output

The SQL filter conditions are filtering data rows of the target table for the output.

CURRENT_FLAG = 1

V

?

SYNTAX CHECK

+

-

*

/

&

|

^

||

=

>

<

>=

<=

(

)

'

>

Security group based filtering

You can create different filters for different security groups. If a user is a member of the security group, only the security groups based filter is used instead of the global BC filter.

Variables are allowed.

Syntax

In the filter, you can use native SQL. It represents the Where clause of the SQL query and filters the output of the target table.

Example

SELECT * FROM target table WHERE [=data output filter]

17.1 Filter widgets

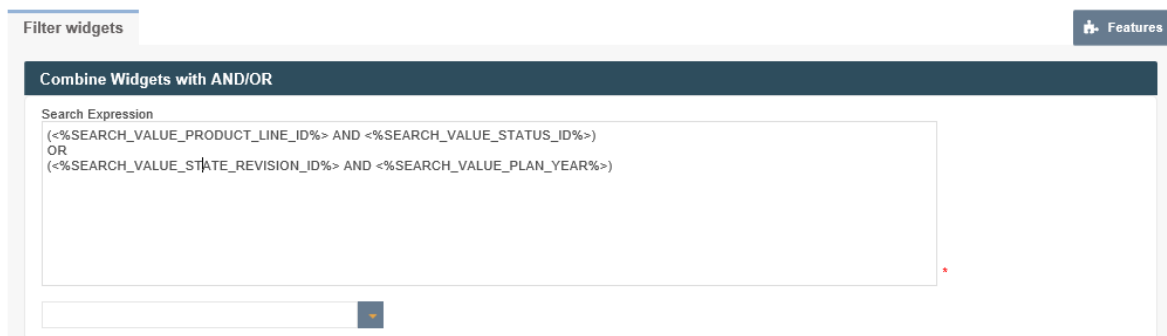
Contains the optical settings for the additional filter page and the settings for combining filter widgets

17.2 Filtering prompt page settings

You can add a filter page that is displayed before the user can see the data page for editing, inserting etc. The filter page will be used automatically if you add a filtering widget that is placed in the filter page. On this page you can define the title, description and other optical definitions of this filter page.

17.3 Combine Widgets with AND/OR

The function extends the filtering possibilities with filter widgets.



Standard type of searching is combining all used searching widgets with logical 'AND' operator. If you want to combine them differently then you must use 'Advanced Type Of Searching'. In 'Search Expression' you can define your own combination of searching widgets. You can combine them with operators 'AND' and 'OR' and you can also use brackets '(' and ')'

Each searching widget must be used exactly once in the search expression.

The following examples contain combinations of these four filters widgets

Examples

One of the set criteria is met:

<%SEARCH_VALUE_PRODUCT_ID%> or <%SEARCH_VALUE_PRODUCT_COLOUR%> or
<%SEARCH_VALUE_PRODUCT_SIZE%> or <%SEARCH_VALUE_PRODUCT_LINE_ID%>

The product line and ONE of the other filter criteria is met:

<%SEARCH_VALUE_PRODUCT_LINE_ID%> and (<%SEARCH_VALUE_PRODUCT_ID%> or
<%SEARCH_VALUE_PRODUCT_COLOUR%> or <%SEARCH_VALUE_PRODUCT_SIZE%>)

The product line or the combination of ALL other filter criteria are met:

<%SEARCH_VALUE_PRODUCT_LINE_ID%> or (<%SEARCH_VALUE_PRODUCT_ID%> and
<%SEARCH_VALUE_PRODUCT_COLOUR%> and <%SEARCH_VALUE_PRODUCT_SIZE%>)

18 Variables

18.1 Definition

Variables are placeholders, they return one or more values. They can contain fixed values or perform calculations and queries dynamically.

At run time, i.e. when running the Business Case, the value of the variable is calculated and returned to the variables location.

Syntax: <%Variable_name%>

Variable names are case-sensitive.

User defined variables

Internal variables

Features

+ Add

✖ Delete

User defined variables

Variable name	Variable type
<%bulk_text%>	Script variable
<%city%>	Script variable
<%NextID%>	SQL variable (target table)

Variables for used filter widgets

Variable name	Variable description
<%SEARCH_KEY_PRODUCT_LINE_ID%>	Key value for filter lookup widget mapped to PRODUCT_LINE_ID column name
<%SEARCH_KEY_STATE_REVISION_ID%>	Key value for filter lookup widget mapped to STATE_REVISION_ID column name
<%SEARCH_KEY_STATUS_ID%>	Key value for filter lookup widget mapped to STATUS_ID column name
<%SEARCH_VALUE_PLAN_YEAR%>	Value of filter widget mapped to PLAN_YEAR column name
<%SEARCH_VALUE_PRODUCT_LINE_ID%>	Value from filter lookup widget mapped to PRODUCT_LINE_ID column name
<%SEARCH_VALUE_STATE_REVISION_ID%>	Value from filter lookup widget mapped to STATE_REVISION_ID column name
<%SEARCH_VALUE_STATUS_ID%>	Value from filter lookup widget mapped to STATUS_ID column name

Basically, there are user-defined variables and internal variables.

Apparo Fast Edit supports 6 different types of variables:

- Internal pre-defined variables
- Operating system environment variables
- Script variables
- SQL variables
- Report variables
- Widget reference variables

Variables can be used in practically all settings and other variables

In Business Cases, you can create these types of variables:

Select the type of new variable

- Script variable**: You can use JavaScript to compute advanced calculations and the result can be used in Apparo as any other variable. The execution is done server side only.
- Report variable**: Report variables: They are used to deliver content to a Business Case using the URL e.g. from a report or to deliver the content of a widget from one Business Case to another one.
- SQL variable (for all tables)**: SQL variable to execute commands on all tables. Every time the SQL variable is used then the defined SQL is executed. The variable content is the first column of the first row of the executed SQL. You can use the variable (e.g. <%current_year%> in many input fields of the Business Case definitions, e.g. in header text, default value, constant value and so on.
- SQL variable (for target table only)**: SQL variable for Business Case target table only.

CANCEL

18.2 Use of variables in the Designer

Many widget settings can be made dynamic with variables.

Examples:

18.2.1 Variables in lookup definitions

Lookup table key column for comparing: ☐ Reading expression

Lookup table value column for output: ☐ Reading expression

The associated database column is composed of, Name_ 'and the return value of the language used. German users are assigned to the column NAME_DE and English users to the NAME_EN column

18.2.2 Variables in labels, hint texts, the header and footer

Column label

Language	Column label
German	<%Product_line%> x v
English	<%Product_line%> v

In this example, the heading of the column is output by variables

18.2.3 Variables in filter definitions:

SQL where condition:

Dynamic SQL filter

18.2.4 Variables used in data validations

Example for the use of dynamic variables as interval:

In a widget of type "input field", the permissible range of values is restricted:

Interval	Minimum allowed:	<%MIN_INTERVAL%>
	Maximum allowed:	<%MAX_INTERVAL%>

Example of dynamic intervals that restrict the values input by calculations.

Dynamic values are realized via variable:

Our SQL variable is of type SQL variable (target table only). This has the advantage that automatically all user-group-dependent filters are used.

The current line is identified by the value in the widget PRODUCT_ID. That PRODUCT_ID is a primary key.

The following sample SQL for SQL variable would be possible:

SELECT min_value FROM target_table WHERE product_id = <%PRODUCT_ID%>

In this case, <%PRODUCT_ID%> refers to the widget PRODUCT_ID in the Business Case and returns the current value.

The SELECT returns the value min_value of the current line and stores it in the new SQL variable "VAR_MIN_CALC".

The SQL is executed every time when accessing the variable "VAR_MIN_CALC".

Example for the use of variables in the data row validation:

Data row validation

Data row validator

```

1  var a = <%WIDGETVALUE1%>;
2  var b = <%WIDGETVALUE2%>;
3  var c = '<%WIDGETVALUE3%>';
4  var d = <%SQL_VARIABLE1%>;
5
6  // prepare empty result, what means that row data is valid
7  var result = '';
8
9  if (c != 'A' && a > b) {
10   if ('<%LANGUAGE%>' == 'en') {
11     result = 'Product data is invalid';
12   } else {
13     result = 'Produktdaten sind falsch';
14   }
15   if (d == 1234) {
16     if ('<%LANGUAGE%>' == 'en') {
17       result = 'Calculation is wrong';
18     } else {
19       result = 'Berechnung ist falsch';
20     }
21   }
22 }

```

SYNTAX CHECK

In this example widget reference variables, SQL variables and internal variables have been used

18.2.5 Variables in variables

Examples for the use in variables

Script variable:

Script body

Script language : javascript

You can see a detailed JavaScript language description including examples by clicking the question mark icon placed next to the editor.

Attention: If you want to use a Apparo variable in Javascript that contains text then you must use it in quotes, for example: `string.replace('<@TEXT1@>','<@TEXT2@>','text')` If a script variable must return value true or false then it must be a string like 'true'.

```

1  var result = <%SQL_COUNT_VAR%> / 100;
2  result;
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

```

SYNTAX CHECK

☒ Calculate the variable value before each usage again

In this example, an internal variable is used within a JavaScript variable

SQL variable:

SQL expression

select PROD_NAME from SAMPLE_PRODUCTS where PORDUCT_ID=<%PRODUCT_ID%>

SYNTAX CHECK

+

-

*

/

&

|

^

||

=

>

<

>=

<=

(

)

'

☒ Calculate the variable value before each usage again

SQL variable: Widget reference variables are often used in SQL variables. JavaScript variables are also possible.

125

18.3 Internal Variables

The following variables are predefined and can be used immediately:

Variable name	Variable description
<%AFE_HOME_DIR%>	Folder on the server which contains AFE settings
<%AFE_BC_NAME%>	Name of currently opened Business Case
<%AFE_BC_ID%>	ID of the Business Case
<%AFE_CLIENT_ID%>	ID of the used client
<%SERVER_NAME%>	Name of server where Apparo Fast Edit is running
<%USER_NAME%>	Name of currently logged user
<%USER_LOGIN%>	Unique login name of currently logged user
<%LANGUAGE%>	Identifier of language in which user interface is displayed
<%CURRENT_DATE%>	Current date and time
<%DATE%>	Current date
<%TIMESTAMP%>	Current date and time
<%TIME_MS%>	The number of milliseconds since 1.1.1970 (UNIX timestamp)
<%PRIMARY_KEY%>	The primary key of current row
<%PRIMARY_KEYS%>	Comma delimited list of the used primary keys
<%ROW_EDIT_TYPE%>	Type of data modification. Output is of type string
<%SELECTED_ROWS_COUNT%>	This variable is helpful for output e.g. "Are you sure you want to delete X rows?"
<%ROWS%>	Count of current visible rows
<%BULK_UPDATED_ROWS%>	Count of all updated rows
<%INSERTED_ROWS%>	Count of all inserted rows during Excel import
<%UPDATED_ROWS%>	Count of all updated rows during Excel import
<%IMPORTED_ROWS%>	Count of all imported rows during Excel import
<%IMPORTED_FILE_NAME%>	Name of the currently imported Excel file
<%EXCEL_IMPORT_ID%>	Universally unique identifier (UUID) of type String of each Excel import
<%EXPECTED_COLUMNS%>	List of expected columns for Excel import
<%LINE%>	This variable is helpful for display error during import e.g. "Import error in line X:"
<%SAME_PK_ROWS%>	It is helpful for display error message like "There is already a row with the same primary key value(s). Counting <%SAME_PK_ROWS%>"
<%UPLOADED_FILE_NAME%>	Name of the uploaded file (file upload/download widget)
<%DELETED_FILE_NAME%>	Name of the deleted file (file upload/download widget)
<%RETURN_VALUE%>	In this variable the return code of the function/script is stored.

If the Business Case uses search fields, e.g. a filter lookup, then the matching variables are automatically defined for each search widget:

<%SEARCH_KEY_COLOR%>	Key-Value of the Lookup widget, mapped to column 'COLOR'
<%SEARCH_VALUE_COLOR%>	Value of the Lookup widgets, mapped to column 'COLOR'

18.4 Report Variables

They are used to deliver content to a Business Case using the URL e.g. from a report or to deliver the content of a widget from one Business Case to another one.

Using a hyperlink in the report, the value can be transported to the connected Business Case.

A report variable in reports has the syntax FE_name.

Here you can define the "name".

Variable for Business Case

Variable name	VAR1 *
Variable description	

Variable value	Data output format
Default value	test

OK
CANCEL

The default value is used only if the report does not provide a value for this variable.

Example of calling a Business Case from a Cognos report (URL):

```
/cognos8/cgi-bin/cognos.cgi?b_action=xts.run&m=portal/bridge.xts&c_env=/portal/env.xml&c_mode=post&c_cmd=/KFE/pages/userInterface.jsf?bc=BCNAME&FE_Var1=1234&backLink=%2Fcontent%2Ffolder%5B%40name%3D%27Apparo+Fast+Edit+Demonstration%27%5D
```

In the URL has the report variable Var1 the value 1234

In reality report variables are often used to transport e.g. the values of prompts to the Business Case.

18.5 SQL Variables

There are 2 different types of SQL variables:

- **SQL variable (for all tables)**

SQL variable for executing selects in all tables. Each time you use the variable the associated SQL is executed. This variable contains the content of the first row, first column (depending on the SQL command)

- **SQL variable (for target table only)**

SQL variable for the Business Case target table. All filters of the Business Case are considered.

Example:

Variable for Business Case

Variable name	<%NextID%>
Variable description	

Variable value	Data output format
<div>SQL expression</div> <div>select NVL(MAX(ID),0) + 1 from FESAMPLES.SAMPLE_FORECAST</div>	

The main difference is that a **SQL-variable (for target table only)** automatic uses:

- The filter of the Business Case
- All security-dependent filters
- All Widget dependent filters

Therefore, the SQL of the variable must also use the target table so that the filter will also find the same column names.

SQL variables (for target table only) are very useful for calculations that relate to the target table - e.g. sum of all sales, as all the used filters are considered automatically.

Since the output changes when using filter widgets, usually this dynamic filter restriction must also be considered.

In a **SQL variable (for target table only)** this is done automatically, in opposite to a SQL variable (for all tables).

An SQL variable is always executed when it is used.
As result, the first result value is used.

18.6 Script Variables

A script variable is a routine that returns a value. It is not connected to a database session.

Variable value	Data output format
<div> <div>Script body</div> <div> <p>Script language : javascript</p> <p>You can see a detailed JavaScript language description including examples by clicking the question mark icon placed next to the editor.</p> <p>Attention: If you want to use a Apparo variable in Javascript that contains text then you must use it in quotes, for example: <code>string.replace('<%TEXT1%>','<%TEXT2%>','text')</code> If a script variable must return value true or false then it must be a string like 'true'.</p> <pre> 1 var groups = afe.getGroupsByRegex('demo_office.*'); 2 var result = groups[0]; 3 if(result) { 4 result = result.substring(12); 5 result = result.toUpperCase(); 6 } 7 result = 'BERLIN'; //simplified for the demo only 8 result; 9 </pre> </div> </div>	

The calculated value is returned by ,result'

You can use in the JavaScript routine SQL variables, reference variables and internal variables too.

The Logic is defined by **JavaScript** and can be combined with SQL-Queries.

You can use scrip variables within database connection settings, but connection pooling will be disabled then.

18.7 Widget reference variables

They include the value of a widget in the corresponding row.

Due the row reference these variables can be used only where a row reference is applicable.

They cannot be used e.g.: in the calculation area or in the header and footer.

Syntax: <%COLUMN_NAME%>

Editing widgets								
<input type="checkbox"/>	Column	Column name	Widget type	Title	PK	RO	H	NN
<input type="checkbox"/>	1	> VALID_FROM	> Input field	> valid-from	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	2	> VALID_TO	> Input field	> valid-to	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	3	> PLAN_YEAR	> Input field	> Year	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	4	> PLAN_MONTH	> Input field	> Month	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	5		> Spacer & Title	>				<input type="checkbox"/>
<input type="checkbox"/>	6	> OFFICE_ID	> Lookup dropdown (for all tables)	> Office	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	7	> PRODUCT_LINE_ID	> Lookup dropdown (for all tables)	> Product line	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	8	> PRODUCT_ID	> Lookup dropdown (for all tables)	> Product	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	9	> SALES	> Input field	> Sales	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	10	> STATUS_ID	> Lookup dropdown (for all tables)	> My status	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	11	> STATE_REVISION_ID	> Lookup dropdown (for all tables)	> Revision status	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	12	> FORECAST	> Input field	> Plan data	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	13	> FORECAST2	> Input field	> Plan2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	14	> FORECAST3	> Input field	> Plan3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	15	> FORECAST4	> Input field	> Plan4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

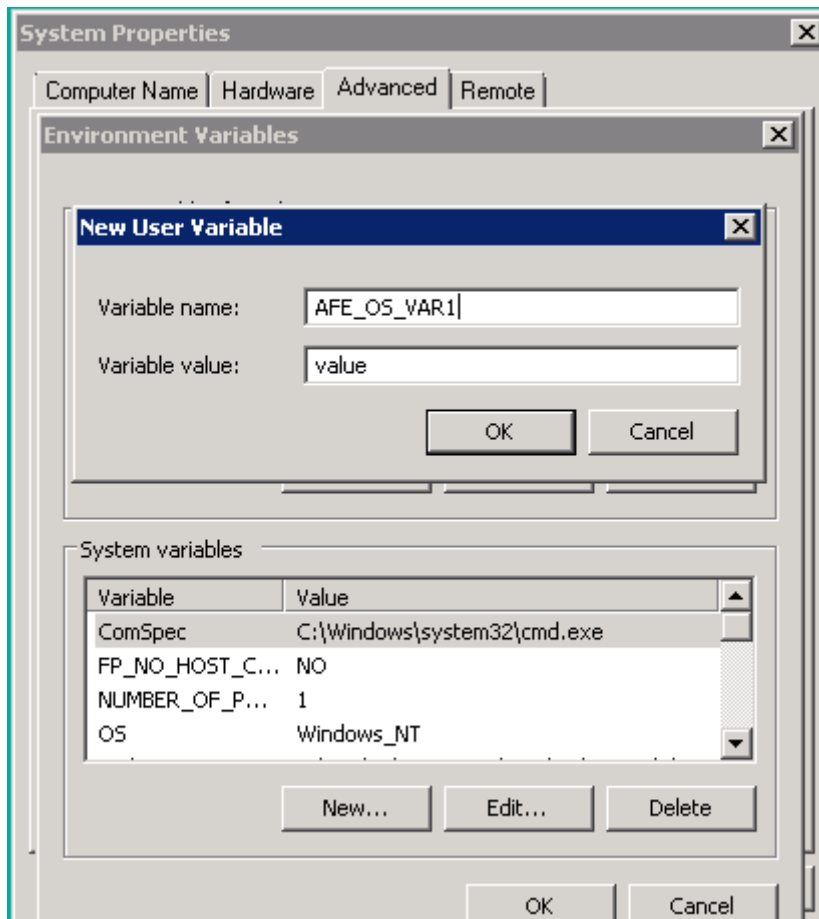
Example of a widget reference variables: <%OFFICE_ID%>

18.8 Environmental variables

These variables are defined in the operating system and can be used in all Business Cases.

Syntax: <%AFE_VARIABLE%>

Example of the definition in Windows:



To be recognized by Fast Edit the variable must start with 'AFE_'

18.9 Debugging variables

In the use of script and SQL variables with increasing complexity occur more frequently logical and/or syntactical errors. This chapter gives you an insight on how to recognize or find the errors.

18.9.1 Definition

With debugging the search and elimination of errors is called in computer science.

18.9.2 Variable output for debugging purposes

If variables are not shown on the Business Case, e.g. when they are used in other variables or used for passing parameters in script or DB procedure calls, it is often difficult to detect errors.

It is therefore advisable to output these variables directly in the Business Case, at least during the development phase.

Variable output in the head area

Variables without row reference can be output directly in the head area.

Target table	Header	Footer	Visual	Colours	Widgets	Row ordering
Title & Description	Language	Title				
	German	Produktliste				
	English	Product list Debug: <%avg_450%> <%NEXT_PRODUCT_ID%>				

Variable output in the edit area

Variables with row reference, that are generally widget reference variables or script- and SQL variables, containing widget reference variables, cannot be output in the header for debugging purposes because widget reference variables always contain the content of the widget, in the row they are used.

18.9.3 Debugging script variables

In addition to the substantive examination by outputting in the head area, there are other ways to check for errors:

Syntax check for script variables

JavaScript variables make it possible to check the included JavaScript commands for syntax errors. If errors occur, the corresponding line is highlighted and a description of the error is displayed below the script area.

In the following example, the semicolon at the end of the first line is missing:

Script body

Script language : javascript

You can see a detailed JavaScript language description including examples by clicking the question mark icon placed next to the editor.

Attention: If you want to use a Apparo variable in Javascript that contains text then you must use it in quotes, for example: `string.replace('<TEXT1>','<TEXT2>','text')` If a script variable must return value true or false then it must be a string like 'true'.

1

var result = 'Rows: '

2

result;

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

?

SYNTAX CHECK

Verification errors:

Line: 1 - Expected ';' and instead saw 'result'.

> /*global afe*/ var result = 'Rows: '

If the syntax is without errors, the error check displays an appropriate message:

SYNTAX CHECK

Verification OK

Error output in the log

Errors in JavaScript variables are always output in the AFE Log.

You can find the file "afe.log" in the folder [Apparo-HOME]/FastEdit/logs/

When using our example variable in the header area you would find the following error in the log:

2015-05-07 14:02:03,547 [ajp-apr-9800-exec-8] WARN ScriptVariableResolver - The exception is:
sun.org.mozilla.javascript.internal.EvaluatorException: syntax error (<Unknown source>#1) in <Unknown
source> at line number 1

2015-05-07 14:02:03,547 [ajp-apr-9800-exec-8] WARN ScriptVariableResolver - Error in script variable
named '100'

2015-05-07 14:02:03,547 [ajp-apr-9800-exec-8] WARN ScriptVariableResolver - The script body is:

var x = * 100;

//Rückgabegabe des berechneten Wertes

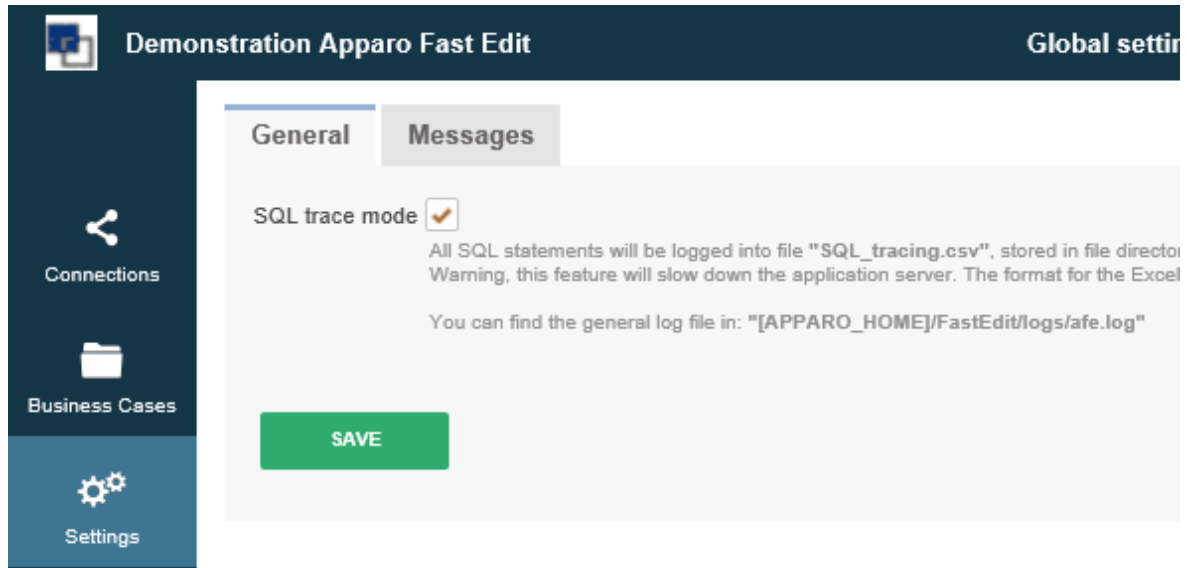
x;

*The widget reference variable here was not resolved because of the missing row reference and was
producing an erroneous formula.*

18.9.4 Debugging SQL variables

Error in SQL variables are displayed in the output in the header or edit area, and in the file afe.log.

A better way to identify problems with SQL variables provides the SQL Trace log, which must be activated in the designer:



You can find the file "afeSQL.csv" in the folder [Apparo-HOME] / FastEdit / logs /

In the trace log all SQL queries are stored in tabular form with the following information:

Timestamp, client, business case, user name, execution time, SQL command

07.05.2015 15:26	T3	sb_variable_sql_TBC1	de	0:00:00.000	Select count(SALES_ID) from TESTING.SAMPLE_SALES
---------------------	----	----------------------	----	-------------	---

Typical error outputs are:

Select count (SALES_ID) from TESTING.SAMPLE_SALES ORA-00904: "CUONT": invalid identifier

Or

Select SALES_DESCR from ".SAMPLE_SALES where SALES_ID = " ORA-00903: invalid table name

19 Data hierarchies

For the presentation of hierarchical data are lookup widgets ideal.

The easiest way is to explain this with an example.

For our example, we select geographical data. The top level are the continents, subordinated therein are countries and therein regions.

19.1 What do we need?

For the illustration of this hierarchy we need four database tables.

1. The target table, which stores the actual records
2. Three lookup tables containing the IDs of the data set, the real name and the ID of the parent element.

The assignment to the parents are done via the ID of the predecessor, for example, each country has additionally added the ID of the continent on which it is located.

To practice it, the scripts of each table:

Target table:

```
CREATE TABLE SAMPLE_LOOKUP
(
    "LOOKUP_ID" NUMBER,
    "LOOKUP_CONTINENT" NUMBER,
    "LOOKUP_COUNTRY" NUMBER,
    "LOOKUP_REGION" NUMBER )
```

Lookup table continent:

```
CREATE TABLE SAMPLE_LOOKUP_CONTINENT
(
    "CONTINENT_ID" NUMBER,
    "CONTINENT_NAME" VARCHAR2(100 BYTE) )
```

CONTINENT_ID	CONTINENT_NAME
1	AMERICA
2	AFRICA
3	AUSTRALIA
4	EUROPE
5	ASIA
6	ANTARCTICA

Lookup table country:

```
CREATE TABLE "TESTING"."SAMPLE_LOOKUP_COUNTRY"
(
    "COUNTRY_ID" NUMBER,
    "COUNTRY_NAME" VARCHAR2(100 BYTE),
    "CONTINENT_ID" NUMBER )
```

COUNTRY_ID	COUNTRY_NAME	CONTINENT_ID
3	BRAZIL	1
4	ZAMBIA	2
5	RSA	2
6	MOROCCO	2


Lookup table region:

```
CREATE TABLE "TESTING"."SAMPLE_LOOKUP_REGION"
(
    "REGION_ID" NUMBER,
    "REGION_NAME" VARCHAR2(100 BYTE),
    "COUNTRY_ID" NUMBER )
```

REGION_ID	REGION_NAME	COUNTRY_ID
1	California	1
2	Illinois	1
3	Alabama	1
4	Ontario	2
5	Quebec	2
6	Manitoba	2

19.2 Expected result

At the end we want a Business Case, in which the selection of a country is depending of the previously selected continent. The selectable countries shown are on the selected continent. If we choose the country, then the only selectable regions in the final lookup widget, are laying in the previously selected country. For the filter widgets used this should also apply.


Demonstration Apparo Fast Edit

BC name: Data hierarchy example

Current language is: en

Searching Continent	Searching country FOR	Searching REGION FOR	Searching CITY FOR	Searching STREET FOR
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="button" value="SEARCH"/> <input type="button" value="RESET FILTERS"/>				

<input type="checkbox"/>	CONTINENT	<input type="checkbox"/>	COUNTRY IN	<input type="checkbox"/>	REGION	<input type="checkbox"/>	CITY	<input type="checkbox"/>	STREET
<input type="checkbox"/>	EUROPE	<input type="checkbox"/>	SPAIN	<input type="checkbox"/>	Catalunia	<input type="checkbox"/>	Barcelona	<input type="checkbox"/>	Barcelona main street
<input type="checkbox"/>	EUROPE	<input type="checkbox"/>	SLOVAK REPUBLIC	<input type="checkbox"/>	High Tatras	<input type="checkbox"/>	Poprad	<input type="checkbox"/>	Plzenska
<input type="checkbox"/>	ASIA	<input type="checkbox"/>	JAPAN	<input type="checkbox"/>	Kyushu	<input type="checkbox"/>	Skoyo	<input type="checkbox"/>	Skoyo street 1
<input type="checkbox"/>	AMERICA	<input type="checkbox"/>	CANADA	<input type="checkbox"/>	Ontario	<input type="checkbox"/>	Hamilton	<input type="checkbox"/>	Hamil street
<input type="checkbox"/>	AFRICA	<input type="checkbox"/>	MOROCCO	<input type="checkbox"/>	Oriental	<input type="checkbox"/>	Oriental 1 city	<input type="checkbox"/>	oriental street 2
<input type="checkbox"/>	AFRICA	<input type="checkbox"/>	MOROCCO	<input type="checkbox"/>	Oriental	<input type="checkbox"/>	Oriental 2 city	<input type="checkbox"/>	oriental street 1
<input type="checkbox"/>	AMERICA	<input type="checkbox"/>	USA	<input type="checkbox"/>		<input type="checkbox"/>		<input type="checkbox"/>	

19.3 Implementation

19.3.1 Creation of the Business Cases

First, we need a Business Case, which is assigned to the target table.

Identifier / Short name	Data hierarchy example	*
Business Case name	Data hierarchy example	*
Database connection	SAMPLES	*
Target database table/view	SAMPLE_LOOKUP	*
Notes	showing a data hierarchy, using widget reference variables	

First, our Business Case looks like this:

Demonstration Apparo Fast Edit Business Case Designer - Data hierarchy example (1) / Data hierar... Anonymous Demonstration

Run Close Save Cancel Target table Header Footer Visual Colours Widgets Row ordering Link into Portal Features

+ Add X Delete

Editing widgets

Column	Column name	Widget type	Title	PK	RO	H	NN
<input type="checkbox"/>	1 > LOOKUP_ID	> Input field	> Lookup id	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	2 > LOOKUP_CONTINENT	> Input field	> Lookup continent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	3 > LOOKUP_COUNTRY	> Input field	> Lookup country	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	4 > LOOKUP_REGION	> Input field	> Lookup region	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	5 > LOOKUP_CITY	> Input field	> Lookup city	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	6 > LOOKUP_STREET	> Input field	> Lookup street	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Show description

19.3.2 Customizing of the Business Case

We set the widget ID as the primary key and hide the widget. We produce the consecutive numbering either by DB trigger or an SQL variable, which we set as a constant in the widget.

Widget settings of database column LOOKUP_ID

Widget type	Mapping & Other	Features	Visual	Help texts	Data output format
Column name	LOOKUP_ID <input type="button" value="v"/> * Enable expressions <input type="checkbox"/>				
Default value	<%NEW_ID%> <input type="button" value="v"/> for all users <input type="button" value="v"/>				
Constant value	Do not use constant value <input type="button" value="v"/>				
Variable for using content in detail BC	<input type="text"/>				
<input type="button" value="OK"/> <input type="button" value="CANCEL"/>					

In the next step we will modify the widget types of the remaining widgets to lookup widget (Lookup dropdown (for all tables))

For the child widgets, we add an additional filter in the setting, SQL where condition. Thus, the selection of widget is limited to all elements in the lookup table associated with the previously selected continent. The value for this comes from the widget continent previously selected, and will be handed over with the widget reference variable <% LOOKUP_CONTINENT%>.

Demonstration Apparo Fast Edit						Business Case Designer - Data hierarchy example (1) / Data hierar...		Anonymous		Demonstration	
Widget settings of database column LOOKUP_COUNTRY											
Widget type	Mapping & Other	Features	Lookup & Dropdown & Multiselect	Visual	Help texts	Data output format					
Database connection	SAMPLES <input type="button" value="v"/> *										
Lookup table	SAMPLE_LOOKUP_COUNTRY <input type="button" value="v"/> *										
Join lookup table key(s) to target table	COUNTRY_ID <input type="button" value="v"/> *			Reading expression <input type="checkbox"/>							
Lookup table value column for output	COUNTRY_NAME <input type="button" value="v"/> *			Reading expression <input type="checkbox"/>							
Multivalue 'Order By'	Ascending <input type="button" value="v"/>										
Lookup table sort column	Use same column like for displaying the value <input type="button" value="v"/>										
SQL where condition	CONTINENT_ID = <%LOOKUP_CONTINENT%> <input type="button" value="v"/> ?										

This applies accordingly for the to be generated filter widgets, but as the widget reference variable for the previous filter element is here used:

CONTINENT_ID = <%LOOKUP_CONTINENT%>

Then our business case should look like this:

Demonstration Apparo Fast Edit Business Case Designer - Data hierarchy example (1) / Data hierar... Anonymous Demonstration

Run Close Save Cancel

Main settings
Standard buttons
Filter data output
Variables
Inserting of new data rows
Editing of data rows
Delete data rows manually

Target table Header Footer Visual Colours Widgets Row ordering Link into Portal Features

+ Add X Delete

Filtering widgets

Row	Column	Column name	Widget type	Title	H
<input type="checkbox"/>	1	> LOOKUP_CONTINENT	> Lookup dropdown (for all tables)	Searching Continent	<input type="checkbox"/>
<input type="checkbox"/>	1	> LOOKUP_COUNTRY	> Lookup dropdown (for all tables)	Searching country FOR <% SQL_FOR_CONTINENT%>	<input type="checkbox"/>
<input type="checkbox"/>	1	> LOOKUP_REGION	> Lookup dropdown (for all tables)	Searching REGION FOR <% SQL_FOR_COUNTRY%>	<input type="checkbox"/>
<input type="checkbox"/>	1	> LOOKUP_CITY	> Lookup dropdown (for all tables)	Searching CITY FOR <%SQL_FOR_REGION%>	<input type="checkbox"/>
<input type="checkbox"/>	1	> LOOKUP_STREET	> Lookup dropdown (for all tables)	Searching STREET FOR <%SQL_FOR_CITY%>	<input type="checkbox"/>

Editing widgets

Column	Column name	Widget type	Title	PK	RO	H	NN
<input type="checkbox"/>	1	> Spacer & Title	>				
<input type="checkbox"/>	2	> LOOKUP_ID	> Input field	> ID	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	3	> LOOKUP_CONTINENT	> Lookup dropdown (for all tables)	> CONTINENT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	4	> LOOKUP_COUNTRY	> Lookup dropdown (for all tables)	> COUNTRY IN	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	5	> LOOKUP_REGION	> Lookup dropdown (for all tables)	> REGION	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	6	> LOOKUP_CITY	> Lookup dropdown (for all tables)	> CITY	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	7	> LOOKUP_STREET	> Lookup dropdown (for all tables)	> STREET	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Show description

In the title of the subordinate filter widgets, we have also added a variable to display the previously selected item.

The SQL variable contains the following Select:

```
SELECT CONTINENT_NAME FROM TESTING.SAMPLE_LOOKUP_CONTINENT WHERE CONTINENT_ID = <%SEARCH_KEY_LOOKUP_CONTINENT%>
```

And the result for the user looks like this:

Demonstration Apparo Fast Edit

BC name:Data hierarchy example

Current language is: en

AFRICA

SEARCH RESET FILTERS

MOROCCO
RSA
ZAMBIA

20 Excel Import

Excel is still one of the most powerful data processing programs.
An ideal way to edit and present data in a simple way.

Unfortunately, Excel has disadvantages, the data is locked in a local file.

Apparo Fast Edit offers several ways for the Excel import. Thus, the data can be tested auditable for errors and transferred in appropriate media (databases).

Excel import options

- With copy & paste directly from an open Excel file (Manual Import)
- By file import via the browser (File Import)
- Through automatic import from defined directories (Auto Import)
- By importing e-mail attachments (E-Mail Import)

The automatic import and import via e-mail attachment always requires a table business case, in which the settings for the (manual) import are defined.

20.1 General Excel Import

General Excel import	Import strategy	Manual import using copy & paste	Excel file import
<input type="checkbox"/> User can use Excel import using copy & paste feature only			
<input type="checkbox"/> User can use Excel file import feature only			
<input checked="" type="checkbox"/> Enable dialog window with result message after Excel data row import			
Dialog window size Width: <input type="text" value="400"/> * Height: <input type="text" value="150"/> *			
Language	Excel row import finish message		
German	Anzahl Zeilen zum Importieren: <%IMPORTED_ROWS%> Zeilen wurden eingefügt: <%INSERTED_ROWS%> Zeilen wurden aktualisiert: <%UPDATED_ROWS%> <input type="button" value="V"/>		
English	Number valid of rows to import: <%IMPORTED_ROWS%> Data rows inserted: <%INSERTED_ROWS%> Data rows updated: <%UPDATED_ROWS%> <input type="button" value="V"/>		

Options

User can use Excel import using copy copy & paste feature only

If enabled then the Business Case has just the Excel import functionality using copy & paste. This means that then directly after Business Case starts the user is seeing only the Excel row import area so this Business Case is just usable for Excel row import.

If this Business Case is using the same primary key(s) like defined in the database table then it is helpful for improving import performance to disable the feature "Check primary key constraints before storing".

User can use Excel file import feature only

If enabled then the Business Case has just the Excel file import functionality. The user can select an Excel file and import the data.

This means that then directly after Business Case starts the user is seeing only the Excel file import page so this Business Case is usable only for the Excel file import.

Enable dialog window with result message after Excel data row import

After an Excel import the user can see a small finishing message. You can alter the text of this message here.

Special Import variables

IMPORTED_ROWS	Count of processed rows
INSERTED_ROWS	Count of inserted rows
UPDATED_ROWS	Count of updated rows

20.2 Import strategy

General Excel import	Import strategy	Manual import using copy & paste	Excel file import
<input checked="" type="checkbox"/> Insert new data row	always		
<input checked="" type="checkbox"/> Update existing data row	always		
Excel import strategy	Import valid rows and ignore invalid rows		
Autocommit after	1000 rows		
<input checked="" type="checkbox"/> Write into a readonly widget too (for Copy & Paste) <input checked="" type="checkbox"/> Write into a hidden widget too (for Copy & Paste) <input type="checkbox"/> Check exact count of decimal places for numeric widgets <input checked="" type="checkbox"/> For file import: User can edit the wrong data rows manually directly in web browser if the whole import file has < 100 wrong data rows			

Insert new data row

If enabled then new data rows (the primary key values of this new data row are not found in the target table) are inserted

There are two options, either a new row is always inserted or only with the prior examination (via JavaScript variable)

always

always

only if script variable returns true

Update existing data row

If this is enabled, existing rows will be overwritten (if the primary key combination is used twice), either always or by variable checking.

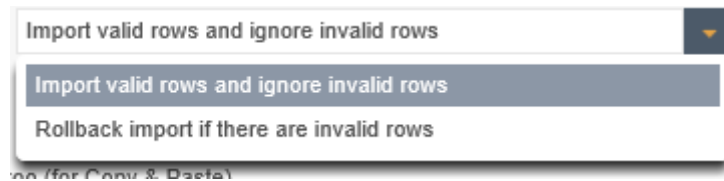
always

always

only if script variable returns true

Excel Import Strategy

With this feature you can configure the behavior of an Excel import.



You can select between a complete rollback if there is invalid content (no data will be imported) or whether only valid content will be imported.

Auto-commit after 1000 rows

Apparo Fast Edit will commit the database transaction after defined number of rows has been processed in Excel import.

If value is 0 or no value is defined then this feature is disabled.

If the above setting is set to "rollback import if there are invalid rows" then this feature is disabled.

Write into a readonly widget too

If enabled then Excel import is overwriting the value of an read-only widget too

Write into a hidden widget too

If enabled then Excel import is expecting a value for a hidden widget too

Check exact count of decimal places for numeric widgets

When enabled then numeric values to be imported must exactly match the specified count of decimal places defined in the widget's data output format setting (must be set to "number").

20.3 Manual Import using copy & paste

This feature enables a direct import of data rows from Excel using copy and paste into this Business Case. The user can mark many Excel rows (even more than 100.000 rows is possible), press the Excel import button and paste it into the text area. Of course the ordering of the Excel columns must be the same like in the Business Case. Read-only and hidden widgets are not used for mapping but are used if they have a constant value. It is not allowed to import Excel cell values that span over multiple rows, in this case use file import.

The screenshot shows the 'Manual import using copy & paste' tab in the APPARO interface. It features a 'Column list description' section with a language dropdown (German/English) and a text area for instructions. Below this is a 'Style' section with options for font face, size, style, align, and colour. The 'Mapping from Excel' section includes a dropdown menu with options like 'using an individual mapping & use widget settings of insert area' and 'like widgets from editing area'. A table below the dropdown lists widget mappings for various fields like PLAN_YEAR, PLAN_MONTH, OFFICE_ID, PRODUCT_LINE_ID, PRODUCT_ID, and SALES.

Widget	Field	Value	Constant value
Input field	PLAN_YEAR	1	
Input field	PLAN_MONTH	2	
Lookup dropdown (for all tables)	OFFICE_ID	3	1
Lookup dropdown (for all tables)	PRODUCT_LINE_ID	4	
Lookup dropdown (for all tables)	PRODUCT_ID	5	
Input field	SALES		

Column list description

You can define a description text that is helpful for the user to know all expected Excel columns.

All variables are useable and HTML tags are possible too.

You can use the internal variable `<%EXPECTED_COLUMNS%>` that has a list of all expected columns using the widget labels.

All hidden or read-only widgets are expecting no Excel column value but the default/constant values of the hidden/read-only widgets are used automatically









Mapping from Excel

If you want to import into different widgets than of the editing/inserting area then you can define an individual mapping for Excel file import too.

20.4 Excel File Import

This feature enables a manual import of a Excel data file into Apparo Fast Edit. The user can select an Excel file and the Business Case is importing the complete file.

Important: You must define a mapping of Excel columns like A,B,C and the associated widget. Just define the Excel column name in the right widget. You can define the count of header rows that must be ignored in tab "File Import".

General Excel import	Import strategy	Manual import using copy & paste	Excel file import										
Header row count	<input type="text" value="2"/>												
Error file format	Same as input file 												
Excel file name	<p>You can define that the name of the Excel file must fulfill a naming convention. For example it must start with "controlling" etc. You can define a regular expression for that. If the setting is empty then file names are not checked. Read more about regular expressions.</p> <table border="1"> <thead> <tr> <th>Language</th> <th>Excel file name</th> </tr> </thead> <tbody> <tr> <td>German</td> <td><input type="text"/></td> </tr> <tr> <td>English</td> <td><input type="text"/></td> </tr> </tbody> </table>			Language	Excel file name	German	<input type="text"/>	English	<input type="text"/>				
Language	Excel file name												
German	<input type="text"/>												
English	<input type="text"/>												
Excel sheet name	<table border="1"> <thead> <tr> <th>Language</th> <th>Excel sheet name</th> </tr> </thead> <tbody> <tr> <td>German</td> <td><input type="text"/></td> </tr> <tr> <td>English</td> <td><input type="text"/></td> </tr> </tbody> </table>			Language	Excel sheet name	German	<input type="text"/>	English	<input type="text"/>				
Language	Excel sheet name												
German	<input type="text"/>												
English	<input type="text"/>												
Data file import description	<table border="1"> <thead> <tr> <th>Language</th> <th>Data file import description</th> </tr> </thead> <tbody> <tr> <td>German</td> <td>Bitte die Excel-Datei für den Import auswählen: </td> </tr> <tr> <td>English</td> <td>Please select Excel file to import: </td> </tr> </tbody> </table>			Language	Data file import description	German	Bitte die Excel-Datei für den Import auswählen: 	English	Please select Excel file to import: 				
Language	Data file import description												
German	Bitte die Excel-Datei für den Import auswählen: 												
English	Please select Excel file to import: 												
Mapping strategy	Using Excel column names like A,B,C 												
Mapping Excel to widgets	<p>To import data from an Excel file it is necessary to map all necessary Excel columns to widgets. If a widget has no Excel mapping then it is not using the default value and the constant value depending on the widget settings. The Excel file import is using the widget settings of the insert area only.</p> <table border="1"> <thead> <tr> <th>Widget type</th> <th>Database column</th> <th>Excel column name (like B or BA)</th> <th>Default value</th> <th>Constant value</th> </tr> </thead> <tbody> <tr> <td>Input field</td> <td>VALID_FROM</td> <td><input type="text"/></td> <td></td> <td><% CURRENT_DATE% ></td> </tr> </tbody> </table>			Widget type	Database column	Excel column name (like B or BA)	Default value	Constant value	Input field	VALID_FROM	<input type="text"/>		<% CURRENT_DATE% >
Widget type	Database column	Excel column name (like B or BA)	Default value	Constant value									
Input field	VALID_FROM	<input type="text"/>		<% CURRENT_DATE% >									


Options

Header row count

This number of rows are ignored during the import

Error file format

Format of a file containing errors and their descriptions, a user can download after an import that finished with errors.



Data file import description

Contains the text of the „Choose file“ dialog

Mapping Excel to widgets

For the import a mapping necessary. This mapping is defining all Excel columns that must imported using this Business Case.

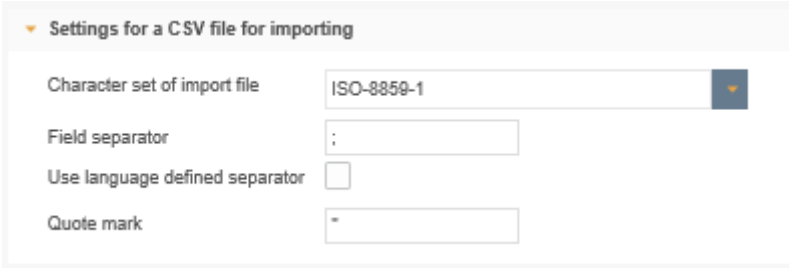
If you input for example Excel column F for the 1. widget then all values of Excel column F will be imported into 1. widget.

If the Excel document has no value in a cell and the mapped widget has a default value defined then Apparo Fast Edit is using automatically the default value.

If a widget has a constant value then this value will be used for import depending on the setting (use constant value in insert case only or in insert/update case).

Settings for a CSV file for importing

When importing CSV files additional settings are required.



▼ Settings for a CSV file for importing

Character set of import file	ISO-8859-1
Field separator	;
Use language defined separator	<input type="checkbox"/>
Quote mark	"

Character set of import file

Enthält eine Liste der verfügbaren Zeichensätze.

Field separator

Definition of the field separator. Using the next setting it is possible to use an own separator for each used language.

Hint: If you need tab character as separator use "\t"

Use language defined separator

When checked then Apparo Fast Edit detects language type from file (for example from filename_en.xlsx as "en") and uses separator defined for detected language from language messages.

Quote mark

Definition of the quote mark - character used to enclose fields containing a separator, usually "

20.5 Automatic Excel file import

It is possible to import files automatically which are accessible by the server (for example Excel files). In this case Apparo Fast Edit checks whether files according to a specific file mask are available in a file directory of your choice.

If yes, these files will be imported into the working directory. After the import these files are stored into the history file directory.

The settings "Field Separator" and "Header Row count" are used for manual file import too.

The time interval for looking into the feeding directory is definable in 'Global Settings'. Automatic import means that the business don't needs to be started. After enabling the features the import happens within the entered time period.

If the feeding directory is located on a second server, the Windows user who starts AFE must have the rights to access the directory on this second server.

All Excel file import settings are also used during automatic Excel file import

General Excel import	Import strategy	Manual import using copy & paste	Excel file import	Automatic Excel file import
All Excel file import settings are also used during automatic Excel file import				
Source file directory	<input type="text"/>			
File Mask	<input type="text" value="*.csv"/>			
Working file directory	<input type="text"/>			
Error File Directory	<input type="text"/>			
Error filename template	<input type="text" value="error_<%IMPORTED_FILE_NAME%>"/>			
History file directory	<input type="text"/>			
History filename template	<input type="text" value="<%IMPORTED_FILE_NAME%>_<%TIME_MS%>"/>			
Language	<input type="text" value="English"/>			

Options

Source file directory

Defines the file directory in which Apparo Fast Edit looks for files to import. It is looking after each 'n' seconds into this directory.

The path can be: \\servername\folder1\folder2 or x:\folder1\folder2 or <%VARIABLENAME%>\folder or <%VARIABLENAME%>. The variable must deliver a correct path.

If empty then no automatic import will occur on this Business Case.

File Mask

The importable files must have the defined file mask.

File mask can contain wildcards ? and *.

Example: *.xls

Caution: If the filename matches the file mask of multiple Business Cases, the used Business Case will be random. If empty, no automatic import will occur on this Business Case.

Working file directory

Optionally Apparo Fast Edit is moving the files to the 'Working file directory' first and then the import process is starting.

If empty, working file and directory will be the same as the feed file.

Error File Directory

File directory for error files with the error messages. If empty, no error results will be stored.

Error filename template

Template file name for error files.

The usage of variables is possible, for example:

<%IMPORTED_FILE_NAME%>

name of the imported file (without path)

<%TIME_MS%>

numeric (UNIX) timestamp

If empty, no error results will be stored.

History file directory

After import the files are moved into this file directory. If empty, no history will be stored.

History filename template

Template file name for history files.

Mask can contain placeholders <%PlaceholderName%> where PlaceholderName is one of:

IMPORTED_FILE_NAME name of the imported file (without path)

TIME_MS numeric (UNIX) timestamp

If empty, no history will be stored.

Language

Language definition (important for formatting such as formatting of date).

21 Conditional formatting

21.1 Background color

21.1.1 Widget background

You will find the conditional settings for the widget background in the tab 'Visual':

Click on add and define the color you want.

In the variable you can calculate anything you want, important is that the variable returns the string 'true' if the condition hits.

Demonstration Apparo Fast Edit
Business Case Designer - SAMPL WF Conditional / SAMPL

Widget settings of database column FORECAST2

Widget type

Mapping & Other

Features

Visual

Help texts

Data output format

Column label

Language	Column label
German	Quartal 1 V
English	Quarter I V

Visual settings

Label style

Font face: Arial ▼

Size: 11 ▼

Style: Bold ▼

Align: Left ▼

Colour: #000000 ▼

Background colour for label: No background colour is defined
ADD

Widget align: Left ▼

Column width (px): 80 *

Border colour: #D0D0D0 ▼

Background colour:

if value of variable <%blue%>

is string 'true' ✕

if value of variable <%yellow%>

is string 'true' ✕

ADD

Example for the blue variable:

```
var z = <%FORECAST%>;
if (z >= 100){var x = 'true';}
else {var x = 'false';}
x;
```

21.2 Row background color

You find this setting in the Business Case edit view, tab Colours:

Target table
Header
Footer
Visual
Colours
Widgets
Row ordering
Link into Portal

Panel colours

Window background	#FFFFFF		
Filter area background	#C6D9F0		
Bulk update area background	#FFFFFF		
Insert area background	#FFFFCC		
Message dialog background	#FFFFFF		
Calculation area background	#FFFFFF		

Other colours

Mandatory mark	#FF0000		
Read-only background	#CCCCCC		
Error background	#FFA0A0		
Modified row	#FFFF00		
Hint text colour	#CCCCCC		

Data row background colour

Data row background colour is	#00B050	if value of variable	<%green%>	is string 'true'	X
Data row background colour is	#FF0000	if value of variable	<%red%>	is string 'true'	X

ADD

Click on add and define the colour you want.

In the variable you can calculate anything you want, important is that the variable returns the string 'true' if the condition hits

Result:

Demonstration Apparo Fast Edit										
Conditional formatting demo										
Row condition: Amount/Year < 100 red ; >100 green; 100 = no hit										
Column condition: Quarter I < 100 yellow ; >100 blue; 100 = no hit										
<input type="checkbox"/>	Year	Product line	Product	Amount/Year	Quarter I	Quarter II	Quarter III	Quarter IV	Last changed by	Last change from
<input type="checkbox"/>	2017	T-Shirts	T-Shirt October	99	100	101	10	10	administrator	03.31.2017
<input type="checkbox"/>	2017	T-Shirts	T-Shirt Vienna	10	100	100	100	100	administrator	03.29.2017
<input type="checkbox"/>	2017	Bikinis	Bino Man	100	1010	0	0	0	administrator	03.29.2017
<input type="checkbox"/>	2017	Bikinis	Joki	100	100	0	0	0	test	08.21.2017
<input type="checkbox"/>	2017	Bikinis	Illo	100	100	100	100	100	administrator	03.29.2017
<input type="checkbox"/>	2017	Underwear	Lino outdoor	102	100	100	100	100	administrator	04.06.2017
<input type="checkbox"/>	2017	Caps	Blue Cap	505	101	100	100	100	administrator	03.29.2017
<input type="checkbox"/>	2017	Caps	Yellow Cap	99	100	100	100	100	administrator	03.29.2017
<input type="checkbox"/>	2017	Caps	Dark Cap	499	99	100	100	100	administrator	03.29.2017

21.3 Conditional formatting using HTML & JavaScript

The goal is to have following output:

<input type="checkbox"/>	Product line	Product	Month	My status	Revision - Status Controlling	Plan data	Trend
<input type="checkbox"/>	Trousers	Hemfort	*08-2011	Ready for approval	OPEN	1400	↑
<input type="checkbox"/>	Trousers	Hemfort	*08-2011	Ready for approval	ACCEPTED	400	↓
<input type="checkbox"/>	T-Shirts	T-Shirt Holiday	*08-2011	Ready for approval	OPEN	123	↓
					OPEN		
<div> <div>Page: 1 / 2</div> <div> <input type="button" value="OK"/> <input type="button" value="Excel Row-Import"/> <input type="button" value="CSV File Import"/> <input type="button" value="Export to Excel"/> <input type="button" value="Cancel"/> <input type="button" value="Close & Save"/> </div> </div>							

This example shows how we created the colored text for the widget 'Revision – Status Controlling' and the arrows for the widget 'Trend'

21.3.1 Colored text for Status Controlling

This is generated by a JavaScript. For the output we are using 'label with variables' containing the variable.

Variable: <%conditional%>

var z = <%STATE_REVISION_ID%>;

if (z == 3){var x = 'DECLINED';}

else if (z == 2){var x = 'ACCEPTED';}

else {var x = 'OPEN';}

x;

21.3.2 Arrows for the ,Trend' widget

This is generated by a JavaScript. For the output we are using 'label with variables' containing the variable. We inserted the arrow images into \Apparo\FastEdit\clients\images

Script-Variable: <%trend%>:

```
var trend = <%FORECAST%>;
if (trend > 1000)
    {var x = '';}
else if (trend < 1000)
    {var x = '';}
else
    {var x = '';}
x;
```

22 Movies

Please find a collection of Videos about Apparo Fast Edit here

Movies.apparo.solutions

23 Support

If you need support please write an email to:

support@apparo.info