TEC-IT

WWW.TEC-IT.COM

TFORMer Designer

Label and Report Designer

Version 6.0

User Manual

22 July 2010

TEC-IT Datenverarbeitung GmbH
Wagnerstrasse 6
A-4400 Steyr, Austria

t ++43 (0)7252 72720
f ++43 (0)7252 72720 77
office@tec-it.com
www.tec-it.com

1 Content

1.1 Table of Figures

1.2 List of Tables

2 Disclaimer

The actual version of this product (document) is available as is. TEC-IT declines all warranties which go beyond applicable rights. The licensee (or reader) bears all risks that might take place during the use of the system (the documentation). TEC-IT and its contractual partners cannot be penalized for direct and indirect damages or losses (this includes non-restrictive, damages through loss of revenues, constriction in the exercise of business, loss of business information or any kind of commercial loss), which is caused by use or inability to use the product (documentation), although the possibility of such damage was pointed out by TEC-IT.

TEC-IT Datenverarbeitung GmbH
Wagnerstr. 6

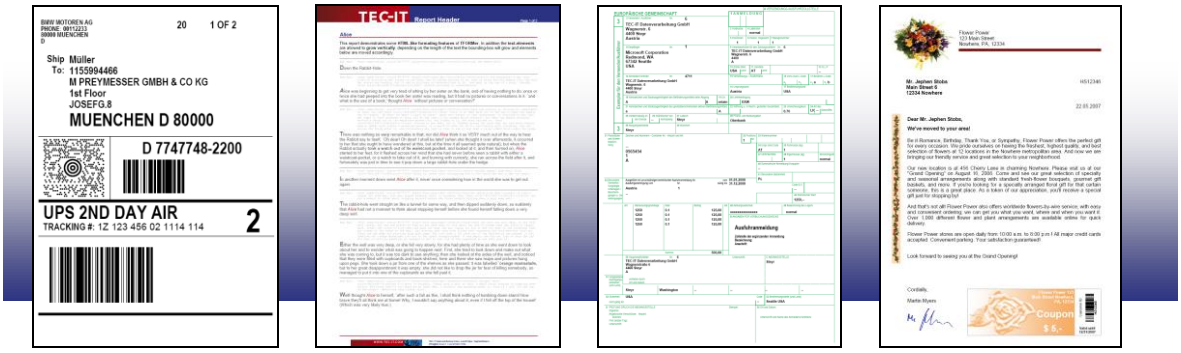
A-4400 Austria
t.: +43 (0)7252 72720
f.: +43 (0)7252 72720 77
http://www.tec-it.com

3 Introduction

3.1 Professional Documents Created With Ease

Many thanks for evaluating **TFORMer**!

The **TFORMer** product family represents a complete, lean and powerful solution for generating arbitrary documents. It combines the features of barcode labeling tools with the characteristics of report generators into a unified printing-solution. It provides *professional layout and output* capabilities, an integrated *barcode generator,* full-featured *UNICODE* support and it supports direct *PDF generation.* It can be used on client and on server side and it is available for all major operating systems.



The functionality of **TFORMer** is available for end-users as well as for software developers:

End-Users benefit from the graphical layout editor with instant output capabilities: **TFORMer Designer** offers outstanding design and printing features for reports, tables, lists, serial letters, industry forms, vouchers and barcode labels. Furthermore the numerous ready-to-use label and report templates for industry and logistics (e.g., VDA-4902, Odette, GALIA, AIAG, …) will be a valuable assistance.

Software developers use **TFORMer SDK** as reporting tool for direct printing, PDF generation, PostScript-, HTML, image or ZPL-II output. Layouts are designed graphically with **TFORMer Designer.** The **TFORMer SDK** is then used to provide dynamic data for the layouts and to generate the output. This core functionality for printing and output is available for all major operating systems.

3.2 TFORMer Basics

In contrast to a word processor which is mostly used for static content, **TFORMer** generates *output based on dynamic data*. Examples for dynamic contents are article-numbers used in product labels or addresses used in serial letters.

Dynamic content is provided by an external data source like a database, a file, an arbitrary application or by the user. Such an external data source is read by **TFORMer** during printing. So-called data fields are used to access the external data. They serve as placeholders for dynamic content in the layout. To simplify usage, **TFORMer** provides a mapping mechanism to bind external data to the data fields used in a layout.

This data-centric output scheme is maintained in all cases. Even when printing pure static documents (like personal address labels without any dynamic data) a data source is required for controlling the number of printed labels.

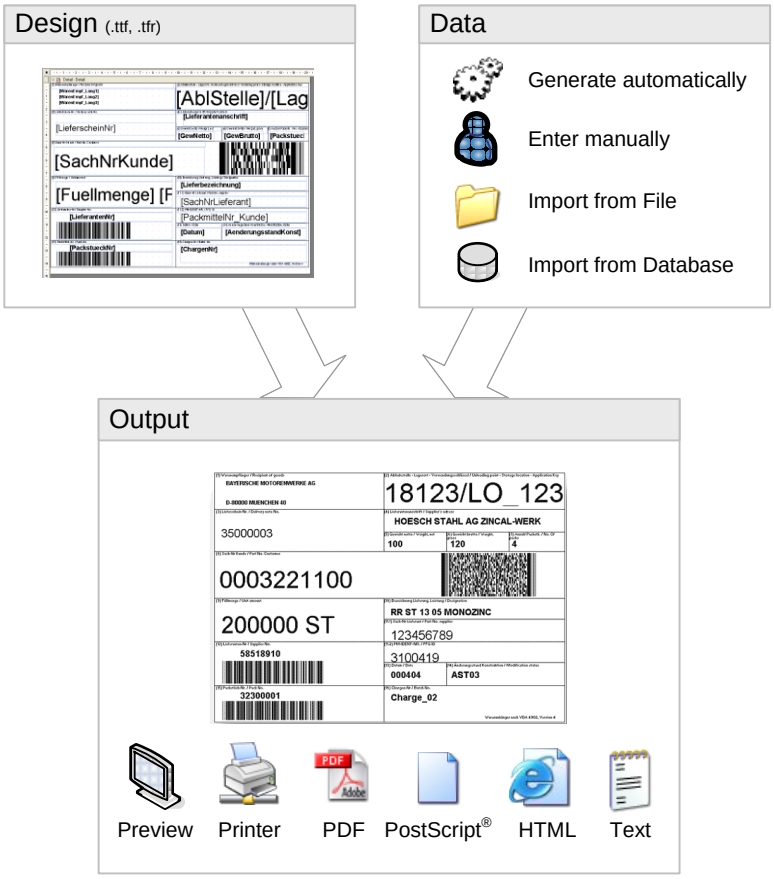


Figure 1: Architectural Overview

Layouts created with **TFORMer Designer** are able to perform computations and to react on the values of datafields. Thus, **TFORMer** enables you to embed the complete output logic of a document into the layout.

3.3 Output Formats

Once a layout was designed it produces identical results across printer models, operating systems and formats. The following formats are supported:

- **Direct Printing**
  Print layouts directly on all printers supported by Microsoft® Windows®.
- **PDF**
  PDF export with full-featured barcode support, Unicode and font embedding.
- **PostScript®**
  Used for printing under Linux/UNIX and for pre-press applications.
- **HTML**[1]
  The built-in HTML output is ideal for previewing and for web-based applications.

[1] Due to the nature of HTML some output features may be limited or not available.

- **Image Formats**
  The built-in image output supports BMP, GIF, JPG, PCX, TGA, PNG and TIF formats (including multipage TIF).
- **ZEBRA®**
  Print to ZEBRA® printers without any additional driver. ZPL-II output is generated directly.
- **ASCII**
  Pure ASCII output without any graphics for special purpose requirements.

3.4 System Requirements

3.4.1 Operating Systems

TFORMer Designer requires one of the operating systems listed below. It works with 32-bit and 64-bit operating systems.

- Microsoft® Windows 2000
- Microsoft® Windows XP
- Microsoft® Windows Vista
- Microsoft® Windows 7
- Microsoft® Windows Server 2003 (including Terminal Server)
- Microsoft® Windows Server 2008 (including Terminal Server)

TFORMer Designer includes the output kernel of the **TFORMer SDK**. This output kernel is available for Linux® and UNIX® as well. For details check out the Developer Manual.

3.4.2 Memory

512 MB RAM is the recommended minimum (1 GB for Windows Vista or higher). Actual memory requirements depend on the type and size of the generated output.

3.4.3 Disk Space

TFORMer needs approximately 30 MB on disk. When using the .NET based API of the **TFORMer SDK** or the printing utility **TFORMer QuickPrint** please also consider .NET 2.0 requirements.

4 User Interface

This chapter will give you an overview of the **TFORMer** user interface.

4.1 Main Window

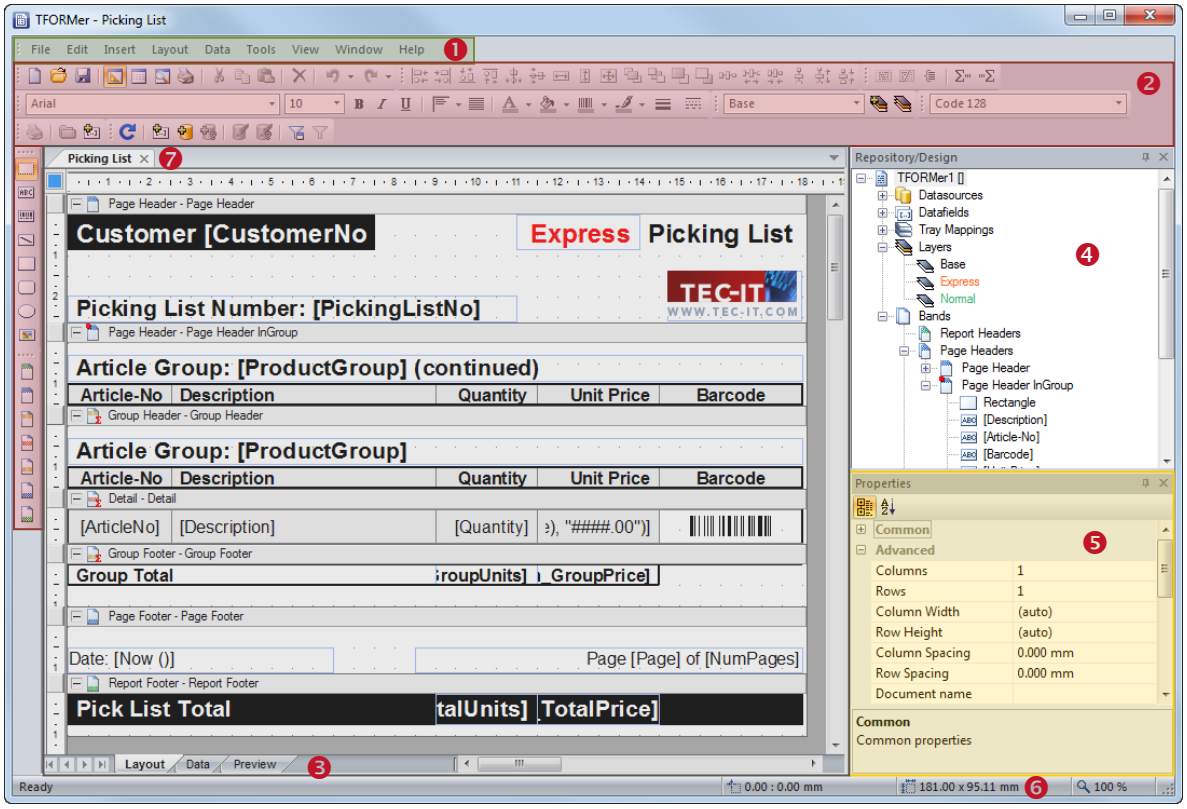


Figure 2: User Interface

The *main window* is divided into the following areas:

1. Menu (see 4.2).
2. Toolbars (see 4.3).
3. Layout View, Data View and Preview (see 4.4).
4. Design Tree Window (see 4.5).
5. Properties (see 4.6).
6. Status Bar (see 4.7).
7. Layout Tabs (see 4.8).

4.2 Menu

The *menu* gives you access to the following functions:

4.2.1 File

New Form…	Create a new layout. Choose a template and adopt it to your needs. Shortcut: *Ctrl+N*
Open…	Open an existing layout (or repository). Shortcut: *Ctrl+O*
Save	Save the current layout. Shortcut: *Ctrl+S*
Save as…	Save the current layout with a new name. You can also save layouts as ZIP-files.
Close Form	Close the current layout. Shortcut: *Ctrl+W* or *Ctrl+F4*
Templates	Templates can be used as basis for new layouts. ▪ Save as Template… Save the current layout as template. ▪ Organize Templates… Organize the templates in a hierarchical structure.
Layout View	Switch to the layout view (see section 4.4.2). Shortcut: *Ctrl+L*
Data View	Switch to the data view (see section 4.4.3). Use this view to manually edit or import data from external data sources. Shortcut: *Ctrl+D*
Preview	Switch to the print preview (see section 4.4.4). Shortcut: *Ctrl+Space*
Page Setup…	Choose the margins and the orientation of the page.
Printer Setup…	Choose a printer and edit the printer properties.
Print…	Print the layout on a printer of your choice or generate file output (PDF, PostScript®, HTML, Text). Shortcut: *Ctrl+P*
Repository	A repository is a central location which references multiple layouts in a structured way (see chapter 16). ▪ New Repository… Create an empty repository. ▪ Open Repository… Open an existing repository. ▪ Close Repository Close an open repository.
Exit	Exit **TFORMer.**

4.2.2 Edit

Undo	Revert the last change. Shortcut: *Ctrl+Z*
Redo	Revert the last *Undo.* Shortcut: *Ctrl+Y*
Cut	Copy all selected objects (rectangles, texts, printing bands, datafield definitions, ...) to the clipboard and remove them from the layout. Shortcut: *Ctrl+X*
Copy	Copy all selected objects to the clipboard. Shortcut: *Ctrl+C*
Paste	Insert the content of the clipboard. Shortcut: *Ctrl+V*
Delete	Remove all selected objects. Keyboard: *Del*
Element-Content…	Edit the content of a text, barcode or picture element. Shortcut: *Double-Click* the Element or press *F2*
Printing-Condition…	Edit the printing condition for the selected band or element (see also section 12.2).
Group-By…	Edit the group-by expression for the selected “group header” or “group footer”.

Pre-Evaluation…	Edit the pre-evaluations for the selected band (see also section 12.4).
Post-Evaluation…	Edit the post-evaluations for the selected band (see also section 12.4).

4.2.3 Insert

Project	Insert a new project into the repository.
Form…	Insert a new layout into the repository.
Data	Insert a new ▪ Datafield… Create a new datafield (see also section 10.2.1.1). ▪ Datasource… Create a new data source (see also section 13.3.2.1). ▪ Source-Parameter… Create a new source parameter (see also section 13.5.2.1).
Tray-Mapping	Insert a new tray mapping (see also section 12.6).
Layer…	Insert a new layer (see also section 12.3).
Band	Insert a new band (page header, detail band, …) – see also chapter 7.
Element	Insert a new element (text, barcode, line, …) – see also chapter 8.

4.2.4 Layout

Layout View	Switch to the layout view (see section 4.4.2). Shortcut: *Ctrl+L*
Z-Order	Overlapping elements can be re-ordered: Send the selected element(s) to the front or to the back (see also section 8.2.10.3).
Align	Align two (or more) selected elements (see also section 8.2.10.1).
Size	Adjust two (or more) selected elements in size (see also section 8.2.10.2).
Horizontal Spacing	Enlarge or reduce the horizontal distance between elements within a selection (see also section 8.2.10.4).
Vertical Spacing	Enlarge or reduce the vertical distance between elements within a selection (see also section 8.2.10.4).
Band Order	Re-arrange the order of selected bands (see also section 7.3.6). Shortcuts: *Alt+↑* and *Alt+↓*

4.2.5 Data

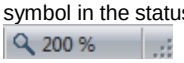
Data View	Switch to the data view (see section 4.4.3). Use this view to manually edit or import data from external data sources. Shortcut: *Ctrl+D*
Refresh	Reload the data from the data source. Shortcut: *F5*
New Datafield…	Insert a new data field (see also section 10.2.1.1).
New Datasource…	Create a new data source (see also section 13.3.2.1).
Edit Datasource…	Edit the current data source (see also section 13.3.2.3)
New Source-Parameter…	Insert a new source-parameter (see also section 13.5.2.1).
Edit Source-Parameter…	Edit a source-parameter (see also section 13.5.2.3).
Edit Filter…	Edit the filter (see also section 13.6.1).
Apply Filter	Apply the filter (see also section 13.6.2).

4.2.6 Tools

Customize…	Customize the toolbar (not available in this version).
Options…	Open the options dialog window (see chapter 17).

4.2.7 View

Layout	Switch to the layout view (see section 4.4.2). Shortcut: *Ctrl+L*
Data	Switch to the data view (see section 4.4.3). Use this view to manually edit or import data from external data sources.

	Shortcut: *Ctrl+D*
Preview	Switch to the print preview (see section 4.4.4). Shortcut: *Ctrl+Space*
Layer Colors	Toggle the display color of the elements between layer colors and element colors (see also section 12.3.3).
Status Bar	Show or hide the status bar (see Figure 2, ❻).
Application Look	Select the appearance of the user interface. Choose between Office XP, Windows XP and Office 2007 (blue, black, aqua, silver) styles.
Properties	Show or hide the properties window (see Figure 2, ❺).
Design Tree	Show or hide the design tree window (see Figure 2, ❹).
Toolbar	Show or hide single toolbars (see Figure 2, ❷).
Zoom	Select the zoom factor. You will find the selected zoom factor shown next to the zoom symbol in the status bar:  Shortcut: *Ctrl++*, *Ctrl+–* or *Ctrl+Mousewheel*

4.2.8 Window

Close All Documents	Close all opened documents.
Document List	At the bottom of the *Window* menu all open documents are listed with their names. To select one of the documents, click at the appropriate menu entry.

4.2.9 Help

TFORMer Help	Display the **TFORMer** manual. Keyboard: *F1*
Online FAQ	Open the default browser and navigate to the **TFORMer Designer** frequently asked questions web page.
License…	Open the License dialog for **TFORMer Designer** (see chapter 18).
Check for Update…	Check if an update of **TFORMer Designer** is available.
Buy TFORMer Online	Open the default browser and navigate to the online order form.
TEC-IT Website	Open the default browser and navigate to http://www.tec-it.com.
About TFORMer…	Display information about **TFORMer Designer** and the program version.

4.3 Toolbars

The most commonly used program functions are accessible via *toolbars*. Each of the toolbars can be shown or hidden (menu *View* ► *Toolbar*).

The following toolbars are available:

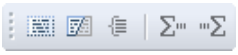
Default

(New/Open/Save File, Layout View, Data View, Preview, Print, …)

Layout

(Alignment, Size, Z-Order, Spacing)

Content and Expressions



(Element-Content, Printing Condition, Group-By, Pre-Evaluation, Post-Evaluation)

Format

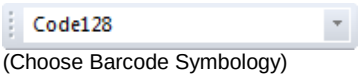
(Font and Font size)

(Font style, Text Alignment, Colors, Line styles, …)

Layer

(Assign Elements to a Layer, Add Layer, Layer Colors)

Barcode



(Choose Barcode Symbology)

Data

(Refresh, New Datafield, New Datasource, New Source-Parameter, Edit Datasource, Edit Source-Parameters, Edit Filter, Apply Filter)

Repository

(New Printer, New Project, New Datafield)

Elements

(Insert Graphical Elements)

Bands

(Insert different kinds of Bands: Report Header, Page Header, Detail Band, …)

4.4 Layout View, Data View and Preview

4.4.1 Introduction

TFORMer offers three different views for a single form.

The default view is the *layout view* (❶). It is used to create the layout. The layout usually utilizes datafields (like *[Printed Items]* in the layout below) as placeholders for the actual data. For details regarding datafields, please refer to chapter 10.

The data for the datafields is provided by a datasource. It can be edited in the *data view* (❷).

TFORMer also offers a *preview*, which instantly renders the resulting output using the layout and the data of the adjusted datasource (❸).

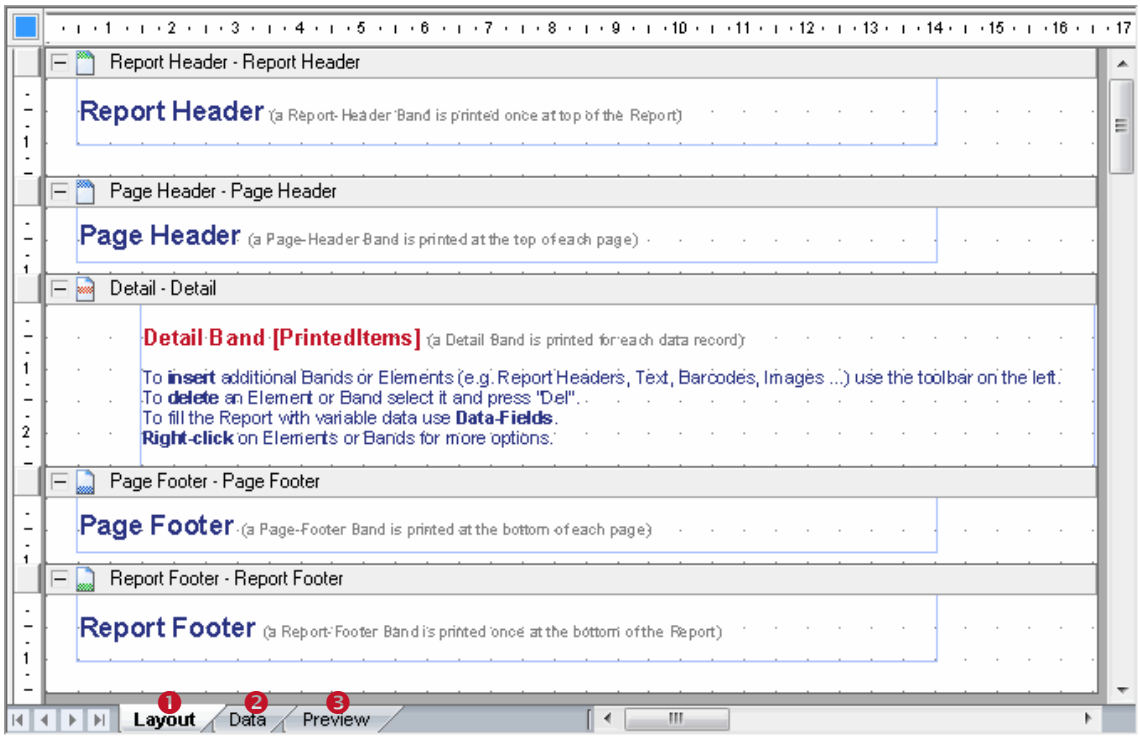


Figure 3: Layout View

You can switch between the views by pressing the corresponding tabs or by using the *File* or *View* menus (see 4.2.1 and 4.2.7).

4.4.2 Layout View

The *layout view* is used to create layouts:

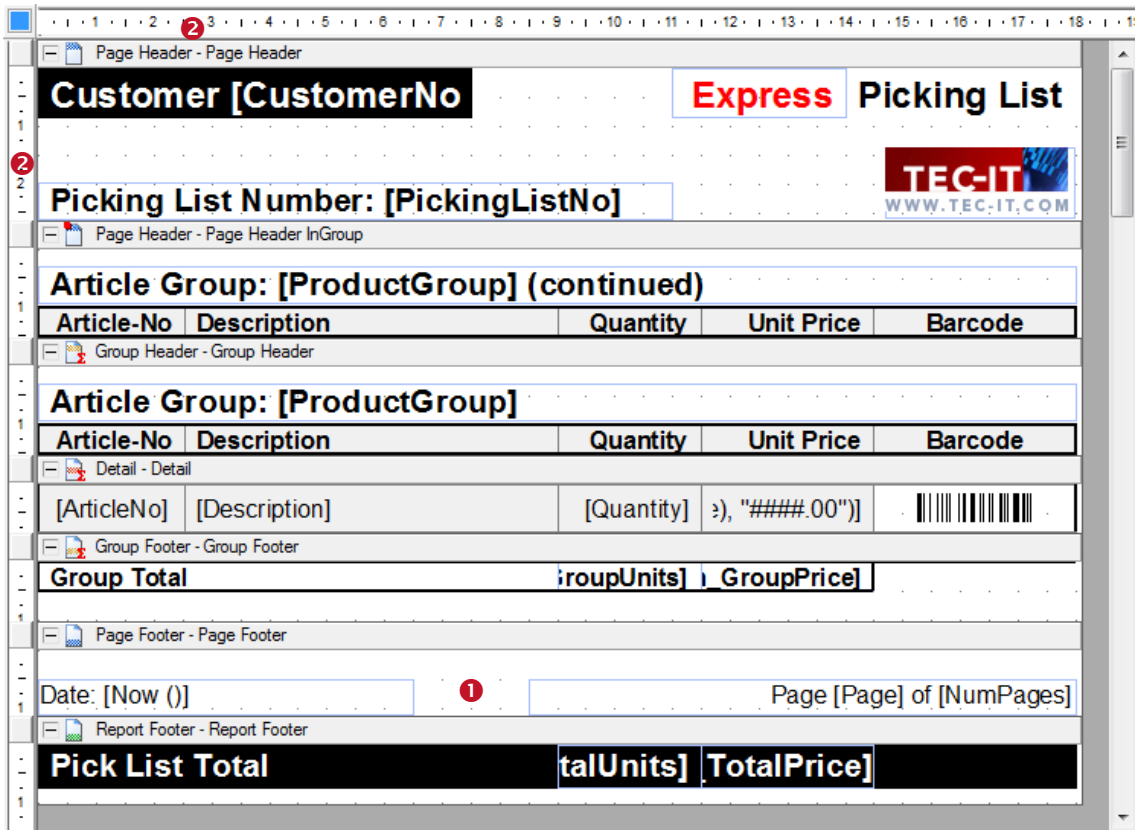


Figure 4: Layout View

On the worksheet ❶ you can place all kinds of graphical elements like text elements, lines, rectangles, barcodes and images. Such elements can be moved, resized or deleted. Furthermore, you can add and remove single print areas like page headers and page footers. These areas are called bands (see chapter 7).

On the left and on the top of the layout there are rulers (❷) which assist you when positioning elements. After starting **TFORMer Designer** for the first time, the measuring unit is set to *"System (Default)"*. You can change this value in the options dialog (menu *Tools* ► *Options…*).

It is possible to zoom the layout using the menu *View* ► *Zoom* or the hotkeys *Ctrl++* and *Ctrl+-*. You can also adjust the zoom factor by holding down the *Ctrl* key and by simultaneously turning the mouse wheel.

For more information on how to design layouts, please refer to chapters 6 to 12.

4.4.3 Data View

The *data view* is used to administrate datasources. You can create, switch and modify datasources. Furthermore, the content of the active datasource is shown. This is the data which is used for printing the layout.

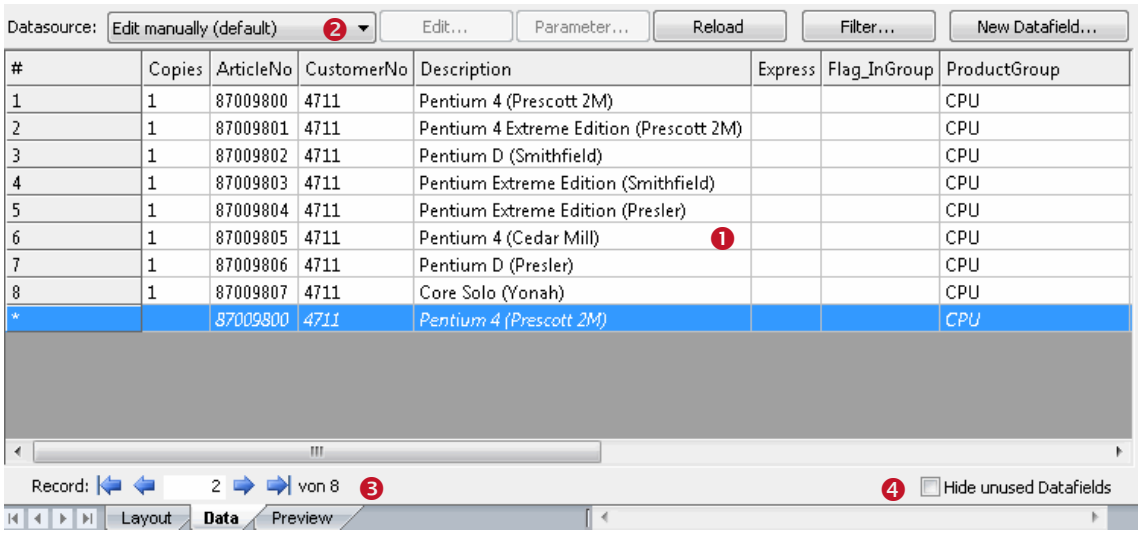


Figure 5: Data View

The *data view* mainly consists of a data grid showing the current data (❶). Use the navigation buttons in ❸ to select rows or to jump to a specific row in the datasource. In ❷ you will find functions for manipulating and selecting datasources. When checkbox ❹ is activated all datafields which are not used in the layout will be hidden.

By default **TFORMer** assigns a manual datasource to a newly created layout. This means that you can enter values for datafields manually in the grid. It is also possible to create and edit other types of datasources (e.g., ODBC database access) in this view. For more information regarding the data view, please refer to section 13.2.

4.4.4 Preview

This tab shows an instant *preview* of the output using the selected datasource.

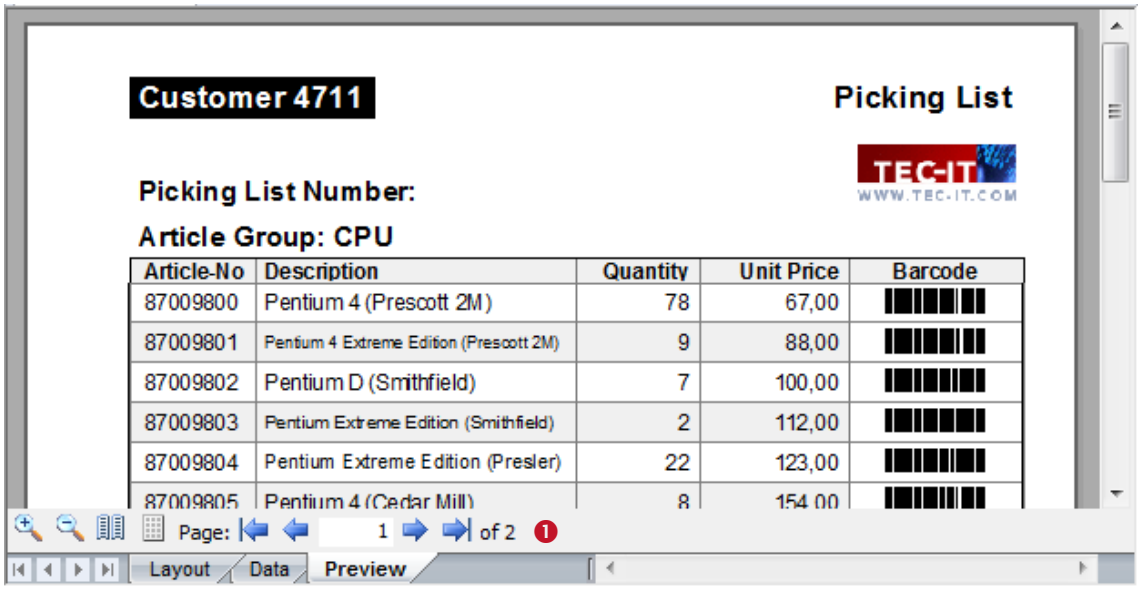


Figure 6: Preview

Use the functions in ❶ to change the zoom factor, switch between single page or double-page preview, show/hide label boundaries, navigate between pages or to jump to a specific page. For details, please refer to chapter 14.

4.5 Design Tree Window

4.5.1 Layout Tree

The *design tree window* shows a structured view of the layout. It lists all objects in a single place and allows the selection of objects per mouse click. Even such elements, which are occluded in the design view, can be selected in the design tree easily.

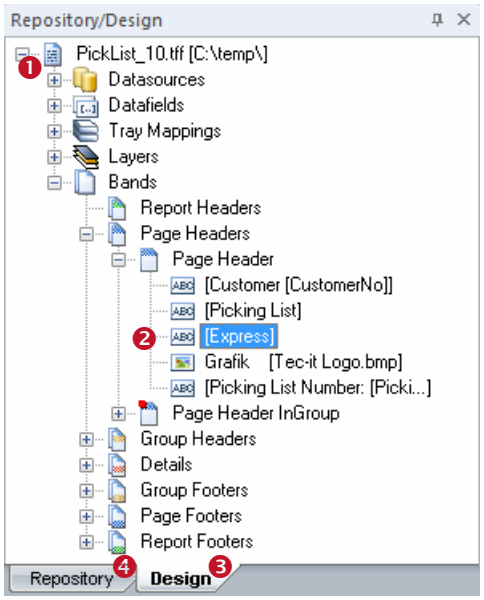


Figure 7: Design Tree Window

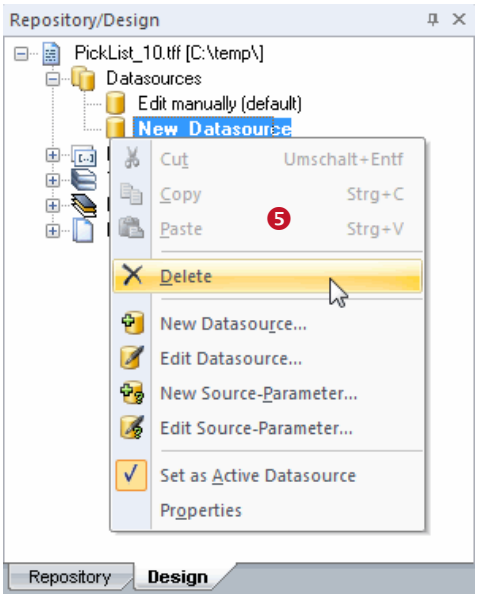


Figure 8: Delete a Datasource

The design tree window ❶ lists all objects in a hierarchical tree structure. You can expand single branches of the tree by clicking the ⊞ symbol and you can collapse them by clicking the ⊟ symbol.

To select an object, click on it with the left mouse button. Hold down the *Shift* key or the *Ctrl* key for multiple selection. Figure 7 shows a selected text element (❷), which is located in the page header of a layout called “PickList_10.tff”.

If you right-click on a tree item a context menu will appear (see Figure 8, ❺). The context menu offers various functions for the selected object.

If **TFORMer** encounters an error in a user defined computation, the affected item is drawn red. For more information, please refer to section 10.4.

4.5.2 Repository Tree

If a repository is used, a second tab will be shown next to the *Design* tab ❸. Clicking tab ❹ switches to the tree view of the repository. A repository is a central database for layouts, datafield definitions and tray-mappings. For details, please refer to chapter 16.

4.6 Properties

The complete layout structure in **TFORMer** is based on objects (bands, text boxes, datasources, etc.). To adjust the properties of such objects the properties window is used. For a completive list of all object properties, please refer to Appendix A.

The properties of the following objects can be modified:

- Forms
- Bands
- Elements (graphical design elements)
- Datasources and Source-Parameters
- Datafields
- Layers
- Tray-Mappings
- Projects (Repository)

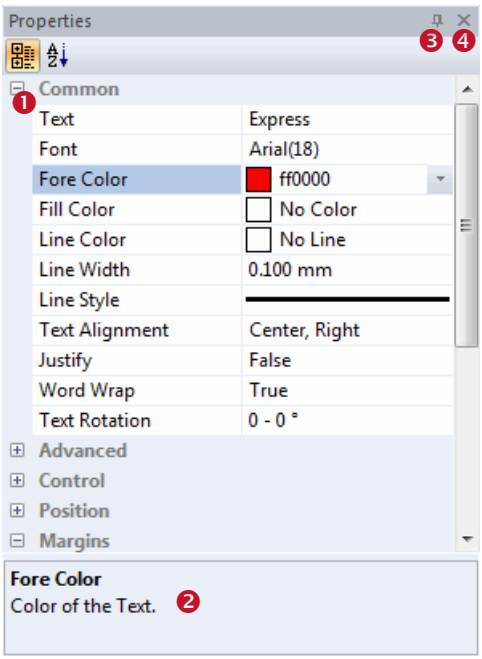


Figure 9: Properties Window

All properties are grouped into categories (like *"Common", "Advanced"* or *"Position"*, see ❶). The categories help you to locate properties more easily.

Properties are accessible as "field:value" pairs. E.g., the property *"Text"* is currently set to "Express". The color (*"Fore Color"*) is set to red.

The area ❷ shows information on the selected property.

Button ❸ is used to turn the auto-hide feature of the properties window on or off. Click the symbol once to make the window slide away. To recover its previous state, hover with the mouse over the Properties button. By clicking the auto-hide button again, the window will be fixed in its previous position.

Button ❹ closes the properties window.

► **TFORMer Designer** supports multiple object selection. This can be used to change the properties of multiple objects simultaneously. A change will be applied to all objects within the selection.

4.7 Status Bar

The status bar displays information about position and measurement of the selected design element. Furthermore the zoom factor of the layout view is shown.

Figure 10: Status Bar

❶ shows the coordinates of the upper left corner of the selected element (or of the element with the mouse focus).

❷ shows the dimensions of the selected element (or of the element with the mouse focus). The measuring unit used for displaying positions or dimensions can be adjusted in the options dialog (menu *Tools* ► *Options…*).

❸ shows the current zoom factor used for displaying the content of the layout view. The zoom factor can be adjusted via the menu (*View* ► *Zoom*) or by holding down the *Ctrl* key and by simultaneously turning the mouse wheel.

4.8 Layout Tabs

When working with multiple layouts, each of these layouts is represented by a tab (see ❶). To switch between layouts click on the corresponding tab in ❶ or use the layout picker ❷. To close the current layout click on the ✕ symbol inside the tab.

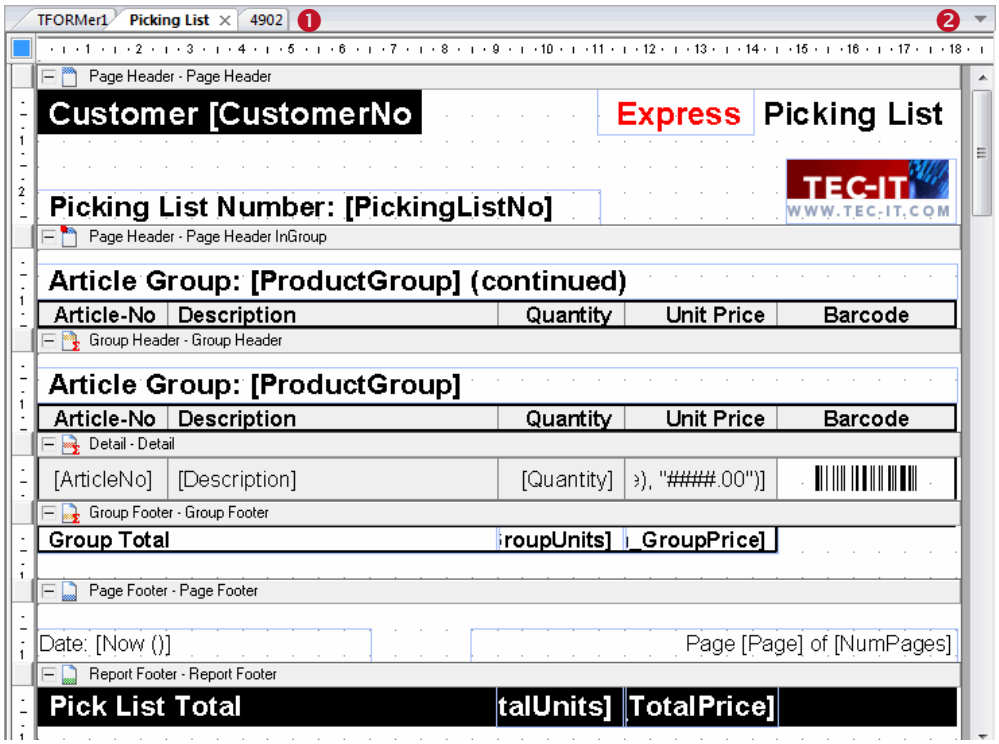


Figure 11: Change layouts quickly by using tabs

5 Printing Concept and Workflow

5.1 Introduction

TFORMer uses one universal output concept to produce all different kinds of outputs like:

- Reports – a table, a list or a letter usually spanning multiple pages.
- Labels (Normal Printer) – a layout which is printed repeatedly in multiple rows and columns on each page.
- Labels (Label Printer) – a layout which is printed once on each page. One page is one label.
- Hybrid Layouts – a mix of the first two variants.

The output concept is based on the logical separation of the *layout* (❶) into different printing areas (e.g., report header, page header, detail area, page footer and report footer). Such a printing area is called *band.*

Each of these bands fulfills a specific purpose. By combining bands and by adjusting their properties, different layout schemes can be produced. The figure below illustrates the printing of a simple report (❷).

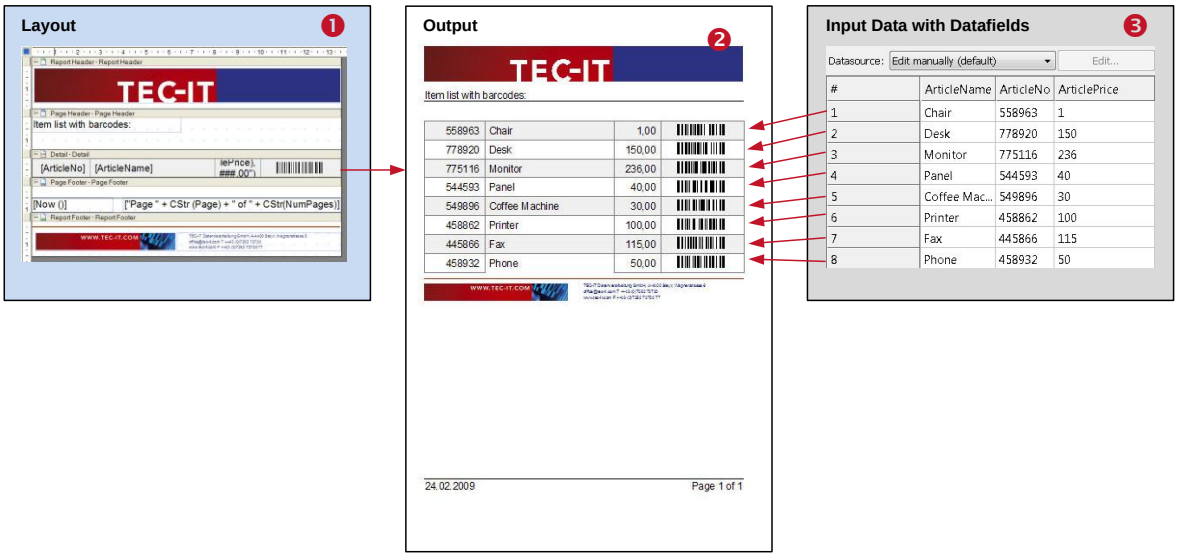


Figure 12: Printing Concept

In contrast to a word processor which is mostly used for static content, a report generator like **TFORMer** generates output based on dynamic data. Good examples for dynamic contents are the article numbers on product labels, the addresses in serial letters and the order positions in invoices.

The dynamic content is usually provided by an external *datasource* (❸). This is either a text file, a database, an arbitrary application or user specified data.

5.2 Rendering the Layout

Each print-out is based on the form layout. The form layout uses different design elements like text elements, barcode elements, shapes and pictures which are placed on bands (see Figure 13, ❶).

When generating the output, each of the bands will be rendered repeatedly: Usually a *detail band* is rendered for each single record (see Figure 13, ❷) of the datasource. *Page header* and *page footer* are rendered at the top respectively bottom of each page automatically (see Figure 13, ②). The

report header and the *report footer* are printed at the beginning and at the end of the report (see Figure 13, ①). In this way the output is assembled.

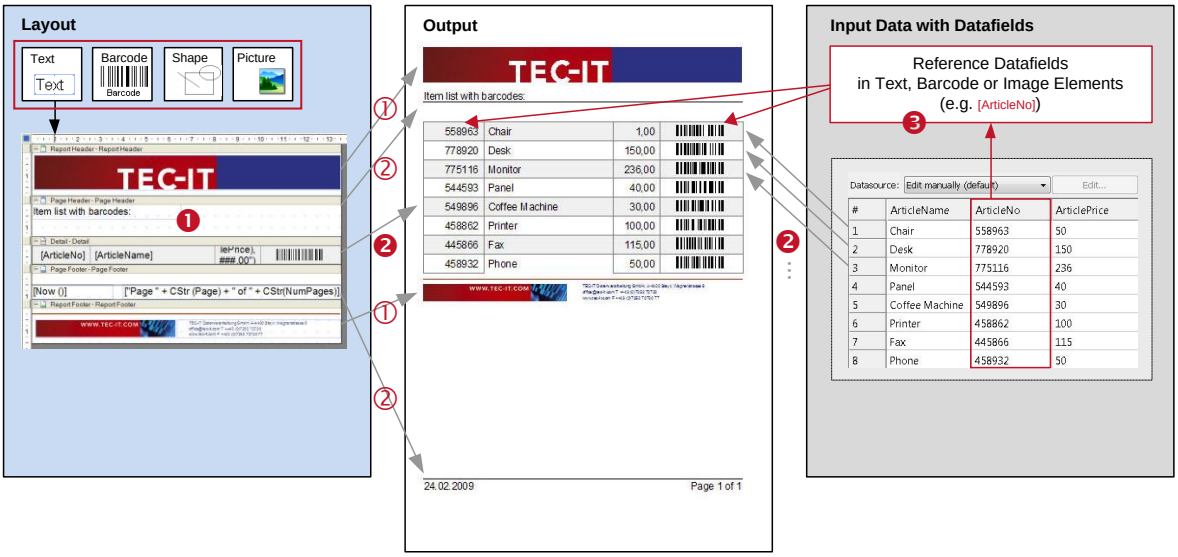


Figure 13: Rendering the Layout

For providing the content you have various possibilities: Text, barcode and image elements can be filled with fixed data. Aside from that they can also reference the fields in a datasource. Such references will be resolved during print time and the actual data is inserted (see Figure 13, ❸). However, not only the fields in the datasource may be used to provide dynamic data. **TFORMer** also provides system fields (e.g., the current page number) and functions (e.g., the current date), and you can perform your own arbitrary computations. Besides that, you can easily create your own serial numbers and use them in the layout.

5.3 Datasource Concept

For generating output **TFORMer** requires data. The records and the datafields control the content and the appearance of the output. As a rule of thumb **TFORMer** prints one detail band per record of the datasource.

Please note, that this data-centric output scheme is maintained in all cases, even when printing pure static documents like personal address labels. In this case the number of records (or the copy counter for records) controls the number of printed labels.

A datasource can be seen as a simple table, which is read row by row (record by record) by **TFORMer** during printing. Each *record* (each row) provides the values for one or more *datafields* used in the layout (e.g., “ArticleName”, “ArticleNo” and “ArticlePrice”).

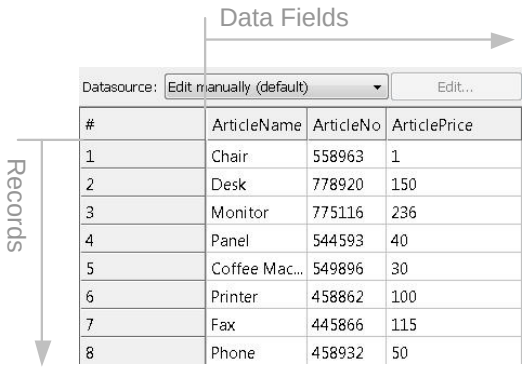


Figure 14: Datasource Concept

5.4 Reading a Datasource and Printing Bands

TFORMer processes an output request in the following way:

A datasource is read record by record in the native order of the datasource. **TFORMer** does not change the order of the records.

After starting a print-job the first data record is loaded. If no record exists **TFORMer** does not produce any output.

Then, for each available data record, **TFORMer** internally iterates through all bands in the layout. Whether a band should be printed on the current printing position on the output page depends on the following rules:

- If the current record is the first record of the input data, all defined *report headers* are printed.
- If the current record is the first record within a group, all defined *group headers* are printed.
- All defined *detail bands* for the current record are printed.
- If the current record is the last record within a group, all defined *group footers* are printed.
- If the current record is the last record of the input data, all defined *report footers* are printed.
- Printing *page headers* and *page footers* is triggered automatically. It works completely independent from the current record.

► Keep in mind that **TFORMer** supports multiple bands of the same type in one layout (e.g., multiple page headers).

► Whether a specific band is printed or not can be controlled via printing conditions.

For more information on bands, please refer to chapter 7.

5.5 Typical Workflow

The typical workflow when designing a new layout is outlined below.

5.5.1 Identify Dynamic Data

The first step before creating a layout is to identify the static and dynamic content of the layout. Dynamic content is usually provided by the user or by external datasources via placeholders. These placeholders are called datafields and are filled with current values during print-time.

5.5.2 Layout Design

To create the layout, you either modify one of the numerous available templates, or you create your own layout.

Layouts are composed of different logical areas called bands. When designing a layout you can combine different bands to create different layouts schemes. Each band can contain different graphical elements (text elements, barcode elements, shapes and images). Dynamic features (e.g., printing conditions) give you additional control over the output.

For more information on designing layouts, please refer to chapters 6 to 12.

5.5.3 Provide Data

Before you can generate any output you have to specify the data to be printed. The data can be provided in different ways:

- It can be edited manually.
- It can be imported from an external file or from a database.
- It can be provided programmatically via the **TFORMer SDK**.

For more information on providing data, please refer to chapter 13.

5.5.4 Generate Output

Once you have designed your layouts you can assign different datasources to generate output with different content. For details, please refer to chapter 15.

6 Forms

6.1 Introduction

TFORMer files are called forms. A form contains the graphical design (the layout), which consists of one or more vertical areas holding all graphical design elements. Furthermore the form contains all datafields, datasources, tray mappings and layers.

6.2 Basic Operations

6.2.1 Open an Existing Form

To open an existing form select *File ► Open…* from the menu (or press the *"Open"* icon in the toolbar). You can also press the keyboard shortcut *Ctrl+O.* Then select the requested file and press *OK* to confirm.

Alternatively you can also double-click on the .tff file in the Windows® Explorer, or drag the file from the Explorer to the **TFORMer** window.

6.2.2 Create a New Form

To create a new form select *File ► New Form…* from the menu (or press the *"New"* icon in the toolbar). You can also press the keyboard shortcut *Ctrl+N.* The following wizard will appear:

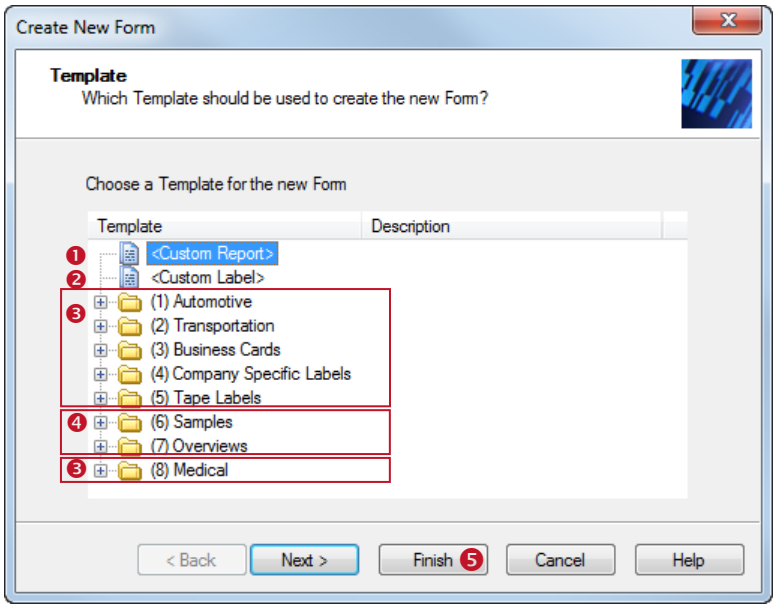


Figure 15: Create a New Layout

Select the template that you want to start with. Choose between:

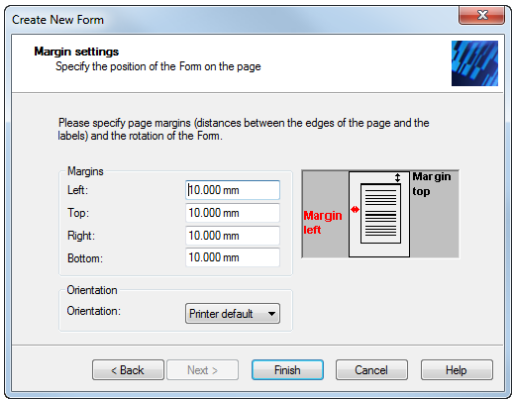
❶ Custom report (see section 6.2.2.1).
❷ Custom label (see section 6.2.2.2).
❸ Ready-to-use label or report templates for industry, logistics and medical branch.
❹ Sample and overview layouts, demonstrating the functionality of **TFORMer.**

First select the desired item. Then click *Finish* (❺) to open the template.

The layout can be adapted to your needs by adding, removing or modifying bands and design elements.

6.2.2.1 Custom Report

If you want to design your own report, choose *"Custom Report"* and then click *Next.* The following wizard will appear:



In this dialog you can specify the page *margins* and the page *orientation.*

Click *Finish* to open the form.

The following layout was created with the settings as shown above. Reports typically have a *report header,* a *page header,* followed by a list of *data records (detail bands),* a *page footer* and a *report footer.* Upon creation of the new report the print data for 10 records is generated automatically. The print preview will show one detail band for each of these dummy records.

Figure 16: Custom Report – Layout View

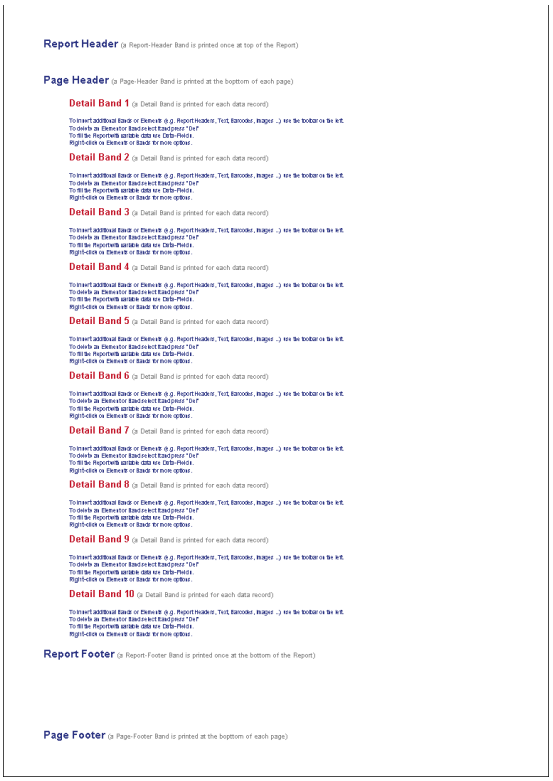
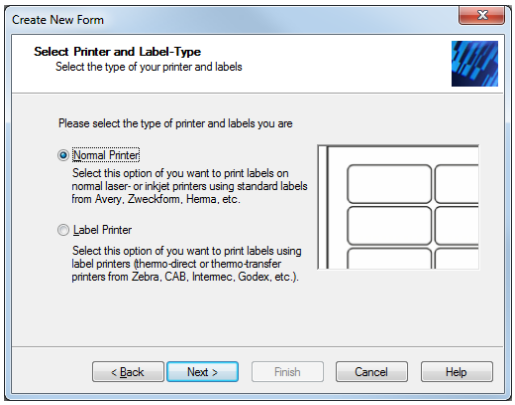


Figure 17: Custom Report – Print Preview (10 Data Records)

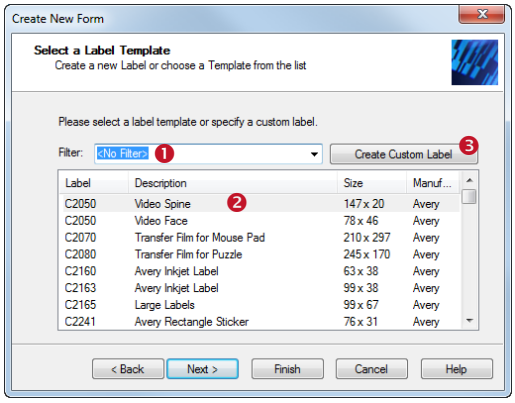
6.2.2.2 Custom Label (Normal Printer)

If you want to design your own label for normal laser or inkjet printers (e.g., Avery® standard label formats), choose *"Custom Label"* and then click *Next.* The following wizard will appear:



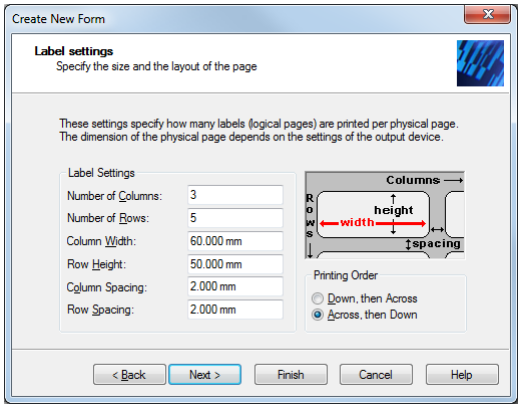
Choose *Normal Printer* (= print multiple labels on one output page).

Click *Next* to continue.



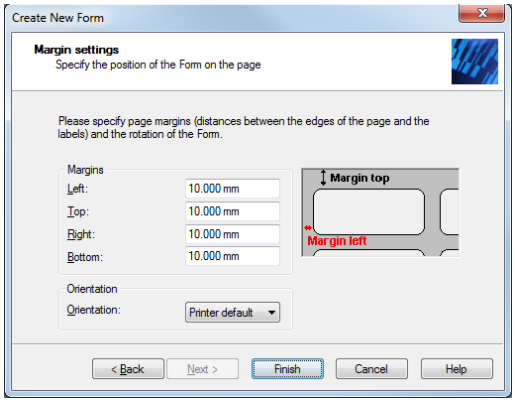
Now select the required label format. Using field ❶ you can filter the template list. To confirm your selection in ❷ click *Finish.*

If the required template is not listed, select *Create Custom Label* ❸.



Use this page to specify the number of *rows* and *columns* (= the number of labels per page), the *label dimensions,* the *row-* and *column-spacing* and the *printing order* of the labels.

Click *Next* to continue.



On this dialog you can specify the page *margins* and the page *orientation.*

Click *Finish* to open the form.

The following layout was created using the settings as shown above (3 columns, 5 rows, label dimensions of 60 mm x 50 mm, row-and column-spacing each 2 mm). Labels typically use the *detail band* only. No page headers or footers are used by default. Upon creation of the new label the print data for 10 records is generated automatically. The print preview will show one label for each of these dummy records.

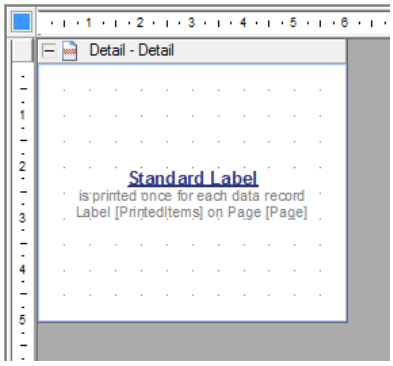


Figure 18: Custom Label – Layout View

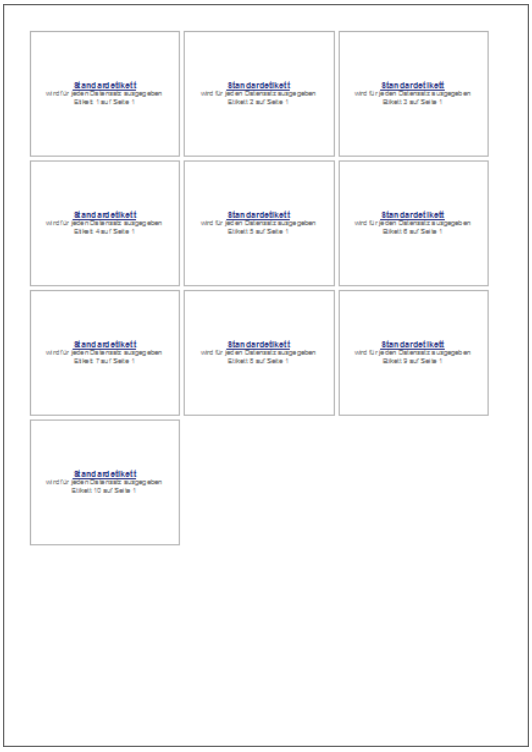
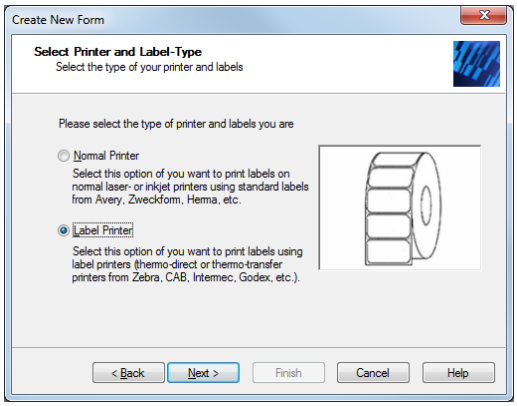


Figure 19: Custom Label – Print Preview (10 Data Records, multiple Labels per Page)

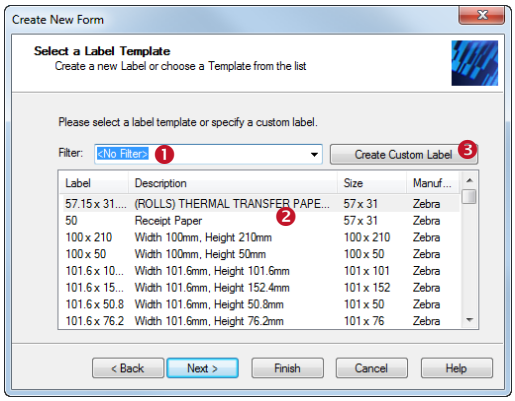
6.2.2.3 Custom Label (Label Printer)

For printing labels with label printers (e.g., thermo transfer printers), choose *"Custom Label"* and then click *Next.* The following wizard will appear:



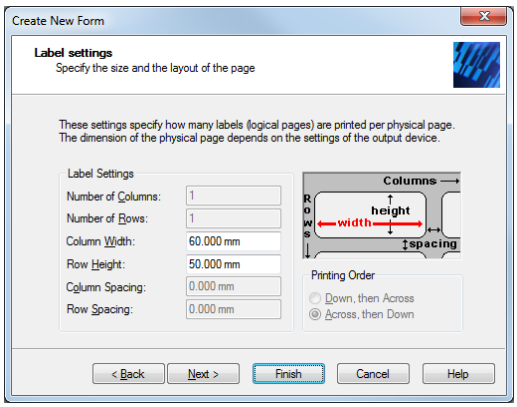
Choose *Label Printer* (= print one label per "page").

Click *Next* to continue.



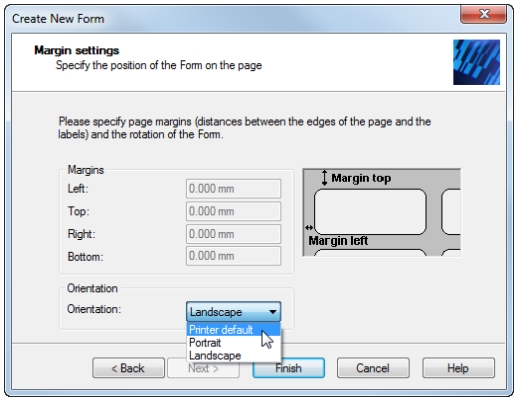
Now select the required label format. Using field ❶ you can filter the template list. To confirm your selection in ❷ click *Finish.*

If the required template is not listed, select *Create Custom Label* ❸.



Use this page to specify the *label dimensions*.

Click *Next* to continue.



On this dialog you can specify the page *orientation.*

Click *Finish* to open the form.

The following layout was created using the settings as shown above (label dimensions of 60 mm x 50 mm). Labels typically use the *detail band* only. No page headers or footers are used by default. Upon creation of the new label the print data for 10 records is generated automatically. The print preview will show one label for each of these dummy records.

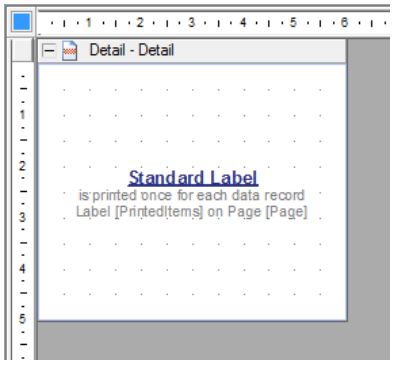


Figure 20: Custom Label – Layout View

Figure 21: Custom Label – Print Preview (10 Data Records, 1 Label per Page)

6.2.3 Page Setup Wizard

The page setup wizard is opened automatically upon creation of a new form (see sections 6.2.2.1 to 6.2.2.3). It is still available for forms, which are already opened in the layout view.

This wizard helps you to adjust the most important output parameters like page margins and page orientation. For labels you can additionally adjust the row- and column- settings and the print order *("Across, then Down", "Down, then Across").*

To open the page setup wizard select *File* ► *Page Setup…* from the menu.

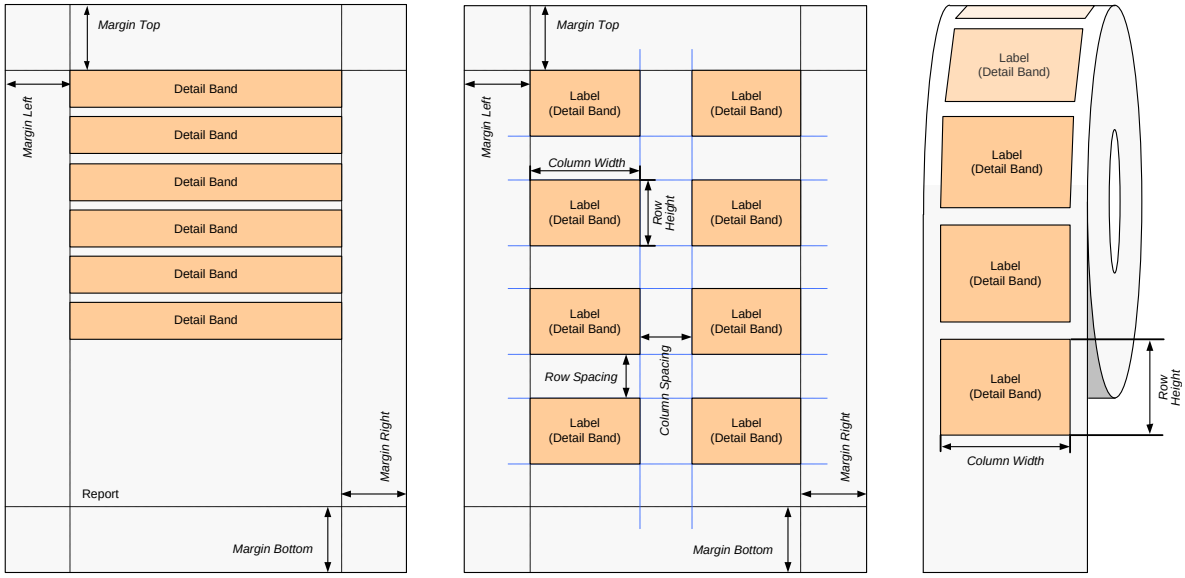


Figure 22: Page Setup (Reports, Standard Labels, and Label Printers)

6.2.4 Save a Form

To save an open layout select *File* ► *Save* or *File* ► *Save as…* from the menu.

File ► *Save* saves the layout with the current file name. Instead of selecting *File* ► *Save* you can also press the *"Save"* icon in the default toolbar or you can press the keyboard shortcut *Ctrl+S.*

Choosing *Save as…* allows you to specify a file name before saving the file.

6.2.5 Switch between Forms

To switch between open forms just click on the corresponding tab in the layout window. Alternatively you can also use the layout picker or the menu *Window.* Using the keyboard shortcut *Ctrl+Tab* you will switch to the next layout tab. *Ctrl+Shift+Tab* switches to the previous tab.

6.2.6 Close a Form

There are different ways to close an open form:

- Click on the small ✕ in the layout window tab (see section 4.8, ❶).
- Select *File* ► *Close Form* from the menu.
- Click with the right mouse button on the form (either in the layout window or in the design tree window) and select *Close Form* from the pop-up menu.

6.3 Form Properties

Using the properties window you can adjust all form related settings. You can

- adjust all properties for page setup (page margins, orientation, rows and columns, …). Most of these properties can be adjusted via the *Page Setup Wizard* as well.
- apply a background image (watermark),
- specify a name for the output or spool file.

First select the form. Thereafter the layout properties can be adjusted.

6.3.1 Form Selection

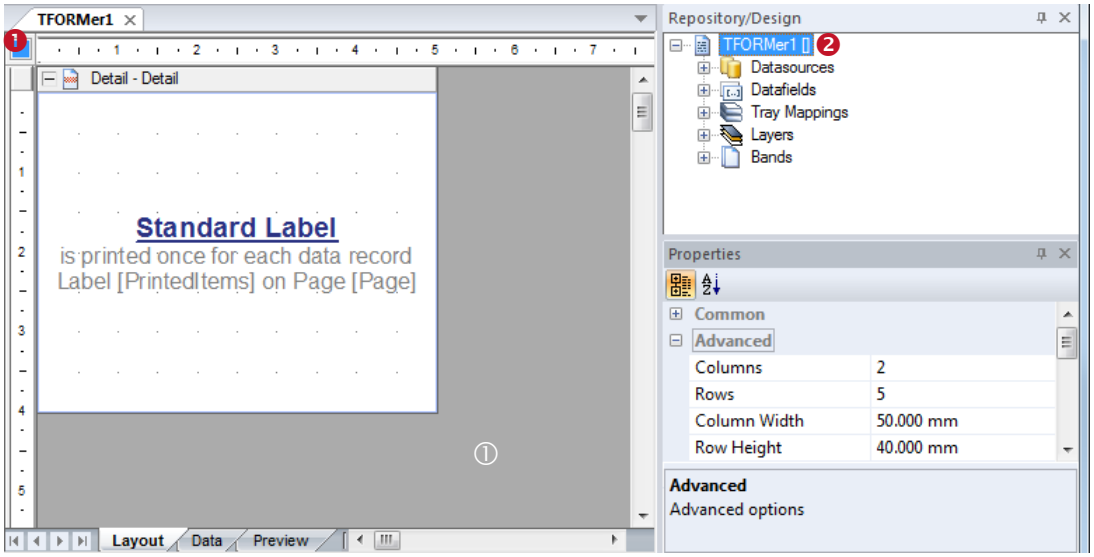


Figure 23: Form Selection

A form can be selected by clicking with the left mouse button in the upper left corner of the layout view ❶ (or anywhere inside the gray area ①). Alternatively you can select the form by clicking on the top level element in the design tree (see ❷). A square (under XP: ▪; under Vista: ■) in ❶ will indicate, that the form is selected.

6.3.2 Page Setup via Form Properties

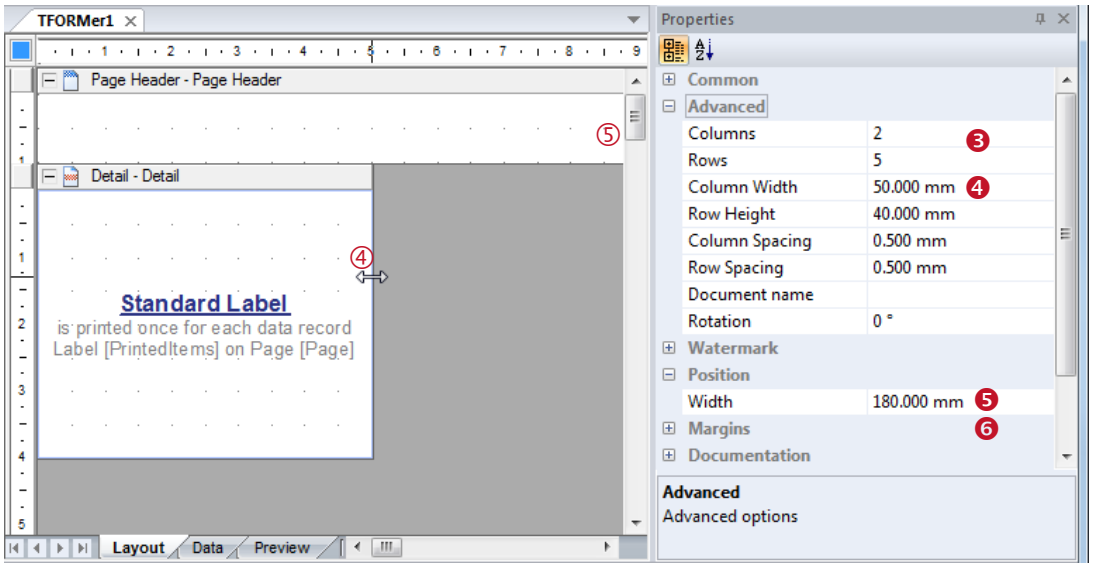


Figure 24: Form Properties

As soon as the form is selected the following properties can be adjusted. For a detailed description on layout properties, please refer to section A.1. The adjustments for specific output schemes are described in Appendix D.

6.3.2.1 Page Margins

The page margins *(Margin Left, Right, Top* and *Bottom)* confine the printing area on the output page. They can be adjusted in the *Margins* group ❻. Page margins are given in respect to the physical borders of the output page.

6.3.2.2 Rows and Columns

These two properties are used to adjust the number of rows and columns for label printing or to switch between label- and report-style output. Rows and columns are adjusted in the *Advanced* group (see ❸).

Whenever both values are equal to 1 **TFORMer** assumes it is printing a report (see section 6.2.2.1) or labels on a label printer (see section 6.2.2.3). All detail bands are printed using the whole page.

Using values greater than one means that the page will be divided into the given number of rows and columns (label printing on normal printers, see section 6.2.2.2).

For detailed information on layout schemes, please refer to Appendix D.

6.3.2.3 Width Specifications

TFORMer distinguishes between two different widths in a layout:

1. *Layout width*
   The layout width specifies the horizontal space, which is available in the layout window (⑤). It defines the width of all bands (see also section 7.3.5). Usually the layout width is set to the width of the output page minus the page margins. The width can either be entered in the *Position* group (see ❺), or it can be changed per mouse. Simply drag the border to the desired position.

2. *Column width*
   When printing labels the width of one label (④) is given via the *column width* property ❹ (see section 6.3.3 below).

6.3.3 Label-Specific Properties (Normal Printer)

6.3.3.1 Row Height and Column Width

In order to print multiple labels on one output page (e.g., multiple business cards on an A4 page) you have to specify the desired number of rows/columns (❸) and/or the exact label size (❹). The following table shows you how to specify the desired page partitioning: You can adjust the entire set of parameters manually, or you can specify just parts of it and let **TFORMer** calculate the rest.

Row Height / Column Width (= label dimensions)	**Rows / Columns** (= labels per page)	**Description**
Specify the row height and the column width manually	Specify the number of rows and columns manually	The page partitioning is specified completely. **TFORMer** does not need to do any additional calculation.
	"auto"	**TFORMer** will compute the number of rows and columns that fit on the output page. Such a layout may be printed on different paper sizes (e.g., A4, A5). **TFORMer** takes care of the correct number of rows and columns automatically.

“auto”	Specify the number of rows and columns manually	TFORMer will compute the label width and height automatically with respect to the size of the output page. Thus the label size may vary!
	“auto”	This combination is not recommended.

The column width ④ (= the label width) applies to all bands which are printed on the label (e.g., detail bands). Though, additional headers and footers which are not printed per label but per page may also use the whole layout width ⑤.

6.3.3.2 Row and Column Spacing

By adjusting the *Row-* and *Column-Spacing* you specify the horizontal and vertical distances that are left empty between the single labels on one output page. The spacing can be adjusted in the *Advanced* group.

6.3.3.3 Printing Order

The *Printing Order (“Across, then Down”* or *“Down, then Across”)* specifies the order in which small labels are printed on one output page. The *Printing Order* property can be adjusted in the *Common* group.

6.3.4 Watermark

The *Watermark* is an image which is used as background for each page. The image file name (or an expression which computes the file name during print-time), the output size and the output position can be adjusted in the *Watermark* group.

6.3.5 Document Name

The *Document Name* is used as name for the generated output file or spool file. If left blank, the file name of the form is used. The document name can be adjusted in the *Advanced* group.

7 Bands

7.1 Introduction

A layout consists of one or more bands. Each band is a vertical section, which contains graphical elements like text elements, lines or barcodes. Each band fulfills a specific purpose and is printed in a clearly defined position and sequence.

Besides that, bands may be used for computations, and they can be used to control the output behavior (see chapter 12).

7.2 Band Types

TFORMer supports the following band types:

- *Report Header*
  This band is printed at the top of the first page of the output. Alternatively it can also be printed at the top of the first label. This band is optional.
- *Page Header*
  This band is printed at the top of each page (on the first page after the report header). Alternatively it can also be printed at the top of each label. This band is optional.
- *Group Header*
  This optional band is printed before a group starts. A group is defined by the *"Group-By"* property in this band. Whenever the *"Group-By"* expression changes, a new group is started.
- *Detail Band*
  This band is printed for each single record.
- *Group Footer*
  This optional band is printed after a group. A group is specified by a *"Group-By"* expression.
- *Page Footer*
  This band is printed at the bottom of each page (on the last page before the report header). Alternatively it can also be printed at the bottom of each label. This band is optional.
- *Report Footer*
  This band is printed on the last page of the report (after the last data record). Alternatively it can also be printed at the bottom of the last label. This band is optional.

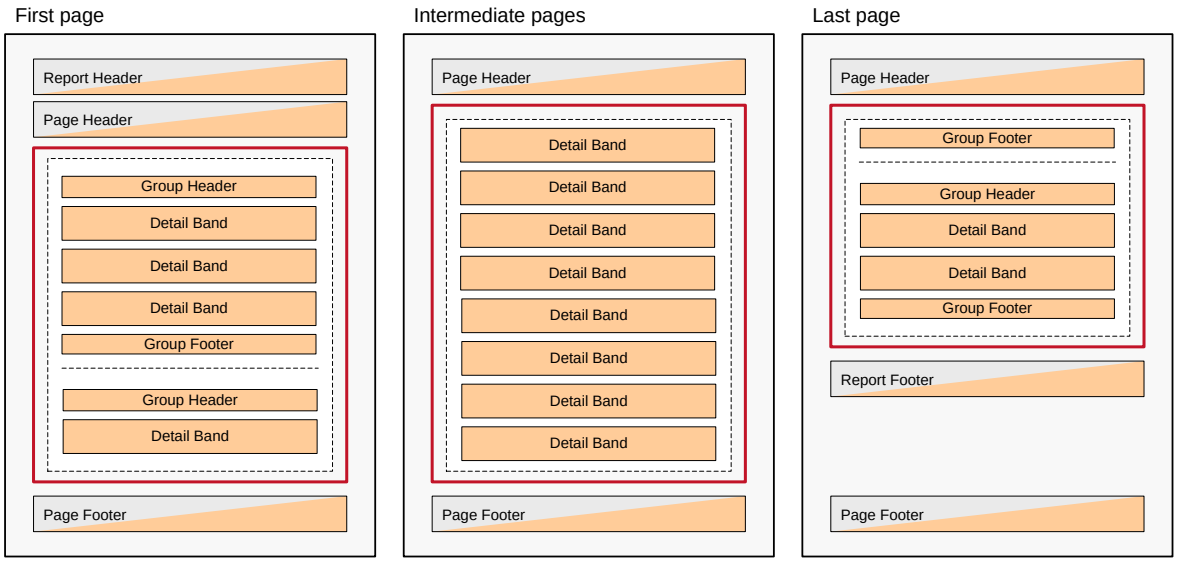


Figure 25: Rendering Scheme for Bands (Report)

The figure above shows the printing of a report. When printing labels, typically no page- and report headers/footers are required. The detail bands will be arranged differently (see also section Appendix D).

7.3 Basic Operations

7.3.1 Insert a Band

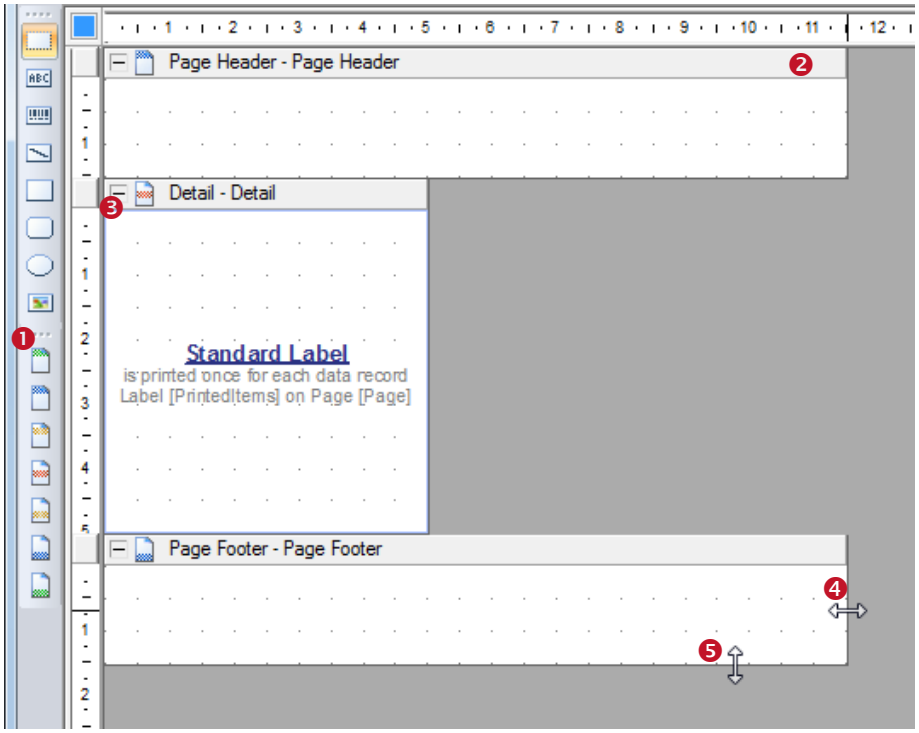


Figure 26: Layout

To insert a band click on the appropriate icon in the toolbar ❶ or select *Insert* ▶*Band* from the menu. The selected band will be inserted as a vertical section on the worksheet.

	New Report Header	Add a new report header to the layout.
	New Page Header	Add a new page header to the layout.
	New Group Header	Add a new group header to the layout. For each group header you have to define a group-by expression.
	New Detail Band	Add a new detail band to the layout.
	New Group Footer	Add a new group footer to the layout. For each group footer you have to define a group-by expression.
	New Page Footer	Add a new page footer to the layout.
	New Report Footer	Add a new report footer to the layout.

7.3.2 Band Selection

To select a band, click on the band header ❷ or on an empty area inside the band. You can also select a band in the design tree. Once a band is selected, it can be copied, deleted or its properties can be adjusted.

7.3.3 Expanded/Collapsed View of Bands

Bands can be displayed in expanded or in collapsed view. The collapsed view is used to hide a band during the layout design. The resulting output is NOT influenced!
Click on the ⊞ symbol to expand, and on the ⊟ symbol to collapse the band view (see ❸).

7.3.4 Adjust Band Properties

To adjust the band properties, first select the band. Then the properties can be edited in the properties window. For a list of available properties, please refer to Appendix A.2.

7.3.5 Change Size

The size of a band can be changed by clicking and dragging its confining borders at the right and at the bottom of the band. Alternatively enter the dimensions in the *Position* group in the properties window (see Figure 26, ❹ and ❺).

Please note: The width property will be applied to all bands in the layout (= layout width, see section 6.3.2.3).

7.3.6 Change Band Printing Order

If your layout contains more than one band of the same type (e.g., three *detail bands),* **TFORMer** processes them from top to bottom.

This printing order of bands can be changed: Select the band that you want to move and choose *Layout* ► *Band Order* from the menu. You can also right-click on the band and then select *Band Order* from the pop-up menu. Alternatively use the keyboard shortcuts *Alt+*↑ and *Alt+*↓.

7.3.7 Delete a Band

Select the band that you want to delete, and then press the *Del* key. Alternatively click on the ✕ icon in the *"Default"* toolbar or press the right mouse button and select *Delete* from the pop-up menu.

7.3.8 Printing Conditions

For each band you can specify a printing condition. A printing condition decides at print-time whether a band is printed or not.

For more details on printing conditions, please refer to section 12.2.

7.3.9 Pre- and Post-Evaluations

For each band you can specify pre-evaluation and post-evaluation expressions. These expressions will be evaluated before and after the band is printed.

For more details on pre- and post-evaluations, please refer to section 12.4.

8 Elements

8.1 Introduction

Elements (or graphical design elements) are all sorts of printable shapes like texts, barcodes, lines, rectangles, circles and pictures. They are placed on the bands to create the layout.

8.2 Basic Operations

8.2.1 Insert an Element

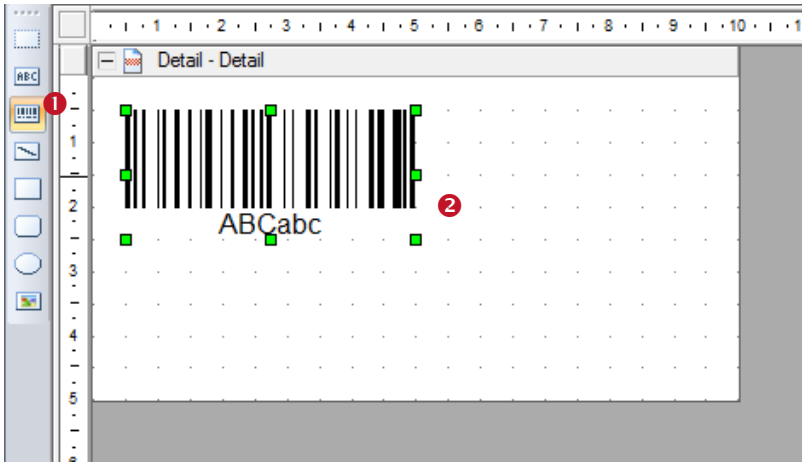


Figure 27: Layout View

To insert an element (e.g., a barcode) into your layout, first click on the appropriate symbol in the *toolbar* (❶). Then place the element on one of the bands inside the layout (❷). You can place elements in two different ways:

- A single mouse click on the worksheet will insert the element with its default size.
- If you click and drag a frame with the mouse pointer (while holding down the mouse button), the element is inserted in the given dimensions upon release of the mouse button.

	Select Elements	Select existing elements (or bands) in the layout view. This tool is enabled by default.
	New Text	Create a new text element. A text element provides formatted text. The content may consist of static and dynamic data (see also section 8.3.1).
	New Barcode	Create a new barcode. A barcode encodes the given data in a machine readable form. The content may consist of static and dynamic data (see also section 8.3.2).
	New Line	Create a new line.
	New Rectangle	Create a new rectangle.
	New Rounded-Rectangle	Create a new rectangle with rounded corners.
	New Ellipse	Create a new ellipse.
	New Picture	Embed a picture. A picture element renders the specified image file. The name of the image file may be specified as fixed text or computed (see also section 8.3.3).

For a list of element properties, please refer to Appendix A.3.

8.2.2 Element Selection

In the layout view you can select elements in various ways:

- Click with the left mouse button on an element.
  Hold down the *Shift* key or the *Ctrl* key for multiple selection.
- Click with the left mouse button in an empty area of the layout view and (while holding the mouse button pressed) drag a selection frame around the required elements. Then release the mouse button. Depending on the direction, in which you have dragged the selection frame, different selection criteria are used:
  - Drag frame *from left to right:*
    Only objects that are entirely overlapped by the frame will be selected.
  - Drag frame *from right to left:*
    All objects that are entirely or partially overlapped by the frame will be selected.

A thin border with small rectangles (= sizing handles) on its corners and on its sides indicates that the element is selected.

► To start a selection frame over an existing graphical element try to hold down the *Shift* or the *Ctrl* key. This will prevent other elements (which might be under the mouse pointer) from being moved.

8.2.2.1 Examples

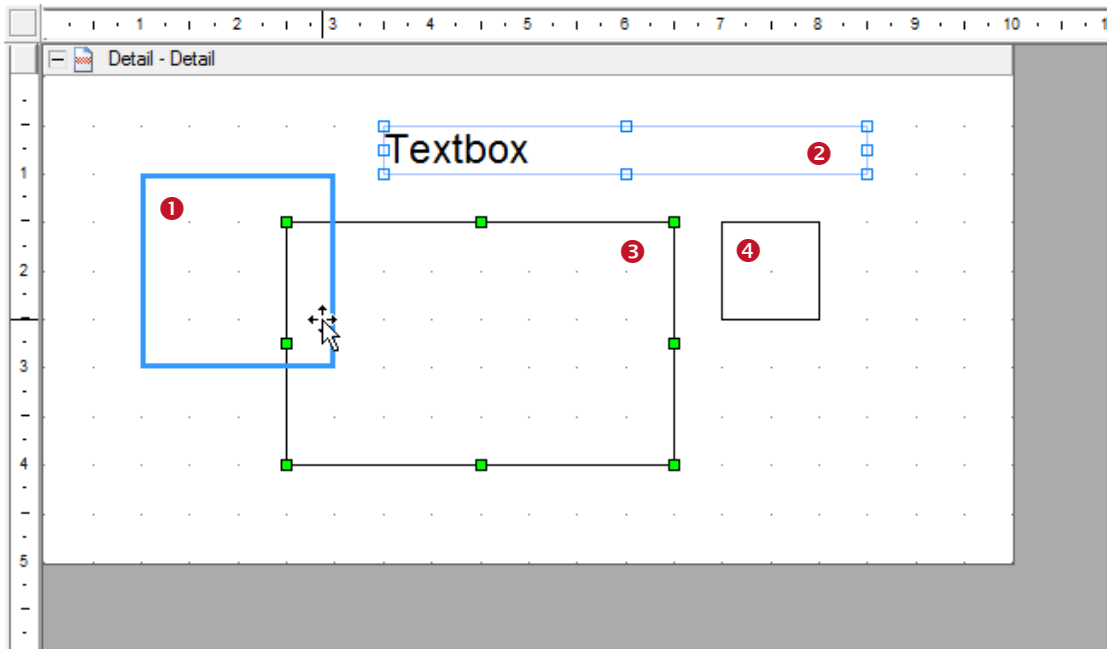


Figure 28: Element Selection

❶ This element has the mouse focus (bold outline). If you click your left mouse button, it will be selected.

❷ This element is part of a selection. Because it does not have the selection focus its sizing handles are drawn as outline (▫).

❸ This element is part of a selection and has the selection focus. This is indicated by solid sizing handles (■).
If you adjust the size or the alignment of two or more elements (see sections 8.2.10.1 and 8.2.10.2), the focused element will act as “master”. This means its properties (or property changes) will be applied to the rest of the selection. The last element that is added to the selection always receives the selection focus.
You can set the focus within a selection by clicking on the required element. Do NOT hold down any modifier keys, like *Shift* or *Ctrl!*

❹ This element is not selected.

8.2.3 Change Element Size

After selecting an element, you can change its size by clicking and dragging the sizing handles. If you do not want the mouse to snap to the grid (see section 17.2.3), press the *Alt* key while dragging.
Another possibility to change the size of an element is to edit the width and the height in the properties window.

> ► Please note: If the selection contains more than one element, all selected elements will be resized simultaneously.

8.2.4 Move an Element

An element can be moved by clicking and dragging. While dragging you can use the following modifier keys:

Modifier Key	Description
Shift	Snap to axis. The movement of the element will be restricted to the horizontal and vertical axis.
Ctrl	Copy mode. A copy of the original element(s) will be created on the release position. In copy mode the mouse pointer changes its shape to  .
Alt	Fine positioning. Elements can be positioned off the grid, without having to turn the “snap to grid” option off.

8.2.5 Element Properties

To change the properties of a selected object, you can either:

- Edit the desired properties in the properties window.
- Use the format toolbar (see section 4.3) to change frequently used settings like font style, background color, etc.

8.2.6 Positioning and Measuring

The rulers and the status bar assist you with measuring and positioning of inserted elements.

When entering measurement values manually **TFORMer** interprets such values as specified in mm or in inch (depending on your system settings). For your convenience you may also append common abbreviations for measurement units (mm, cm, inch, mil) to the numeric input.

8.2.7 Delete an Element

You can delete selected elements in various ways:

- Press the *Del* key on the keyboard.
- Right-click on the element. Then select *Delete* from the pop-up menu.
- Select *Edit* ► *Delete* from the menu.
- Click the delete symbol ✕ in the toolbar.

8.2.8 Printing Conditions

For each element you can specify a printing condition. A printing condition decides at print-time whether the element will be printed or not. For more details, please refer to section 12.2.

8.2.9 Assign Elements to a Printing Layer

A printing layer gives you the possibility to use one single printing condition for multiple elements. It lets you control whether all elements assigned to this layer are printed or not.

By default, only one layer (the *"Base"* layer) is available. Newly created elements are assigned to the *"Base"* layer by default.

The printing layer does NOT affect the z-order, in which the elements are printed!

For more details on printing layers, please refer to section 12.3.

8.2.10 Design Functions for Elements

These functions simplify the creation of layouts. You can make all selected elements the same size, you can adjust the spacing between selected elements, and more.

The following properties can be adjusted:

- Alignment
- Size
- Z-Order
- Spacing (horizontal and vertical)

The design functions are available through the toolbar (see Figure 29) or through the *Layout* menu.

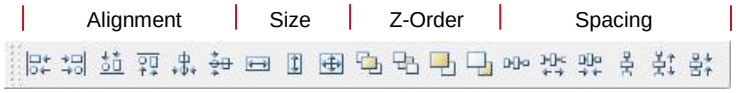


Figure 29: Layout Toolbar

> ► When changing the alignment or the size of selected elements, the element with the selection focus (see 8.2.2.1, ❸) acts as master: all selected elements will be changed according to its values!

8.2.10.1 Alignment

Select at least two elements. Then choose one of the following functions. The element with the selection focus acts as master.

	Align Left	The selected elements will be aligned left. Shortcut: *Ctrl+*←
	Align Right	The selected elements will be aligned right. Shortcut: *Ctrl+*→
	Align Bottom	The selected elements will be aligned at the bottom. Shortcut: *Ctrl+*↓
	Align Top	The selected elements will be aligned at the top. Shortcut: *Ctrl+*↑
	Align Vertically	The selected elements are center-aligned vertically.
	Align Horizontally	The selected elements are center-aligned horizontally.

8.2.10.2 Size

Select at least two elements. Then choose one of the following functions. The element with the selection focus acts as master.

	Make same Width	The same width is applied to all selected elements.
	Make same Height	The same height is applied to all selected elements.
	Make same Size	The same size is applied to all selected elements.

8.2.10.3 Z-Order

Select at least one element. Then choose one of the following functions.

	Move to Top	The selected elements are moved in front of all other elements.
	Move to Bottom	The selected elements are moved behind all other objects.
	Move Up	The selected elements are moved up one level.
	Move Down	The selected elements are moved down one level.

8.2.10.4 Spacing

Select at least two elements. Then choose one of the following functions. The element with the selection focus acts as master.

	Make Horizontal Spacing Equal	Ensure equal horizontal spacing between the selected elements.
	Increase Horizontal Spacing	Increase the horizontal spacing between the selected elements.
	Decrease Horizontal Spacing	Decrease the horizontal spacing between the selected elements.
	Make Vertical Spacing Equal	Ensure equal vertical spacing between the selected elements
	Increase Vertical Spacing	Increase the vertical spacing between the selected elements.
	Decrease Vertical Spacing	Decrease the vertical spacing between the selected elements.

8.3 Elements with Content

8.3.1 Text Elements

Text elements are used for printing formatted text. Single phrases, words and characters can be formatted individually. Datafields and expressions can be embedded within the text.

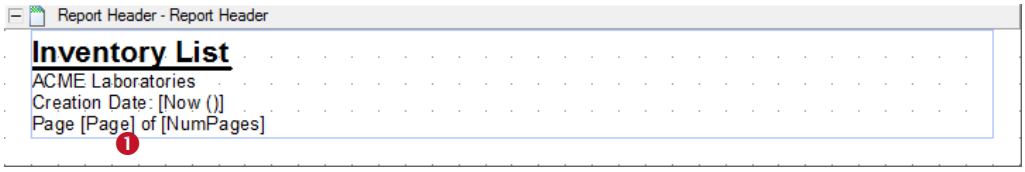


Figure 30: Text Element

In the example above static and dynamic content is mixed. The embedded expressions for the current date *[Now ()],* the current page number *[Page]* and the total number of pages *[NumPages]* are placed in-line with the static text. During print-time these expressions will be replaced by the actual values automatically. Such dynamically provided content is always displayed between square brackets “[]” (see ❶) or angle brackets “<>”:

- Square brackets indicate that the content of the datafield respectively the result of the expression will be inserted as is.
- Angle brackets indicate that the content of the datafield respectively the result of the expression will be interpreted as HTML. This allows you to specify dynamic content with additional formatting (see also section 9.3.3)!

For information on how to edit the content of text fields, please refer to chapter 9.

8.3.2 Barcode Elements

Barcode elements encode the given data in a machine readable form.

Figure 31: Barcode Elements

The barcode data can contain static content (e.g., the fixed string “AaBb12”, see ❶), dynamic content (e.g., the datafield, see ❷), or a combination of both. Dynamically provided content is always displayed between square brackets “[]”.

For information on how to edit the barcode data, please refer to chapter 9.

> ► Please note: The barcode symbol in the layout view is just a sample. It does not encode the actual data! The correct barcode is computed during print-time.

The *Barcode Type* and other barcode specific settings are adjusted in the properties window. You can choose from 1D, 2D and composite barcode symbologies. Depending on the selected barcode type different property groups are available

For more information on available barcode types and their possible adjustments, please refer to the “Barcode Reference” which is available for download on www.tec-it.com.

8.3.3 Picture Elements

Picture elements are used to embed images in a layout. Supported image formats are BMP, GIF, JPG, PCX, PNG, TGA and TIF.

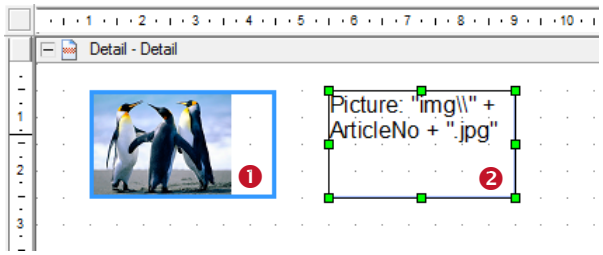


Figure 32: Picture Elements

The actual image is usually specified through a fixed file name (see ❶). If you want to use different images for different records (dynamic logos) you can also compute the file name dynamically (e.g., based on a datafield value, see ❷).

For information on how to edit the image file name, please refer to chapter 9.

> ► You can use absolute file names (e.g., “C:\sample.jpg”) or relative file names (e.g., “sample.jpg” or “img\sample.jpg”). Relative file names are treated relative to the base path of the layout file.

9 Element Content

9.1 General

All text, barcode and picture elements are filled with specific content. This content can be edited using the following dialog.

When inserting a new element or when double-clicking an existing element the editing dialog will be opened automatically. Alternatively select the element and press *F2*, or right click on the element and select *Element-Content…* from the pop-up menu.

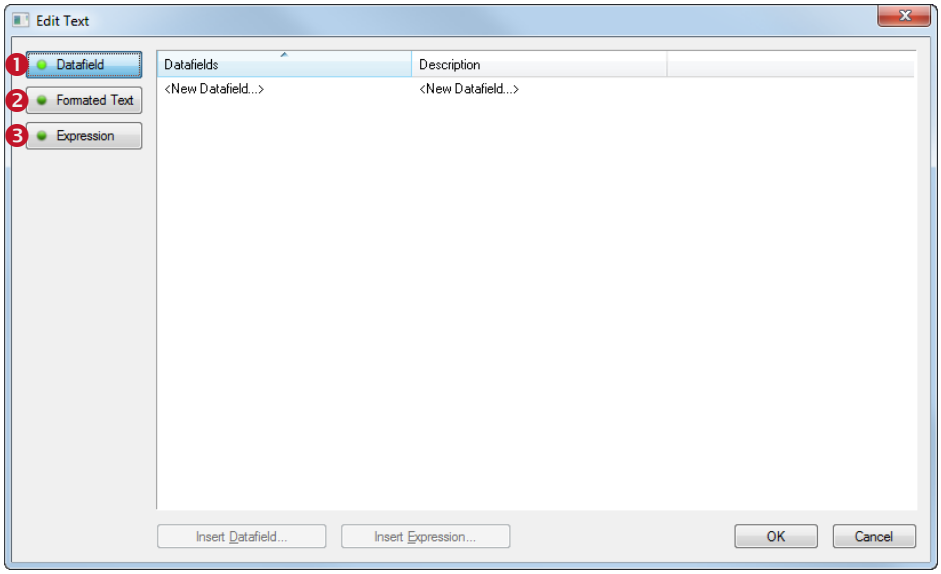


Figure 33: Edit Dialog for Element Contents

First choose the type of content that you want to assign. The following options are available:

- ❶ Datafield (see 9.2)
- ❷ Formatted Text (for text elements, see 9.3)
  Simple Text (for barcode elements, see 0)
  File (for picture elements, see 9.5)
- ❸ Expression (see 9.6)

If you want to assign a datafield (dynamic data) with no additional text or whatsoever your choice will be option ❶. This option is typically used for output as barcode or as simple text element.

If you want to assign a fixed text or if you want to combine the content of a datafield with additional text, other datafields and/or expressions, your choice will usually be option ❷. This option is the standard choice for text elements and barcodes. It can be used for the creation of simple and complex contents.

Option ❸ will be your favorite choice only, if you want to compute the content of an element, based on the provided functions, constants and datafields. An example would be use of dynamic picture file names.

Depending on your selection (❶, ❷ or ❸) one of the following dialogs is shown:

9.2 Datafield

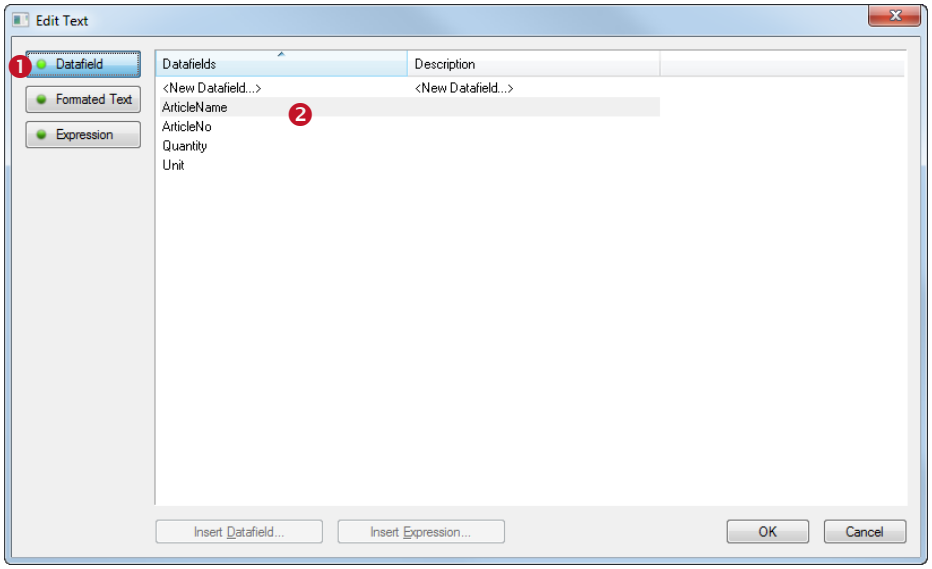


Figure 34: Edit Dialog – Datafield

If the content of the selected element is supposed to be a datafield, select the *Datafield* option ❶. Then select the desired datafield in list ❷ and confirm with *OK.*

The creation of a new datafield is also possible. When double-clicking on “<New Datafield…>” an appropriate dialog will be opened (see section 10.2.1.1).

9.3 Formatted Text (Text Elements)

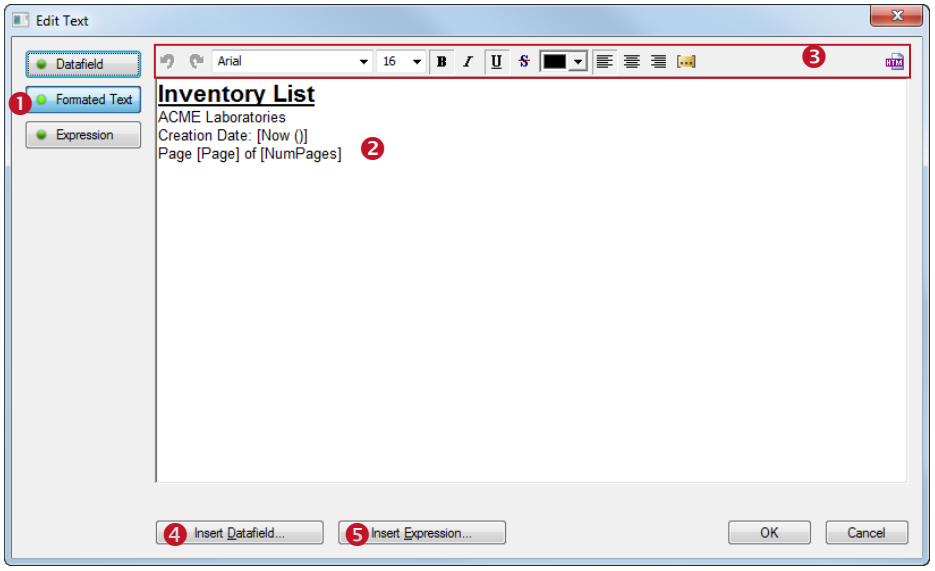


Figure 35: Edit Dialog – Formatted Text

If the content of the selected text element is supposed to be a formatted text, select the *Formatted Text* option ❶. You can now edit the text in ❷. The toolbar ❸ may be used to apply different formatting (single phrases, words and characters can be formatted individually – see section 9.3.1). Using the buttons ❹ and ❺ you can insert datafields and expressions (see section 9.3.2). When you are done with editing confirm with *OK.*

9.3.1 The Toolbar

Toolbar ❸ provides the following functions:

	Undo	Revert the last change. Shortcut: *Ctrl+Z*
	Redo	Revert the last *Undo* Shortcut: *Ctrl+Y*
Arial	Font Name	Select the typeface.
10	Font Size	Select the font size (in points).
B	Bold	Toggle bold text *on/off*. Shortcut: *Ctrl+B*
I	Italic	Toggle italic text *on/off*. Shortcut: *Ctrl+I*
U	Underline	Toggle text underline *on/off*. Shortcut: *Ctrl+U*
S	Strike Through	Toggle text strike through *on/off*.
	Font Color	Select the font color.
	Align Left	Align the text left. This alignment is applied to the whole text element and *not* for single lines or paragraphs. (Same as the *Text Alignment* property *"Top, Left"*.)
	Center	Center the text. This property is applied to the whole text element and *not* for single lines or paragraphs. (Same as the *Text Alignment* property *"Top, Center"*.)
	Align Right	Align the text right. This alignment is applied to the whole text element and *not* for single lines or paragraphs. (Same as the *Text Alignment* property *"Top, Right"*.)
	Highlight Expressions	Highlight all embedded datafields and expressions.
	HTML Source	**TFORMer** internally uses a subset of HTML (see Appendix B) for storing formatted text. With this button you can toggle between the WYSIWIG view and the HTML source view.

9.3.2 Datafields and Expressions

In order to provide dynamic content you have the possibility to insert datafields and expressions in line with the formatted text. Use the following buttons for inserting. For more information on datafields and expressions see chapters 10 and 11.

Insert Datafield...	Insert Datafield	Opens the dialog *Select Datafield* (analogous to section 9.2). In this dialog select the desired datafield from the list and confirm with *OK*.
Insert Expression...	Insert Expression	Opens the *Expression Builder* (see sections 9.6 and 11.2). In this dialog compose your desired expression using the available functions, constants and datafields. Then confirm with *OK*.

The datafield respectively expression will be inserted at the current cursor position. Datafields and expressions are always marked with square brackets "[]" (or angle brackets "<>" in case of HTML content – see section 9.3.3):

Please note:

- ► Datafields and expressions must always be inserted using the appropriate buttons!
- ► If you try to enter the expression start- and end-markers [] (or <>) via keyboard, the text between will not be evaluated as an expression!

To format an in-line expression (font size, color, etc.) – or rather the text which is returned as result – use the format toolbar ❸ as you would do for the rest of the formatted text. For dynamic formatting you can also use HTML formatted expressions (see below).

To edit an existing datafield or expression place the cursor between the square brackets and click on the *Insert Datafield…* or *Insert Expression…* button. The appropriate dialog will be opened. Alternatively you can also double-click on the datafield (or expression) which will always open the expression builder.

Hint: The functions *Insert Datafield…* and *Insert Expression…* are also available via context menu when right clicking inside the text area.

9.3.3 HTML Formatted Expressions

In order to provide dynamic text formatting you can use HTML formatted expressions.

> This feature allows you to control the appearance of a text element during print-time.

If the HTM button is pushed inside the expression builder (see Figure 46, ❻), the result of the expression will be interpreted as HTML content. This means, all supported HTML tags that are found in the result of the expression are directly applied. For a list of supported HTML tags, please refer to Appendix B.

In this example the datafields “ColorOn” and “ColorOff” are utilized for dynamic text formatting.

#	Copies	ColorOff	ColorOn
1	1		
2	1		
3	8		
*			

In order to apply different font colors the appropriate HTML tags (and) are inserted in the Data window. In this case this is done for the first two records. The following eight records are printed with default formatting (as applied in the Edit Text dialog).

Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World

When printing, **TFORMer** reads the content of the fields in the datasource. The contained font tags are directly applied to the output.

9.4 Simple Text (Barcode Elements)

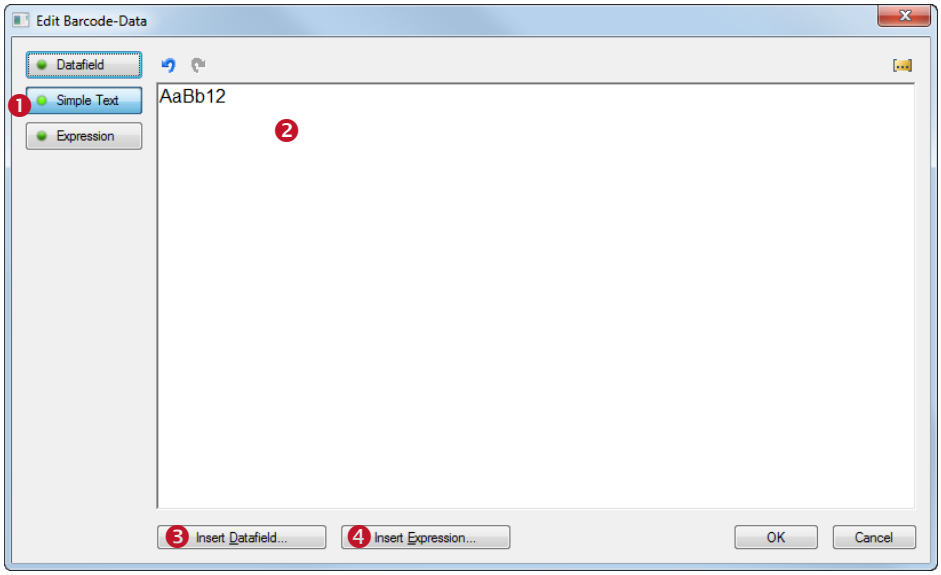


Figure 36: Edit Dialog – SimpleText

If the content of the selected barcode element is supposed to be a simple text, select the *Simple Text* option ❶. You can now edit the text in ❷. Using the buttons ❸ and ❹ you can insert datafields and expressions (analogous to section 9.3.2). When you are done with editing confirm with *OK.*

The button turns expression highlighting on or off.

9.5 File (Picture Elements)

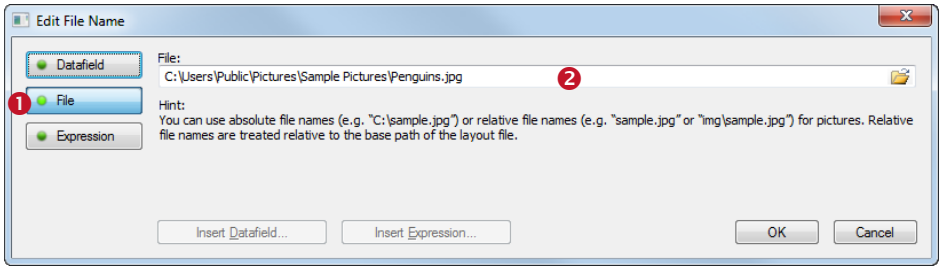


Figure 37: Edit Dialog – File

If the selected picture element is supposed to show a statically defined image (no different images for different records), select the *File* option ❶. Then specify the file name in ❷ and confirm with *OK.* The button opens a file selection dialog.

For dynamically assigned image files use either the option *Datafield* or *Expression*.

9.6 Expression

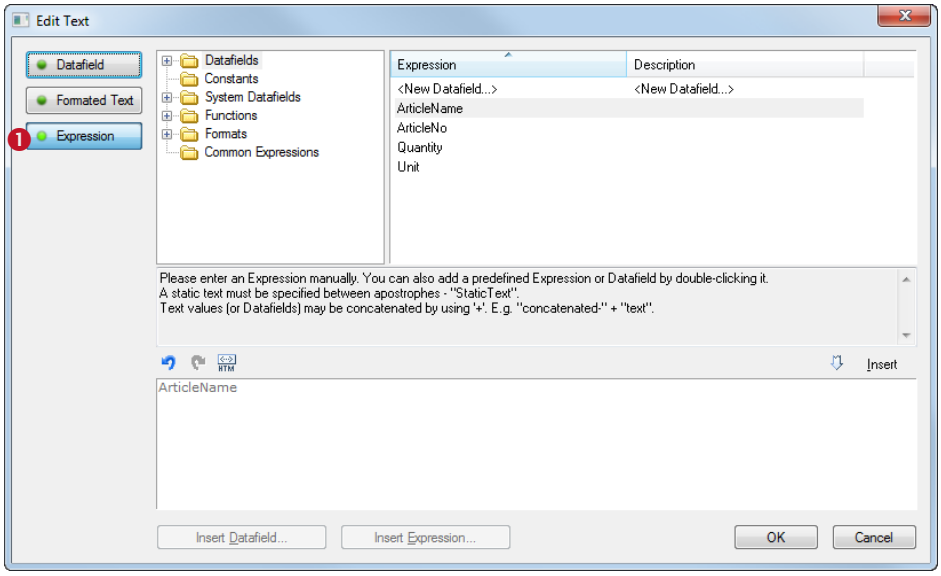


Figure 38: Edit Text Dialog – Expression

If the content of the selected element is supposed to be the result of an expression, select the *Expression* option ❶.

However, the use of pure expressions is only required in special cases (e.g., calculating the file names for dynamic picture content). For all common applications it is recommended to use the option *Datafield* or *Formatted/Simple Text*. Expressions can be embedded in-line with such text content.

For more information on expressions see chapter 11.

10 Datafields

10.1 Introduction

For providing dynamic content **TFORMer** uses so called *datafields.* These datafields serve as placeholders for the actual data. They can be used in text, barcode and picture elements (dynamic logos).

A datafield must be defined before it can be used. This definition takes place directly in the layout (see below) or in a repository (see chapter 16). One layout can use an arbitrary number of datafields (0..n).

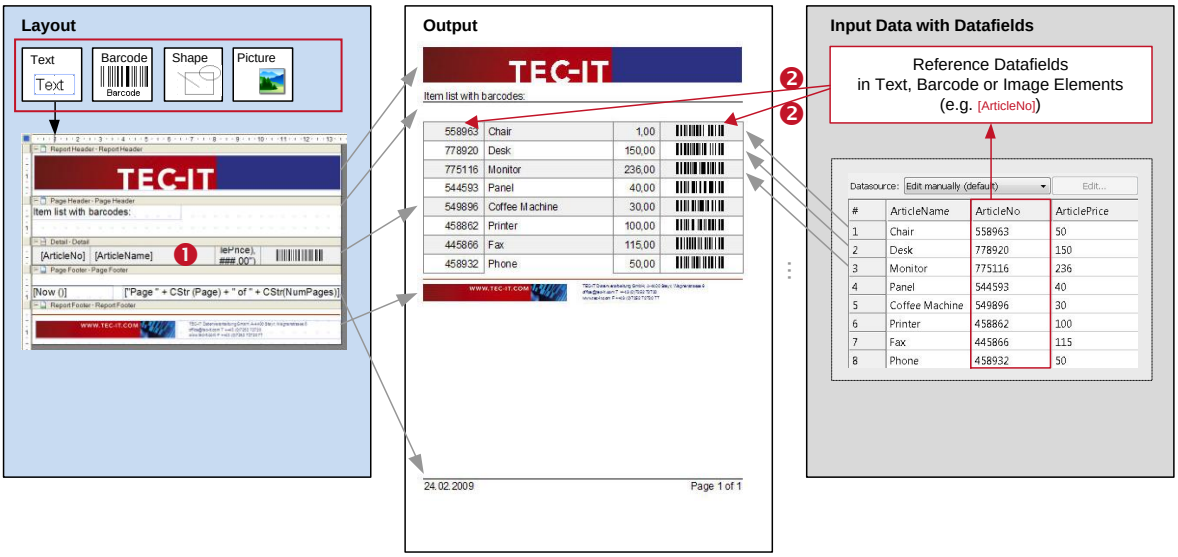


Figure 39: Datafields as Part of the Printing Concept

In the layout **TFORMer** displays each datafield in square brackets "[]" (or angle brackets "<>" for HTML content) – see ❶. The current value of a datafield is available on the print-out only (❷). The value of a datafield is either

- provided manually (see section 13.3.1),
- provided by an external datasource (e.g., imported from a database – see section 13.3.2),
- a serial number (see section 10.3.3),
- computed per record (see section 10.3.2),
- computed for specific bands (pre- and post-evaluation – see section 12.4) or
- provided programmatically by a software developer (see section 15.3).

If no value was loaded (or computed), the datafield returns its default value (see section 10.3.1), its start-value (see section 10.3.3) or zero (see section 10.3.2).

► In order to provide the data from an external datasource, the datafield has to be bound to the required field in the datasource (to the so called sourcefield). Please note: Datafield and sourcefield need not necessarily have the same name (see section 13.4).

10.2 Basic Operations

10.2.1 Create Datafield Definitions

Datafield definitions can be created manually, or they can be imported from a datasource.

10.2.1.1 Create Datafield Definitions Manually

To create a datafield definition, the dialog below is used. You can open this dialog using one of the following methods:

- Right-click in the gray area of the layout or on the *"Datafields"* entry in the design tree and select *New Datafield…* from the pop-up menu.
- Click on the *"New Datafield"* icon in the toolbar.
- Select *Insert* ► *Data* ► *Datafield…* from the menu.
- Select *Data* ►*New Datafield…* from the menu.

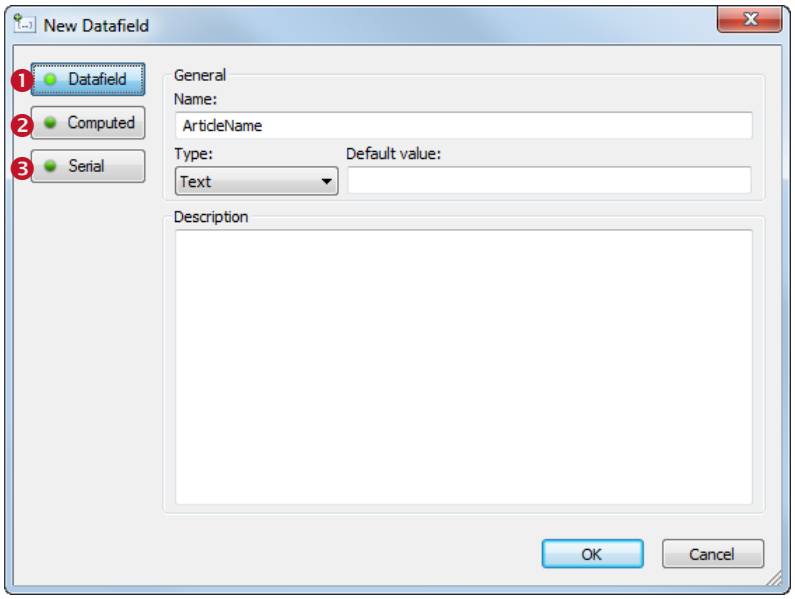


Figure 40: New Datafield Dialog

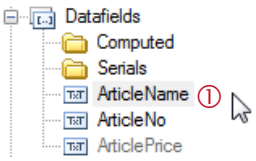
Depending on its purpose, choose the kind of datafield that you want to create. The following options are available:

❶ Datafield (default – see 10.3.1)
❷ Computed (see 10.3.2)
❸ Serial (see 10.3.3)

If you want to reference a field in a datasource, your choice will always be option ❶.

For computing the content of the datafield using an expression, your choice will be option ❷.

For generating serial numbers choose ❸.



Once the datafield is inserted you will find the appropriate entry in the design tree: Normal datafields are listed directly under the "Datafields" branch (see ①). Computed fields and serials are listed in the corresponding subfolders.

Used datafields are displayed in black. Unused (= unassigned) datafields are displayed in gray.

10.2.1.2 Create Datafield Definitions Automatically

When editing a datasource all datafield definitions can be created automatically (see section 13.4). **TFORMer** suggests to create and assign a new datafield for each field in the datasource which is not already used in the current layout (or repository). The data type *"Text"* is assigned automatically.

Using this function you can create numerous datafield definitions automatically. All field names of an existing datasource are imported.

10.2.2 Datafield Properties

The datafield properties can either be edited in the properties window, or in the *"Edit Datafield"* dialog.

For a description of datafield properties, please refer to sections 10.3 and 0.

10.2.3 Use Datafields in Layouts

Once a datafield was defined it can be used

- as element content in
  - text elements – see section 8.3.1,
  - barcode elements – see section 8.3.2 and
  - picture elements (dynamic logos) – see section 8.3.3,
- in control expressions (e.g., printing conditions – see sections 12.2 and 12.3) and
- in computations (e.g., as operand for additional computed fields – see section 10.3.2).

10.2.3.1 Element Content

To assign a datafield as element content use either of these two methods:

1. Double-click on the element. An edit dialog will be opened (see chapter 9). In this dialog select the desired datafield from a list.
2. *Drag and drop* the datafield from the design tree to the layout view (see ❶ below). The left or the right mouse button may be used for the drag and drop operation. When using the right mouse button a few insert options can be selected.

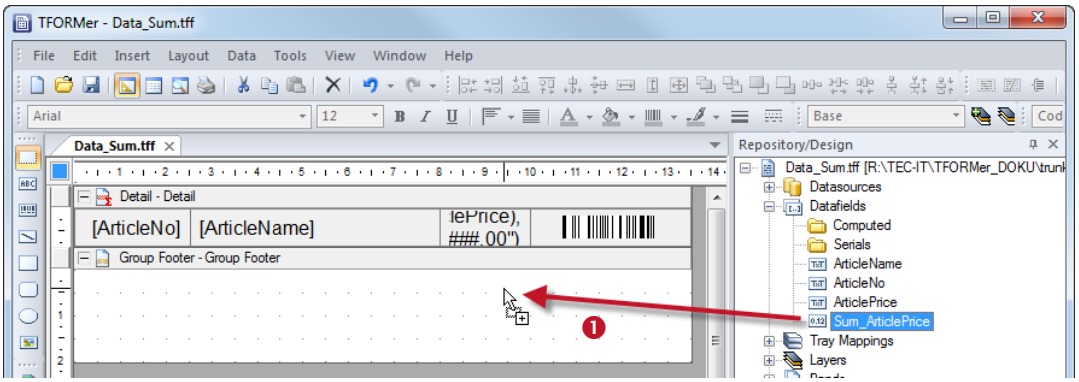


Figure 41: Drag and Drop a Datafield

10.2.4 Edit a Datafield

To edit an existing datafield double-click it in the design tree. The *"Edit Datafield"* dialog will appear (see section 10.3). Modify the required settings and then confirm with *OK.* Alternatively use the properties window for editing the datafield properties.

10.2.5 Rename a Datafield

To rename a datafield select it in the design tree and thereafter press *F2* (or click again on the datafield with the left mouse button). Now you can directly edit the datafield name. Alternatively use the properties window or the *Edit Datafield* dialog to rename a datafield.

> ► Attention: When renaming a datafield **TFORMer** will NOT automatically update all references to this datafield (e.g., in text elements, pre-/post-evaluations, printing-conditions, …). This has to be done manually (see also section 10.4). – Invalid references will throw an error and are marked with red color!

10.2.6 Delete a Datafield

You can delete a datafield in various ways: Select the datafield in the design tree and press the *Del* key on the keyboard. Alternatively use *Delete* in the pop-up menu, the delete symbol ✕ in the toolbar or *Edit* ► *Delete* from the menu.

> ► Attention: When deleting a datafield, **TFORMer** will produce an error in all bands and elements where the datafield is used (e.g., in text elements, pre-/post-evaluations, printing-conditions, …). Affected elements are marked with red color (see section 10.4).

10.3 The Edit Datafield Dialog

10.3.1 Standard Datafields

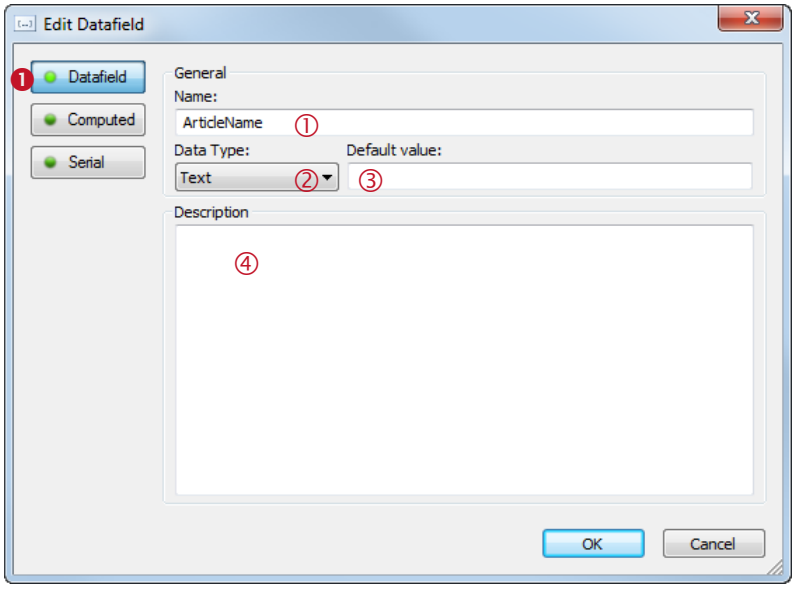


Figure 42: Edit Datafield Dialog – Datafield

If the datafield is supposed to reference the fields in a datasource, select the *Datafield* option ❶.

In ① enter a name for the datafield. Depending on the field type in your datasource you may adjust a suitable type in ② (Text, Number, Floating-point). However, for most cases the type *"Text"* will be fine. In ③ you can specify a default value. This value will always be used if no other value is computed respectively provided by the datasource. In ④ you can add an arbitrary description text.

> ► Please note: Before you can use the datafield to access the data from an external datasource you have to create the necessary field bindings (see section 13.4)!

10.3.2 Computed Fields

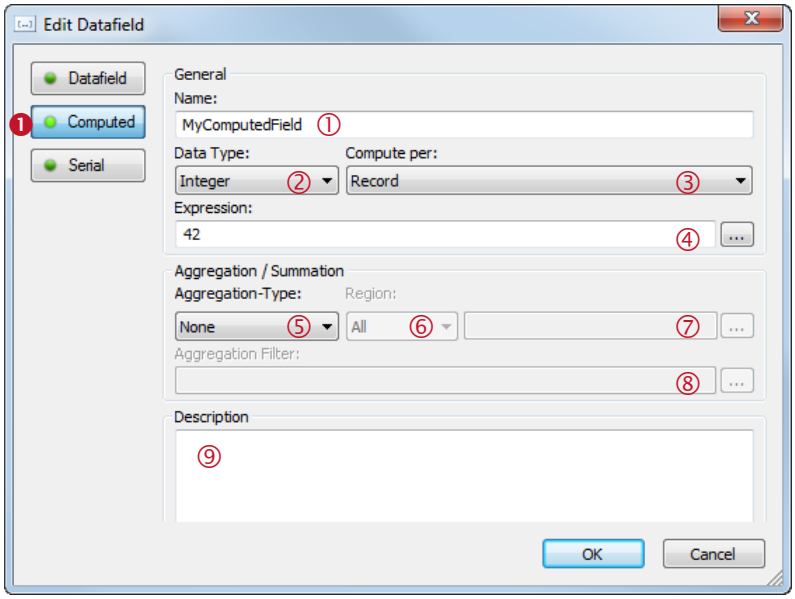


Figure 43: Edit Datafield Dialog – Computed

If the datafield is supposed to return the result of a computation, select the *Computed* option ❶.

In ① enter a name for the datafield. In ② select the datafield type. The computed value can either be a text, an integer or a floating-point number. The field value is calculated by evaluating the expression in ④. In ③ you can choose, whether the expression is computed for every record or for every record copy.

Example:
If you set the expression in ④ to “NumRecordCopies” the datafield will return the number of copies, which is adjusted in the datasource.

A typical application for computed fields is the aggregation / summation of other datafield values. In ⑤ you can select one of the following aggregation types:

Selection	Description
None	Do not calculate any sums.
Running Average	Calculate the arithmetic average for all values within each enclosed region ⑥ up to the current record. – This means, the expression ④ is calculated for each record respectively record copy (see ③). The result is then added to the series of values within region ⑥. The datafield provides the average for these values. Pease note: The average of all(!) records is not available before the last record!
Running Sum	Calculate the sum for all values within each enclosed region ⑥ up to the current record. – This means, the expression ④ is calculated for each record respectively record copy (see ③). The result is then added to the series of values within region ⑥. The datafield provides the sum of these values. Pease note: The sum of all(!) records is not available before the last record!

► Please note: These functions are only available for numeric data types (integer, floating-point) and not for text!

For *Running Sums* and *Running Averages* specify the region in which the values shall be evaluated (see ⑥):

Selection	Description
All	Do one aggregation within the whole range of the print-job.
Page	Do a separate aggregation within the range of each new page.
Label	Do a separate aggregation within the range of each new label.
Group	Do a separate aggregation within the range of each new group. In order to identify the group enter the same condition as for the group header/footer (property *"Group By")* in field ⑦. Whenever the result of the expression changes a new group is started.

When using a filter in field ⑧, single values will only be considered if the expression returns true. Thus you can define, which values are considered for the aggregation, and which not.

In ⑨ you can add an arbitrary description text.

10.3.2.1 Example: Calculate Sum

In the following example we will summarize all listed prices in a report. The result is printed underneath the last detail band.

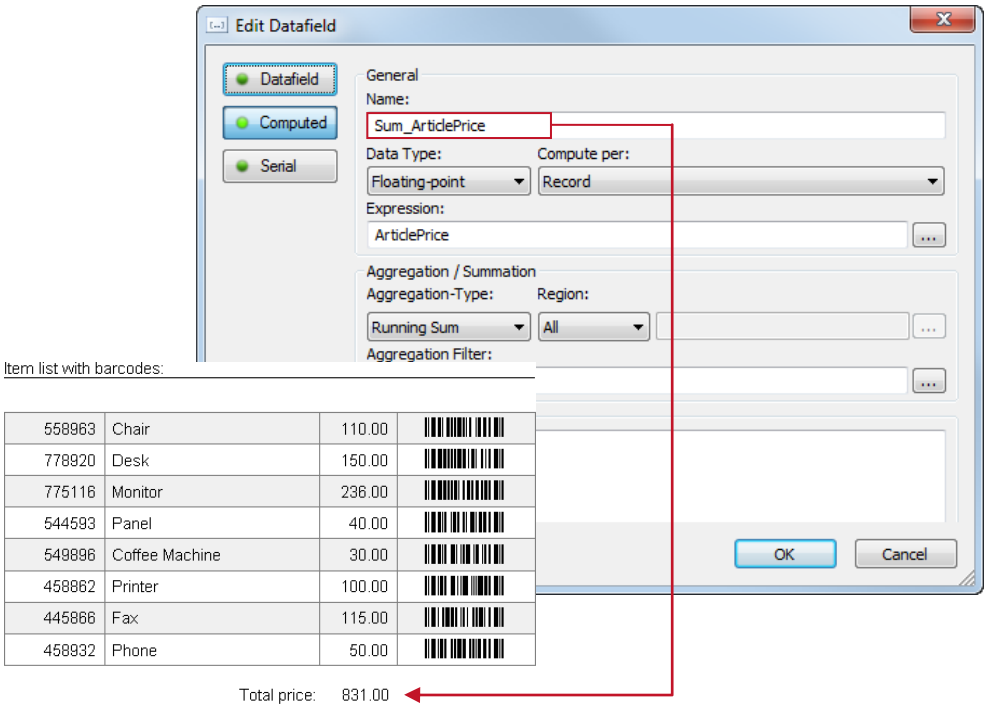


Figure 44: Calculate Sum

First create a new computed datafield (e.g., "Sum_ArticlePrice"). Set the data type to *"Floating-point"* (prices are usually given with decimal digits). As expression enter the name of the datafield, that you want to sum up (in this case "ArticlePrice"). Choose the aggregation type "Running Sum" and confirm with *OK*.

The newly created datafield "Sum_ArticlePrice" can now be used in the layout. Place it in the report footer to print the sum underneath the last list item. You can print the sum directly, or you can format it. Use the following expression to create a number with up to eight digits before the decimal point and exactly two digits after the decimal point:

```
[Format (Sum_ArticlePrice, "#######0.00")]
```

If you want to, you might also use additional digit grouping:

```
[Format (Sum_ArticlePrice, "##,###,##0.00")]
```

10.3.3 Serial Numbers

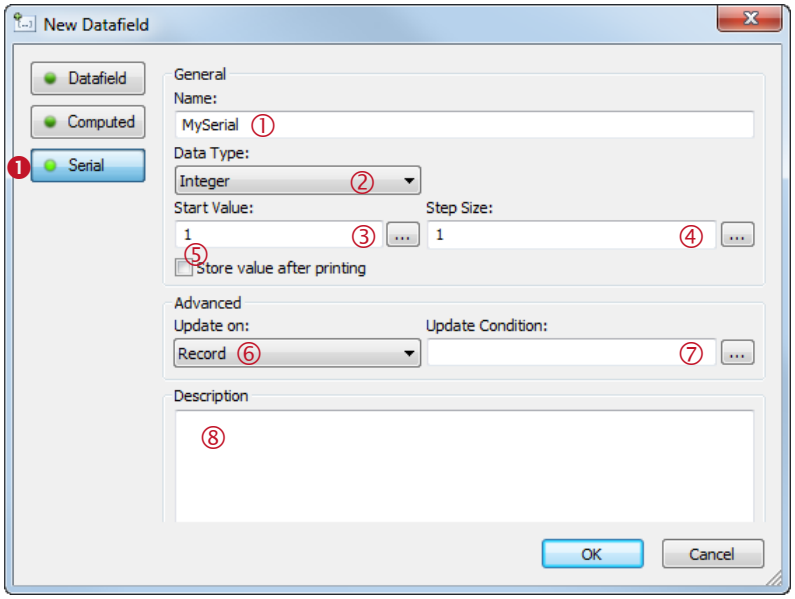


Figure 45: Edit Datafield Dialog – Serial Number

If the datafield is supposed to be a serial number calculated by **TFORMer,** select the *Serial* option ❶.

In ① enter a name for the datafield. In ② select the datafield type. A serial number can either be an integer or a floating-point number. Enter the start value in ③ and the increment in ④. If you want **TFORMer** to remember the last value which was printed you have to select option ⑤. In this case **TFORMer** will continue with the next serial number on the next print-out.

Instead of making **TFORMer** remember the last serial value you can also provide each starting value via an expression (compute it, read it from a datafield). To do this, click button ... . The expression editor will open.

In field ⑥ you can specify, whether the serial number should be incremented for each new

- Document,
- Label,
- Page,
- Record (default) or
- Record copy.

When using an update condition in ⑦ the serial number will only be incremented if this condition returns true.

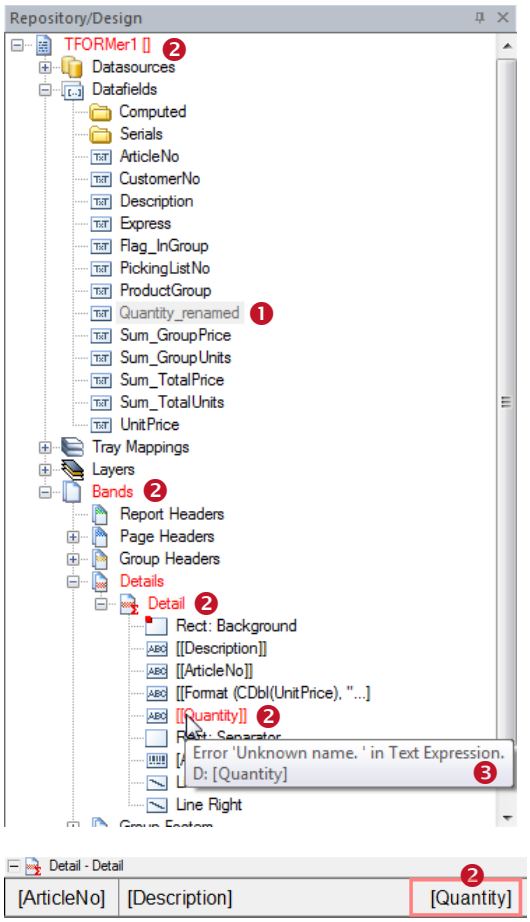
In ⑧ you can add an arbitrary description text.

After you are done with editing close the dialog with *OK.* The serial number can now be used in the layout. You can print the counter directly, or you can format it. Use the following expression to create an eight digit number with leading zeros:

```
Format (MySerial, "00000000")
```

10.4 Faulty Datafield References

When renaming or deleting a datafield, all references to this datafield will become invalid immediately (e.g., in a text element, in the printing condition of a band, etc.). The concerned bands and elements are marked as corrupted in red. In order to fix the problem all broken references have to be replaced manually.



In this example the datafield ”Quantity” ❶ was renamed to “Quantity_renamed”.

The original datafield name is still in use in various bands and elements. For this reason some errors are displayed (❷).

When hovering the mouse over one of the red objects a short error description is displayed (❸).

11 Expressions

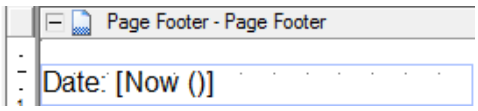
11.1 Introduction

Expressions are used for computing element content or controlling output behavior during print-time. Expressions can be used

- as dynamic content in text, barcode and image elements (see section 8.3).
- as printing conditions for bands and graphical elements (see section 12.2).
- as printing conditions for layers (see section 12.3).
- for computed fields and serial numbers (see sections 10.3.2 and 10.3.3).
- for pre- and post-evaluations in bands (see section 12.4).
- to compute the name of the generated output or spool file (see section 6.3.5).
- to filter the input data (see section 13.6).

The syntax used by **TFORMer** is similar to the programming language C.

Example:



This text element contains formatted text. In line with the text the expression *"Now()"* was inserted. It will return the current system date.

Date: 02.05.2009

During print-time the expression is evaluated: Instead of *[Now()]* the current date is printed.

11.2 Expression Builder

Expressions are edited using the *Expression Builder*:

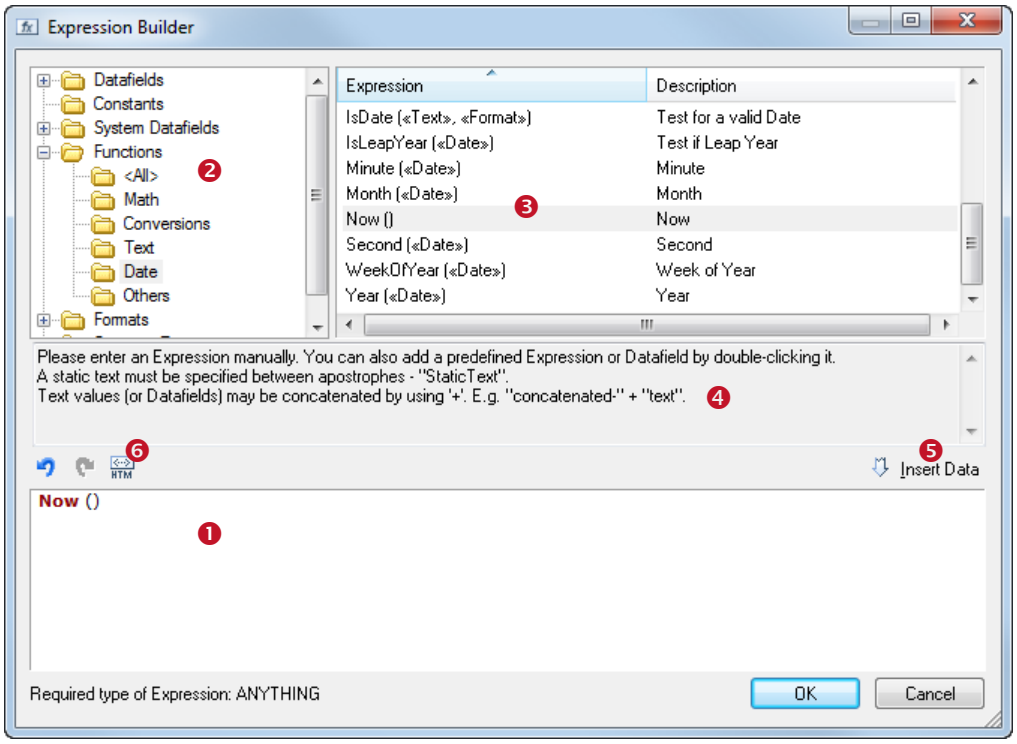


Figure 46: Expression Builder

Field ❶ shows the current expression. To extend the expression in ❶ you can select one of the predefined

- *Datafields*
  The datafields available in your layout.
- *Constants*
  True, False and *Linefeed* ("\n").
- *System Datafields*
  Datafields that are maintained by **TFORMer** automatically (the current page number, the band name, …)
- *Functions*
  Mathematical functions, conversion functions, string manipulation, …
- *Formats*
  Common formats for numbers, date and time values.
- *Common Expressions*
  Frequently used expressions like page numbering ("Page N of M"), the current date and time, currency formatting, …

First select the category in ❷. Then mark the required item in ❸ and click *Insert* ❺ (or just double-click on the list item). The item will be inserted at the actual cursor position in ❶. The text area ❹ shows additional information on selected items.

If you click *OK* the expression will be validated. Only expressions with a correct syntax are accepted.

A list of all available functions, constants, system datafields, formats and common expressions is available in Appendix C.

For text elements, additionally the HTM button ❻ is available. When pushed, the evaluated expression in ❶ will be interpreted as HTML. Thus you can provide additional format specifications within the text. HTML-formatted expressions are marked with angle brackets "<>" (see section 9.3.3).

11.3 Evaluation Order

The over-all evaluation order for expressions is defined as follows (see also figure below):

1. *Printing Condition* of the *band*
2. *Pre-Evaluation* of the *band*
   (Can be used for computing *datafields)*
3. *Printing Conditions* of *layers*
   (These are computed for every band, the results are used later when printing the elements)
4. *Printing Condition* of *elements* in the *band*
5. *Dynamic contents* for *elements*
6. *Post-Evaluation* of the *band*

Steps 2 to 6 will only be performed if the band is printed – in other words: if the printing condition of the band is true.

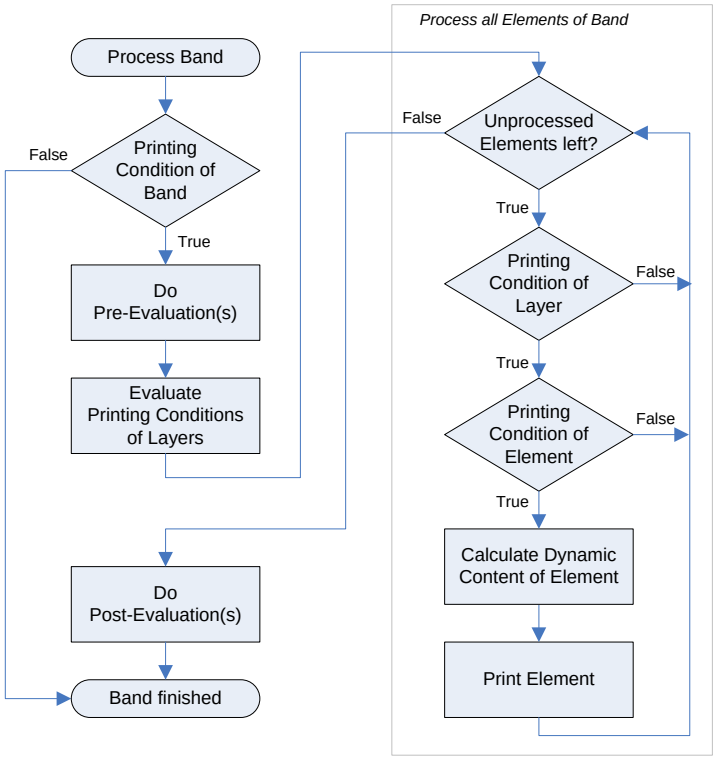


Figure 47: Evaluation Order

12 Smart Layouts

12.1 Introduction

TFORMer offers a number of features, which give you additional flexibility for formatting the output:

- *Printing Conditions*
  (Control whether single bands or elements are printed)
- *Printing Layers*
  (Assign elements to layers and control the printing of the layers via printing conditions)
- *Pre- and Post-Evaluations*
  (Perform computations which are specific to a certain band)
- *Invisible Bands*
  (Control special document layout features)
- *Tray Control*
  (Switch printer trays – even within a print job!)

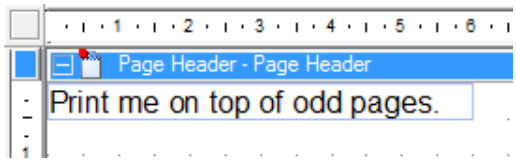
12.2 Printing Conditions

Printing conditions are used to control the visibility of elements, layers or bands during print-time.

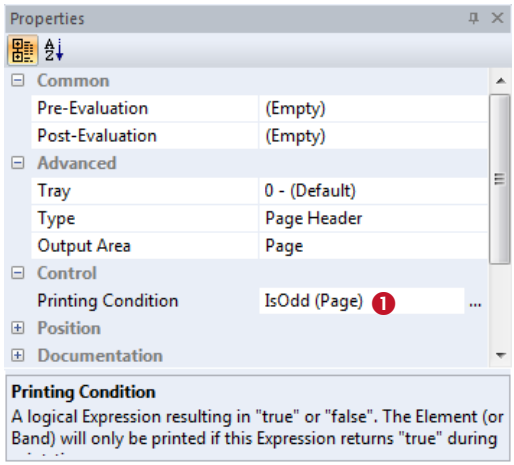
A printing condition is an *expression* (see chapter 11) which returns either *true* or *false.* Printing conditions can be assigned to elements (text elements, lines, …), to entire bands and to printing layers (see below). If a printing condition returns *false* the corresponding object will not be printed.

► Please note: The result of a printing condition will be converted to the data type *"Integer"* (numerical value) if required. A value of 0 is interpreted as *false.*

12.2.1 Edit a Printing Condition



If you want to edit a printing condition first select the object (e.g., the *page header).* Then go to the properties window.

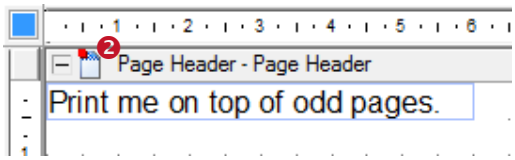


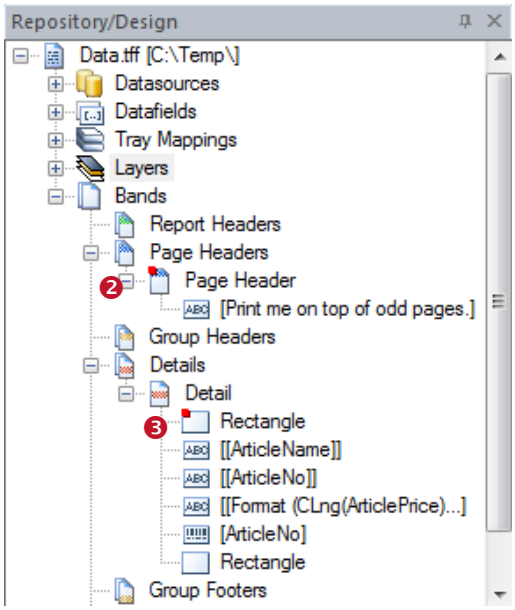
Activate the *Control* group and click anywhere in ❶ (or on the … button) to open the *Expression Builder* (see section 11.2). Alternatively use the icon in the toolbar, or right-click on the element and select *Printing Condition…* from the pop-up menu.

Use the *Expression Builder* to edit the printing condition. When you are done confirm with *OK.*

12.2.2 Visual Cues for Printing Conditions

If you have applied a printing condition, the band or the element will be marked with a small red dot:





Bands are marked in the layout view and in the design tree (❷). Graphical elements are marked in the design tree only (❸).

12.2.3 Examples

12.2.3.1 Different Headers or Footers

To print different headers and footers create at least two bands of the same type. Then use the printing condition to decide which band shall be printed. E.g., create one page header for odd pages and one page header for even pages. The expressions “IsOdd(Page)” respectively “IsEven(Page)” will serve as printing condition.

12.2.3.2 Alternating Background Colors

To print rows with alternating background colors draw a filled rectangle in the background of the detail band. Then set the printing condition to “IsOdd(Record)”.

12.3 Printing Layers

Printing layers are used to control the *visibility of multiple elements with a single printing condition.* The printing condition of the layer decides whether all elements assigned to that layer will be printed or not.

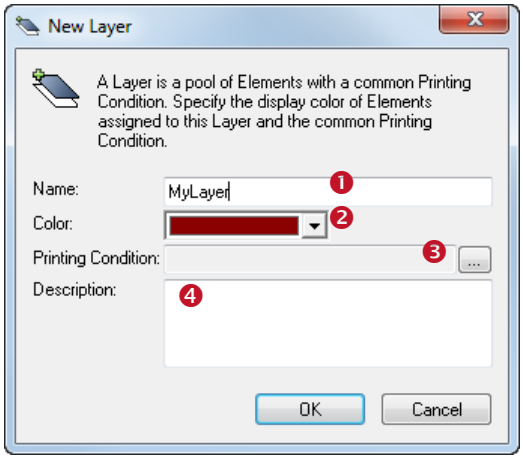
By default only one printing layer, the “Base” layer, is defined. Newly inserted design elements are always added to this layer.

Please note:

- One element can be assigned to one printing layer only.
- Layers do NOT influence the z-order of elements!
- Elements assigned to a printing layer may still use additional element-specific printing conditions.

12.3.1 Create a New Layer

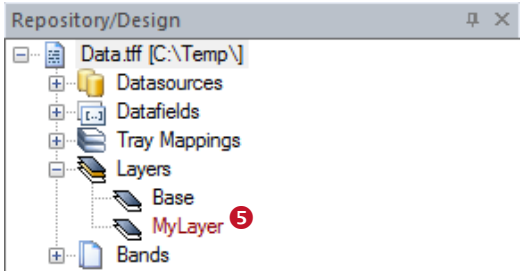
You can create a new printing layer using either of the following methods:



- Select *Insert* ► *Layer…* from the menu.
- Click on the *"Add Layer"* icon in the toolbar.
- Right-click on the form, on a band, or on the *"Layers"* entry in the design tree and select *New Layer…* from the pop-up menu.

The *"New Layer"* dialog will appear.

In this dialog enter the name (❶), the color (❷) and the printing condition (❸) for the layer. In ❹ you can enter an arbitrary description text.

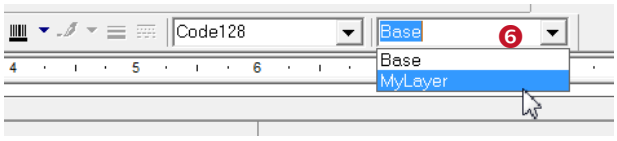


Once the printing layer is inserted you will find the appropriate entry in the design tree (❺).

For a description of all layer properties, please refer to Appendix A.4.

12.3.2 Assign Design Elements to a Layer

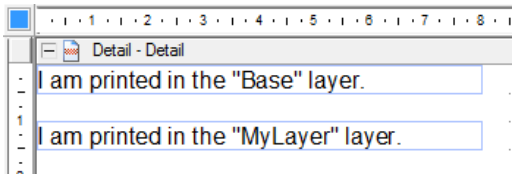
To assign design elements to a specific layer, perform the following steps:



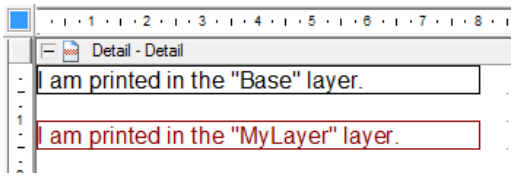
First select all elements that you want to assign to the printing layer. Then select the layer in ❻. Alternatively right-click on the selection and select *Assign Layer* ► *MyLayer* from the pop-up menu.

12.3.3 Display Layer Colors

If you want to see which design elements are assigned to which layer, you can turn on the display of layer colors:



By default all design elements are displayed in their printing color (see left).
Click on the icon *"Toggle Display Color"* in the toolbar, or select *View ► Layer Colors* from the menu to change the display mode.



If the display of layer colors is enabled, all design elements are drawn in the color as adjusted in the layer properties (see ❷ on page 66).
Click the icon again to switch back to normal.

12.3.4 Hide Layer Contents

To hide all elements which are assigned to one layer, first select the layer in the design tree and then use one of the following methods:

- Right-click on the layer in the *design tree* and deselect *Visible* in the pop-up menu.
- Double-click on the layer in the *design tree.*
- In the *properties window* go to the *Common* group and set the property *Visible* to "False".

12.4 Pre-Evaluation and Post-Evaluation

The pre- and the post-evaluation can be used to compute datafields during print-time.

With this type of evaluation each band can perform arbitrary computations. You have the choice between computations before a band is printed *(Pre-Evaluation)* and computations after a band was printed *(Post-Evaluation).* If the band is not printed at all (if the printing condition of the band returns false), no evaluations are performed.

Usually pre- and post-evaluations are used for

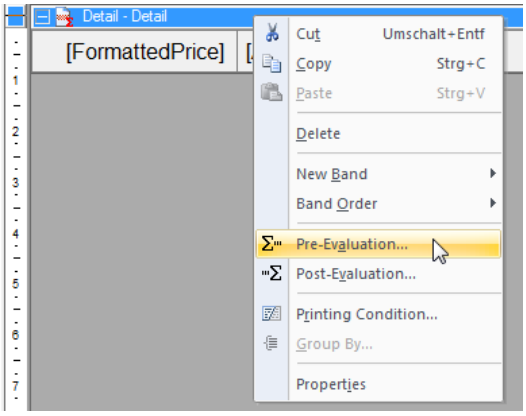
- implementing computations which are specific to a band,
- implementing counters,
- formatting data prior printing.

For example, the pre-evaluation in the *report header* may be used to evaluate one or more expressions at the "beginning" of the report.

Please note:

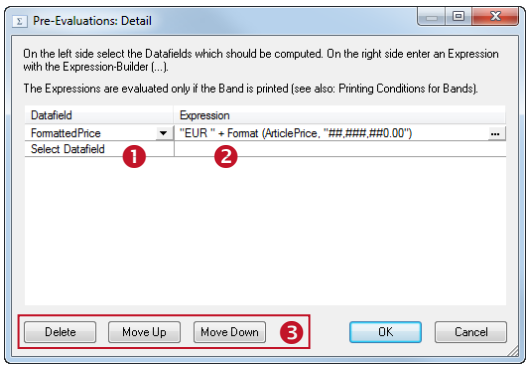
► When assigning a value to a datafield during print-time (computed field, pre-/post-evaluation), this datafield will be excluded from data import. It will no longer be filled with any values from the datasource. – A datafield which provides the data from a datasource (which is bound to a sourcefield) may therefore never be filled with computed values. Always use a separate datafield for computations!

12.4.1 Edit Pre- and Post-Evaluations



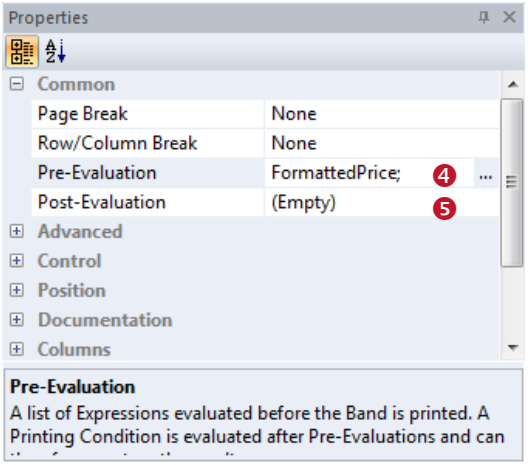
Right-click on the required band and select *Pre-Evaluation…* or *Post-Evaluation…* from the pop-up menu. Alternatively use the toolbar icons Σ‴ and ‴Σ.

The following dialog will appear:



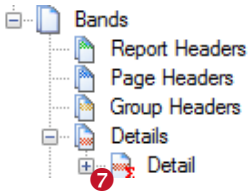
First select the datafield that you want to compute in ❶. Then enter an expression in ❷. Click on the ... button to open the *Expression Builder* (see section 11.2).

The computations are evaluated in the order as they appear in the dialog (from top to bottom). If you want to change the computation order or if you want to delete a computation, use the buttons in ❸.



The *Common* group in the *properties window* of the band shows all datafields calculated in the pre-evaluation ❹.
The *post-evaluation* ❺ is empty. No datafields are computed.

12.4.2 Visual Cues for Pre- and Post-Evaluations



If you have entered a pre- or a post-evaluation expression, the band will be marked with a small red sigma symbol in the layout view ❻ and in the design tree ❼.

12.5 Invisible Bands

Invisible bands are used to perform computations or to control the output behavior of the layout.

An invisible band is a band with zero height. It does not contain any graphical elements like text or lines, and therefore nothing is being printed.

Invisible bands can be used for:

- Computations *(pre-* and *post-evaluation).*
  You can initialize a *datafield* in the *report header* (at the beginning of the printing process).
- Feed Control (insert page- or column-breaks, triggered by printing conditions).
- Tray Control (see section 12.6).

Please note:

- If the printing condition for the band returns *false,* it will not be printed. This means, no pre- and post-evaluations are performed, no page- or column breaks are inserted and no tray will be selected.

12.6 Tray Control

The tray control is used to select or switch printer trays during print-time.

For every single page of the output, you can dynamically select a tray on the target printer. This is useful for printing the first page of an invoice on a letterhead, or for adding an envelope to a print-out. You could also print a number of labels, and add a cover sheet which is printed on non-sticky paper. In **TFORMer** all this can be done in one single print-job. The selection of the trays is performed during print-time.

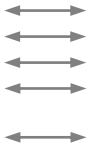
The tray selection works completely device-independent: **TFORMer** uses logical tray numbers (tray 1 to tray 10). Thus it is possible to select the trays without having to take care for the actually used hardware or printer driver. A tray is always selected by its logical number.

The mapping of the device-dependent printer trays to the logical tray numbers is done in the *Tray Mappings.* These tray mappings have to be configured beforehand. Then, for each print-job, you can select the appropriate mapping.

Example:

Assume you are printing one layout on two different printer models (Printer A and Printer B):

Name	Tray Mappings for Printer A
Tray 1	Automatically Select
Tray 2	Upper Paper Tray
Tray 3	Manual Paper Feed
Tray 4	Envelope, Manual Feed
…	
Tray 10	A4



Name	Tray Mappings for Printer B
Tray 1	Auto
Tray 2	Tray 1
Tray 3	Tray 1 (Manual)
Tray 4	Envelope Feeder
…	
Tray 10	A4

The first page of the layout should be printed using the manual paper feed which is named differently on both printers (due to different printer drivers). Using the tray mappings as shown above, the tray number 3 can be assigned in both cases. When printing, take care to select the appropriate tray mapping for the actual output device.

The following steps are necessary for tray control:

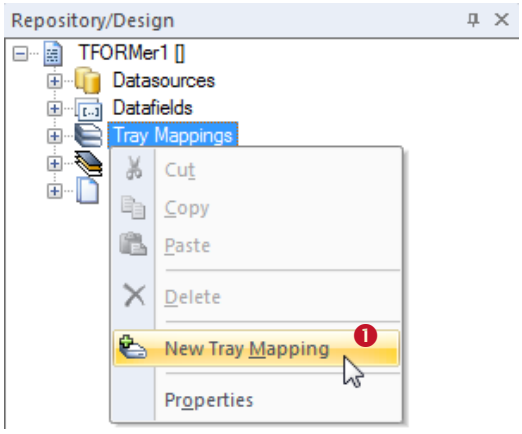
- Create and configure the required tray mappings.

- Perform the tray selection in the band properties.
- Select the correct tray mapping for printing.

Please note:

- The "_Default_" tray mapping is always available.
- You can define as much tray mappings as required (this is useful when printing one layout on different printers).
- When using form names (e.g., A4 or Letter) in the tray mappings, be careful to configure your printer drivers correctly. Otherwise the mapping between a form name and the printer tray will not work.

12.6.1 Create a New Tray Mapping

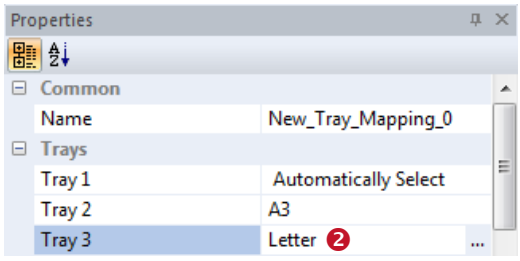


To create a new tray mapping, right-click on the folder *"Tray Mappings"* in the design tree and select *New Tray Mapping* from the pop-up menu (see ❶). A new entry "New_Tray_Mapping_0" will be created.

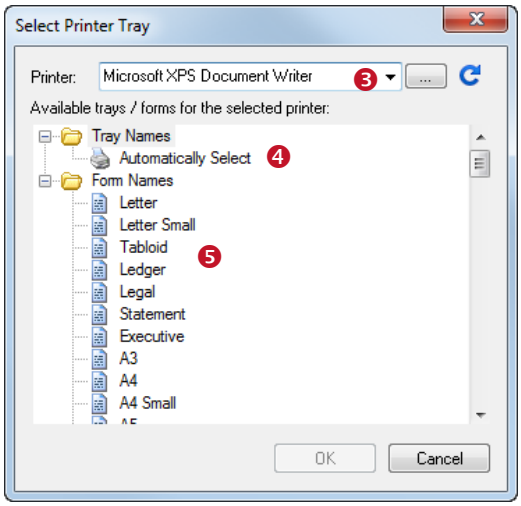
To rename the tray mapping select it in the design tree and thereafter press *F2* (or click again on the item with the left mouse button). Now you can directly edit the name. Alternatively use the properties window to rename a tray mapping.

12.6.2 Configure Tray Mappings

Each tray mapping has ten logical trays which can be filled with device specific physical trays. The trays are assigned as follows:



Enter the name of the logical tray directly in ❷ or select it from the list provided by the printer driver. Click on the ... button to open the dialog below.



First select the target printer in ❸. Then select one of the listed items: You can choose a *"Tray Name"* (❹) or a *"Form Name"* (❺).

In order to refresh the try list for manually entered printer names, use the button .

Click *OK* to assign the selection.

12.6.3 Tray Selection in the Layout

To select a tray in the layout simply enter the required tray number (1 to 10) in the *Tray* property of the band. By default *"Tray 0 - (Default)"* is pre-selected for every band. This means, the settings of the current printer driver are used for printing.

- Tray 0 (default) uses the settings of the current printer driver.
- Tray 1 to 10 can be pre-configured in the tray mappings.
- The first band that is printed on a page (e.g., the *page header)*, performs the tray selection. Tray settings of subsequent bands which are printed on the same page are ignored.

12.6.3.1 Example

If you want to print the first page of a report on a letterhead, assign the required tray to the report *header.* After the first page you have to switch back to normal paper. Therefore enter the number for the default tray in the page *header*.

12.6.4 Select the Tray Mappings for Printing

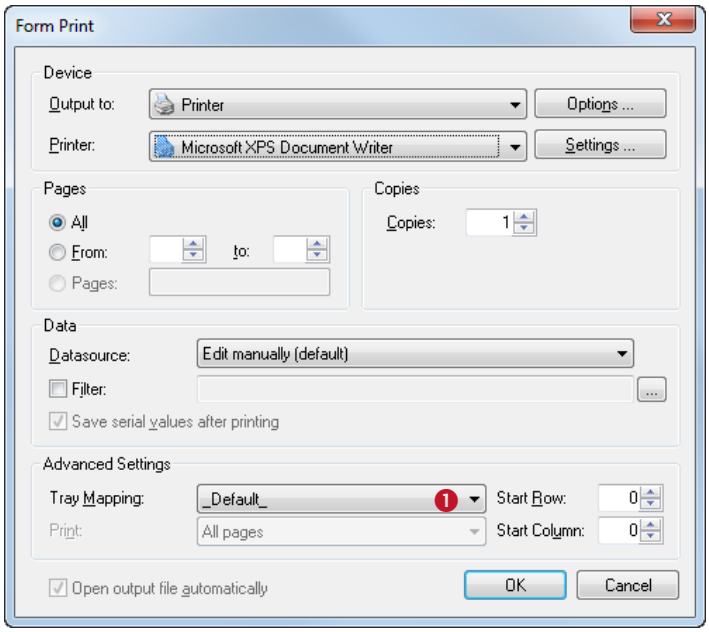


Figure 48: Select Tray Mappings in the Print Dialog

When printing, select one of the pre-configured tray mappings in ❶. For a more information on the print dialog, please refer to chapter 15.

13 Providing Data

13.1 Introduction

As mentioned in chapter 5 **TFORMer** requires data in addition to the layout for printing. The following datasource types are supported:

- *Manual Datasource*
  The default datasource allows you to edit every single data value manually (see section 13.3.1).
- *ODBC Datasource*
  The data will be imported from an ODBC connection (see section 13.3.3).
- *Flat Text File*
  The data will be imported from a text file (CSV, TSV,…) (see section 13.3.4).
- *XML File*
  The data will be imported from a XML file (see section 13.3.5).
- *TFORMer SDK API*
  This datasource is not selectable in the user interface. The data will be provided programmatically from outside **TFORMer Designer.** This method can be used by software developers. For details see section 15.3.

► For each layout you can create multiple datasources and switch between them as required. So you have the possibility to print one single layout with data from different datasources without modifying the layout.

13.2 The Data View

For managing datasources **TFORMer** offers a separate view, the *data view*. In the data view you can…

- inspect the data which is provided by the currently selected datasource,
- create, edit, rename and delete datasources,
- switch between different datasources,
- reload a datasource,
- set parameters for a datasource,
- bind sourcefields (the fields provided by a datasource) to datafields (the placeholders, which are used in the layout),
- and apply a filter to the current datasource.

To switch to the data view click on the *"Data"* tab ① at the bottom of the application window (see Figure 49 below) or select *Data* ► *Data View* from the menu. Alternatively press the keyboard shortcut *Ctrl+D.*

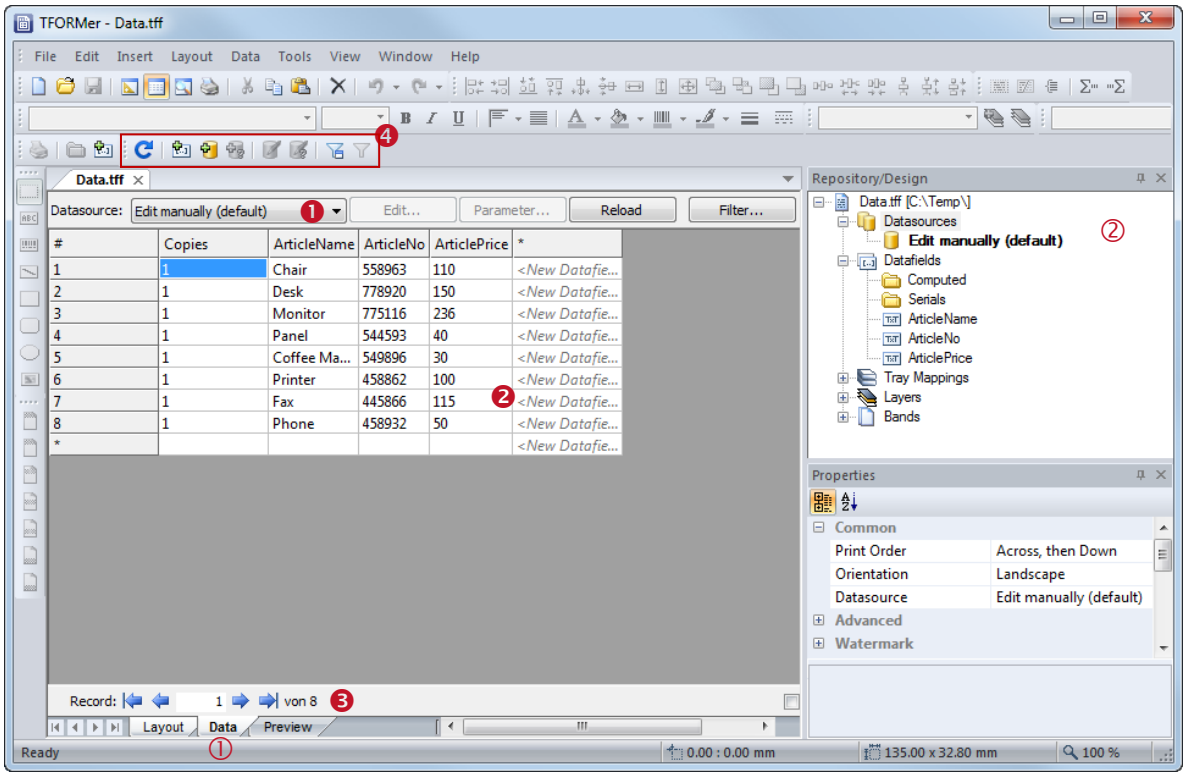


Figure 49: Data View

The *data view* is divided into the following areas:

- ❶ Datasource Selection
- ❷ Data Grid
- ❸ Record Navigation
- ❹ Data Toolbar

In the drop-down list ❶ you can set the active datasource or you can create new datasources (see section 13.3.2). By default, the manual datasource is selected. This datasource is used for entering data within **TFORMer** directly.

The design tree ② gives you an overview over all available datasources. The active datasource is displayed in bold. Right click on a datasource to open the context menu. The context menu allows you to perform various operations on a datasource (e.g., set it as active datasource).

The data grid ❷ shows the data which is provided by the active datasource. The grid is divided into rows and columns. The rows represent the data records, the columns represent the datafields (*ArticleName, ArticleNo* etc…).

The first column in the data grid is the "*Copies"* column. This column is always available. It tells **TFORMer** how often each single record should be printed.

The datafield columns are ordered alphabetically by default. Using the datafield property *"Display Order"* (see section 0) a custom display order can be specified.

The record navigation ❸ allows you to navigate to the next, previous, first or last record, or to a specific data record directly.

The data toolbar ❹ provides the following functions for manipulating datasources:

	Refresh	Reload the current data.
	New Datafield	Insert a new datafield.
	New Datasource	Create a new datasource.
	New Source-Parameter	Create a new source-parameter.
	Edit Datasource	Edit the datasource.
	Edit Source-Parameter	Edit the source-parameters.
	Edit Filter	Edit the current filter expression.
	Apply Filter	Apply the current filter expression.

- **TFORMer Designer** always uses the active datasource for printing.
- **TFORMer Designer** only prints the records which are shown in the data grid. This is usually the content of the active datasource, but the number of records may be reduced by a filter (see section 13.6).
- Sorting the records in the data grid is not supported. If the printing order has to be changed, please re-arrange the records manually or adjust the appropriate datasource definition (e.g., by inserting an ORDER BY instruction in the SQL statement).

13.3 Datasources

13.3.1 Manual Datasource (Default)

For each layout **TFORMer** provides a manual datasource (see ❶). This datasource is always available. It allows you to enter the required data directly into the data grid.

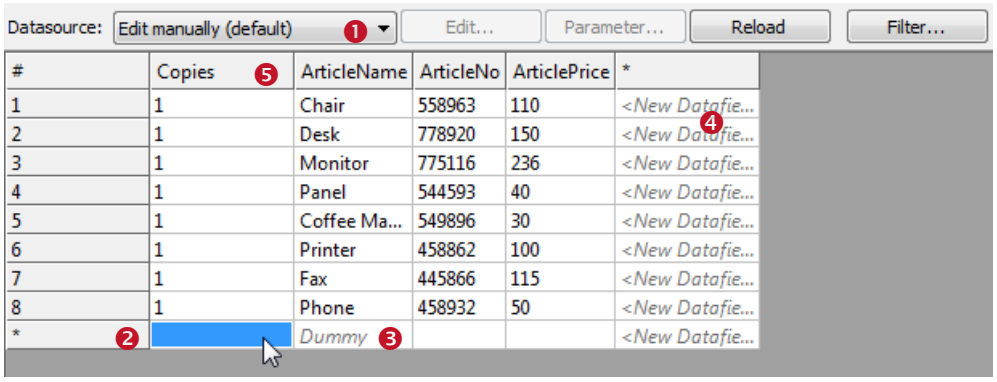


Figure 50: Edit Data Manually

To insert a new record move the cursor to the last data row ❷ (which is marked with a ” *“), enter the required data and confirm with enter. If a default value was assigned to a datafield, this default value is displayed as gray text (see ❸). It will be used, if no other value is entered.

To change an existing value in the data grid, select the required cell with the mouse and press *F2* (or double-click on the cell). This allows you to place the cursor at the required position inside the cell and to edit the content.

A new line within a cell is inserted by pressing *Ctrl+Enter*. Please note that this line break is not visualized in the data grid view. Though, it will be inserted on the print-out.

To select a row click on the record number in the first column “#”. It is possible to select more than one row by holding down the *Shift* or the *Ctrl* key. The selection can be deleted (press the *Del* key) or it can be copied and pasted into any of the other rows (*Ctrl+C* and *Ctrl+V*).

- Only the manual datasource allows you to edit the data within **TFORMer Designer** directly. For all other datasources (ODBC, flat text files and XML) the data grid is read-only.

- Read-only cells are marked with gray background color.
- Inserting new records is only possible in the last row ❷. You cannot perform an insert operation between two existing rows.

If required, you can create a new datafield by using the toolbar icon or by clicking in the column *<New Datafield…>* ❹.

- For the manual datasource the content of the new datafield can be edited immediately.
- When using an external datasource (e.g., ODBC) you need to provide the content for the new datafield via sourcefield binding (see section 13.4).

- The field *"Copies"* ❺ is not an actual datafield. It is used to determine how often a record is printed in the output.

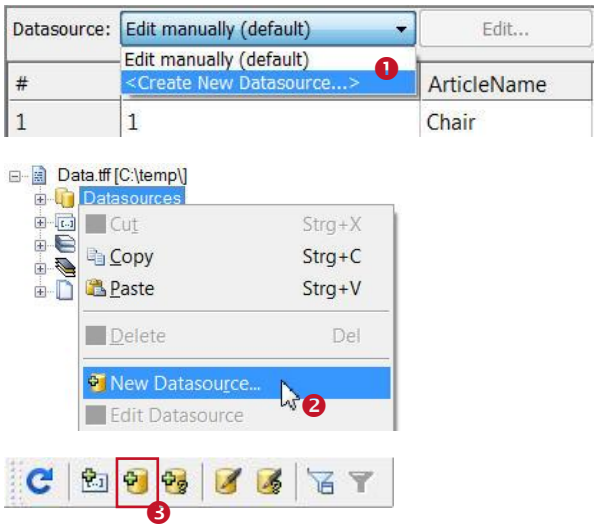
13.3.2 External Datasources

In addition to the manual datasource you can specify one or more of the following external datasources:

- ODBC Datasource (see section 13.3.3)
- Flat Text File (CSV, TSV, …) (see section 13.3.4)
- XML File (see section 13.3.5)

For managing these datasources use the operations described below:

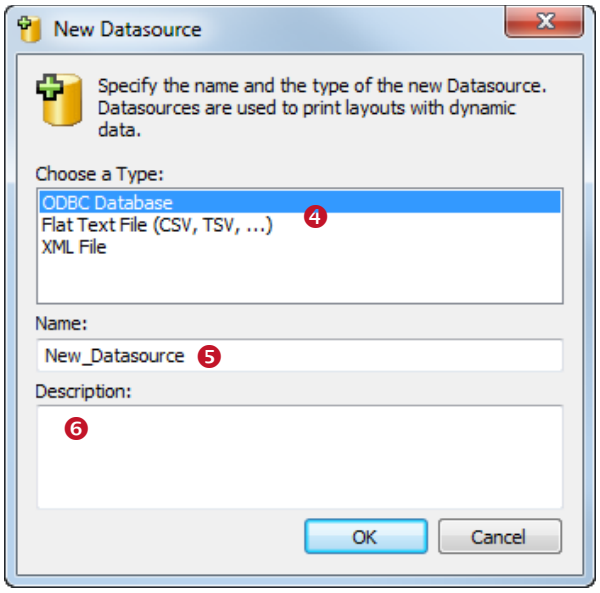
13.3.2.1 Create a New Datasource



Create a new datasource using either of the following methods:

- Select *<Create New Datasource…>* from the drop-down list ❶.
- Right-click on the *"Datasources"* entry in the design tree and select *New Datasource…* from the pop-up menu ❷.
- Click on the "*New Datasource"* icon ❸ in the data toolbar.

The dialog *"New Datasource"* will appear:



Choose one of the available datasource types in ❹. Specify a name (❺) and an optional description (❻) for the datasource. Then confirm with *OK*.

A dialog for setting up the respective datasource will appear. Setting up the different types of datasources is described in sections 13.3.3 to 13.3.5. Additional adjustments (which apply to all of these datasources) are discussed subsequently in sections 13.4 to 13.6.

Once the datasource is inserted and set up you will find an appropriate entry in the design tree.

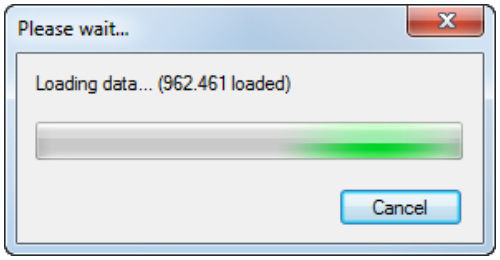
A newly created datasource is set as the active datasource automatically.

13.3.2.2 Loading Data and Progress Bar

For all external datasources **TFORMer** caches the data locally in order to provide a stable set of data for printing (you see the cached data in the data view). This means that the datasource is read completely, before a print-job or the rendering of the print preview is started.

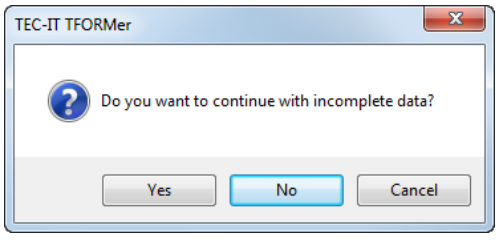
Whenever a new datasource is created or when switching to an existing datasource (see sections 13.3.2.1 and 13.3.2.6) the data from this source is read automatically. However, it is also possible to manually force the reloading of the data (see section 13.3.2.7).

For large amounts of data, sometimes the loading may take several minutes. While loading the records **TFORMer** displays a progress bar:



This bar shows you the progress of the loading operation.

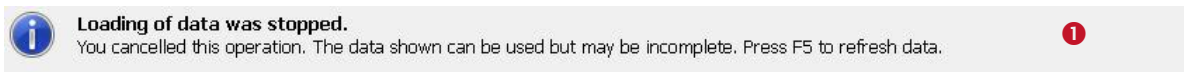
By clicking *Cancel* you have the possibility to cancel the operation:

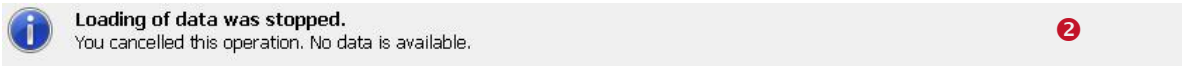


Yes aborts loading immediately. All data read so far is kept in the data cache. This results in an incomplete data cache. Note ❶ below will be displayed at the bottom of the data view or preview.

No aborts loading immediately. All data read so far is discarded. This results in an empty data cache. Note ❷ below will be displayed at the bottom of the data view or preview.

Cancel continues loading.





13.3.2.3 Edit a Datasource

To edit an existing datasource, open the *"Edit Datasource"* dialog using either of the following methods:

- Click on the *"Edit Datasource"* icon in the toolbar or use *Data* ► *Edit Datasource…* from the menu (only available in the data view and in the preview).
- Right-click on the datasource entry in the design tree and select *Edit Datasource …* from the context menu.
- Select the appropriate datasource from the drop-down list in the data view and click the *Edit…* button.

13.3.2.4 Rename a Datasource

To rename a datasource select it in the design tree and thereafter press *F2* (or click again on the datasource with the left mouse button). Now you can directly edit the datasource name. Alternatively use the properties window to rename a datasource.

13.3.2.5 Delete a Datasource

Datasources can be deleted in various ways:

- Select the datasource in the design tree and press the *Del* key.
- Select the datasource and press the delete symbol in the toolbar or choose *Edit* ► *Delete* from the menu.
- Right-click the datasource and select *Delete* from the context menu.

13.3.2.6 Switch between Datasources

To switch to the required datasource the following methods can be used:

- Use the drop-down list to switch to the appropriate datasource (see Figure 49, ❶).
- Right click on the appropriate datasource in the design tree and select *Set as Active Datasource* in the context menu.

13.3.2.7 Reload a Datasource

You can refresh the cached data any time by pressing the icon in the data toolbar or by pressing the *Reload* button in the data view.

13.3.3 ODBC Datasource

ODBC datasources are used to retrieve data from a database (Microsoft Access, SQL Server, Oracle,…). Each database with a suitable ODBC driver is supported. On Microsoft Windows the drivers for Microsoft Access and SQL Server are pre-installed. For more information on how to install additional ODBC drivers please refer to the documentation of the database system in question.

13.3.3.1 DSN (Database Selection)

In the *DSN* tab specify the ODBC connection:

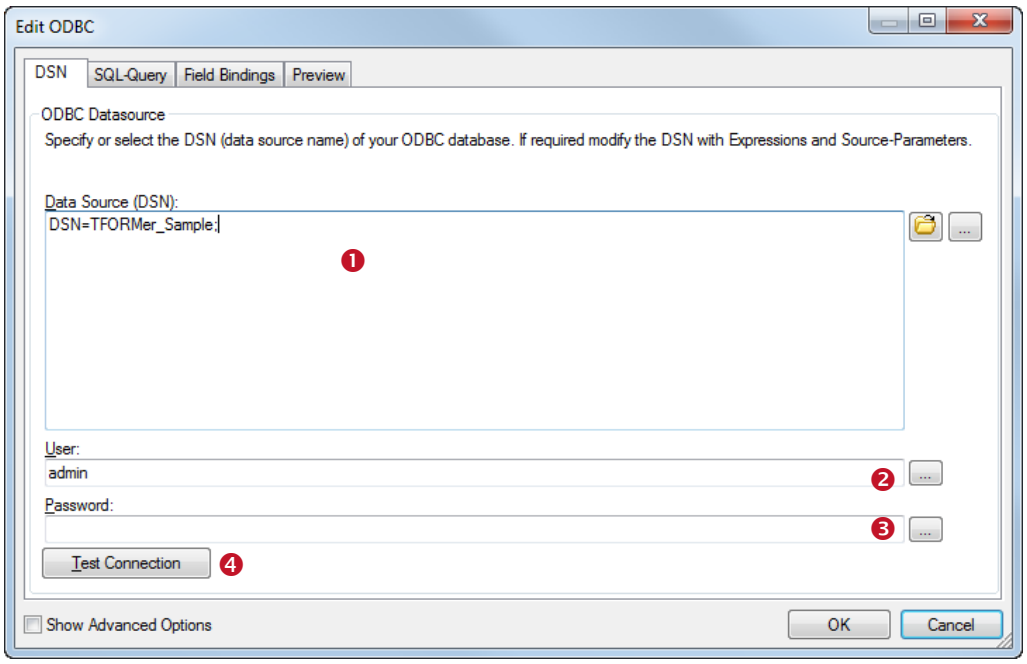


Figure 51: Data Source (DSN)

- *Data Source (DSN)* ❶
  Enter the connection string which identifies your required ODBC connection. You can select from a list of available datasources as defined under Microsoft Windows *(Control Panel ► Administrative Tools ► Data Sources (ODBC))* by pressing the button . Or you can enter the string manually by pressing the button .
- *Username* ❷
  Enter a username for the database connection by pressing the button .
- *Password* ❸
  Enter a password for the database connection by pressing the button .
- *Test Connection* ❹
  Use this button to test the connection settings.

► For advanced users: With the help of expressions and source-parameters you have the possibility to build dynamic connection strings (e.g., use a variable data source (DSN), user name and/or password). First open the expression builder for ❶, ❷ or ❸ by clicking the button, then use the corresponding button *Insert Source-Parameter…* or *Insert Expression…*. For details on expressions and source-parameters see chapter 11 and section 13.5.2.

13.3.3.2 SQL-Query

In the *SQL-Query* tab you specify the SELECT statement for fetching the data:

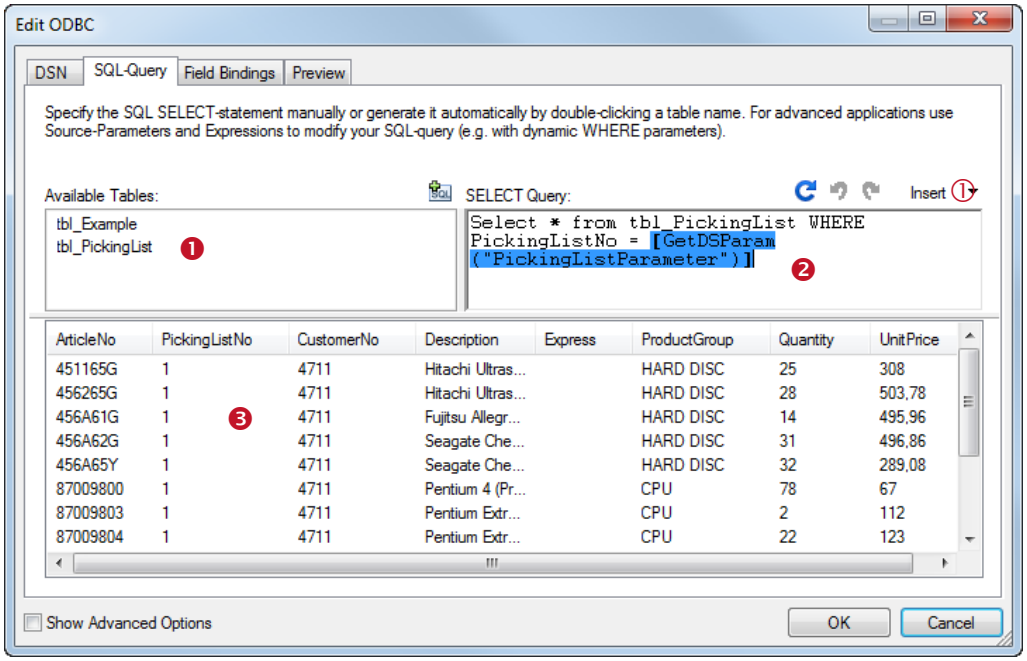


Figure 52: SQL-Query

- *Available Tables* ❶
  This window lists all available tables for the ODBC connection specified in the DSN tab.
- *SELECT Query* ❷
  The SQL query is used for data selection. This statement can be typed manually. Alternatively you can also generate a “Select *” statement by double-clicking on the table name in ❶ (or by selecting the table and clicking on the button).
- *Preview* ❸
  A preview of the selected data is displayed. To update this preview press the button .

> ► For advanced users: With the help of expressions and source-parameters you have the possibility to build a dynamic SQL SELECT statement. First place the cursor in ❷ on the required text position, then use the *Insert* button ① to insert an expression or a source-parameter. For details on expressions and source-parameters see chapter 11 and section 13.5.2.

13.3.3.3 Field Bindings

To complete the setup of the datasource adjust the required field bindings (see section 13.4).

13.3.4 Flat Text Files (CSV, TSV, …)

Text file datasources are used to retrieve data from a file (CSV, TSV, …).

13.3.4.1 File

In the *File* tab specify the text file to be loaded:

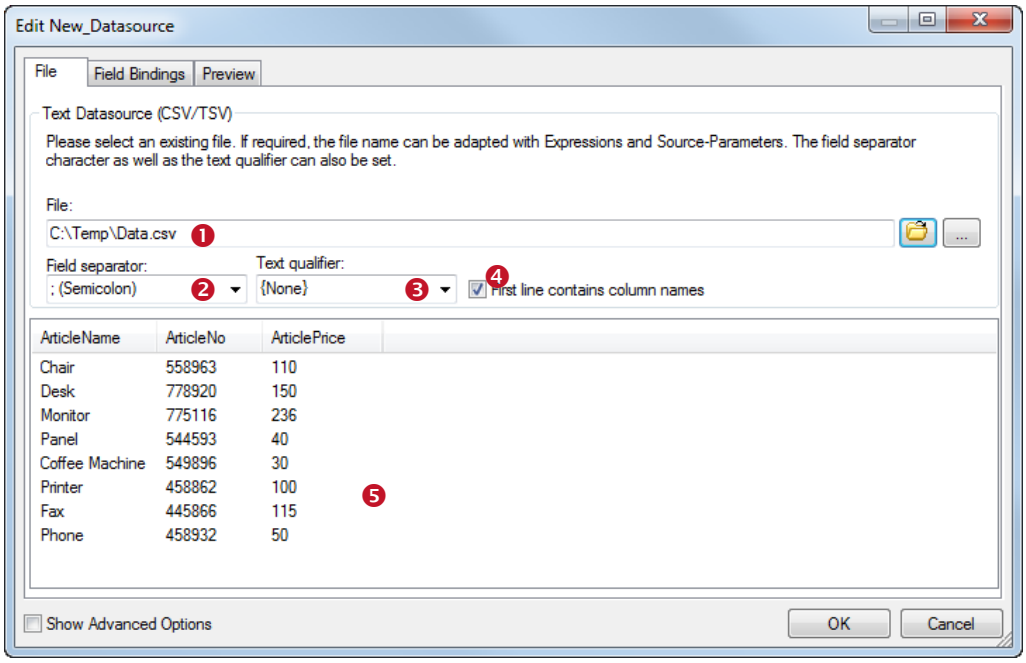


Figure 53: File Datasource

- *File* ❶
  Enter the name of the file which you want to use as datasource. You can select a file by pressing the button [📁] or you can enter the file path manually by clicking the button [...].
- *Field separator* ❷
  The field separator specifies the character used to separate the fields in the text file. Select a predefined separator character from the list or enter a custom separator character.
- *Text qualifier* ❸
  The text qualifier specifies the character which is used to enclose data values in the file (e.g., data values may be enclosed in quotes). This may sometimes be necessary to differentiate field values from the separator character. Select a predefined text qualifier character from the list or enter a custom character.
- *First line contains column names* ❹
  Specify whether the first line in the text file contains the column names or not. If selected **TFORMer** does not treat data in the first line as data values. Instead these names are used as the names of the sourcefields.
- *Preview* ❺
  A preview of the datasource values is displayed.

▶ For advanced users: With the help of expressions and source-parameters you have the possibility to build a dynamic file path and/or file name. First open the edit dialog for ❶ by clicking the [...] button, then use the corresponding button *Insert Source-Parameter…* or *Insert Expression…*.. For details on expressions and source-parameters see chapter 11 and section 13.5.2.

13.3.4.2 Field Bindings

To complete the setup of the datasource adjust the required field bindings (see section 13.4).

13.3.5 XML File

XML file datasources are used to retrieve data from a well-formed XML file. In addition to the XML file you can specify an optional transformation file (XSLT file). This file may be used to transform a custom XML file into a structure accepted by **TFORMer**.

13.3.5.1 File

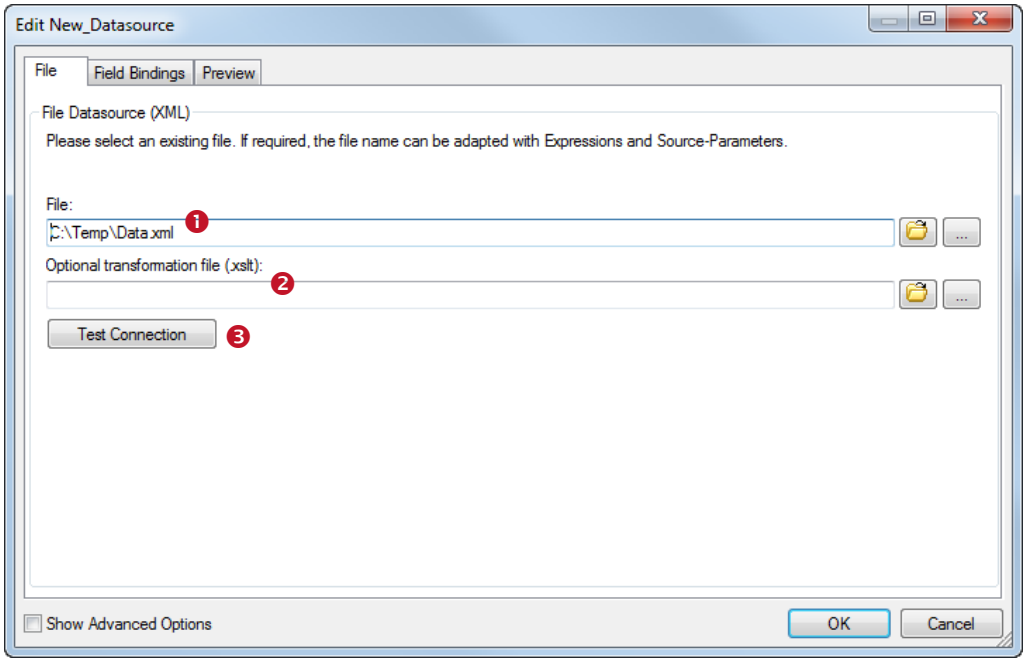


Figure 54: XML Datasource

- *File* ❶
  Select the XML file which you want to use as datasource. You can select a file by pressing the button 📁 or you can enter the file path manually by clicking the button ….
- *Optional transformation file (.xslt)* ❷
  Select an optional transformation file which should be applied to the XML file. Select the XSLT file via 📁 or enter the file path manually by clicking the button ….
- *Test Connection* ❸
  Use this button to check if the XML file exists and if the transformation was successful.

> ► For advanced users: With the help of expressions and source-parameters you have the possibility to build a dynamic file path and/or file name for the XML file and for the XSLT file. First open the edit dialog for ❶ or ❷ by clicking the … button, then use the corresponding button *Insert Source-Parameter…* or *Insert Expression…*.. For details on expressions and source-parameters see chapter 11 and section 13.5.2.

13.3.5.2 Field Bindings

To complete the setup of the datasource adjust the required field bindings (see section 13.4).

13.4 Field Bindings

Whenever you edit an external datasource it is essential to specify the field bindings. In this step you associate the required sourcefields of the external datasource with datafields. Only datafields can be used as placeholders within text boxes, barcodes, images etc. Sourcefields are not directly available in the layout.

> ► All sourcefields that you want to use in the layout must be bound to datafields.

To edit the field bindings switch to the *Field Bindings* tab in the *"Edit Datasource"* dialog. There you see all available sourcefields and their datafield bindings. Directly after the creation of a new datasource no bindings are specified by default.

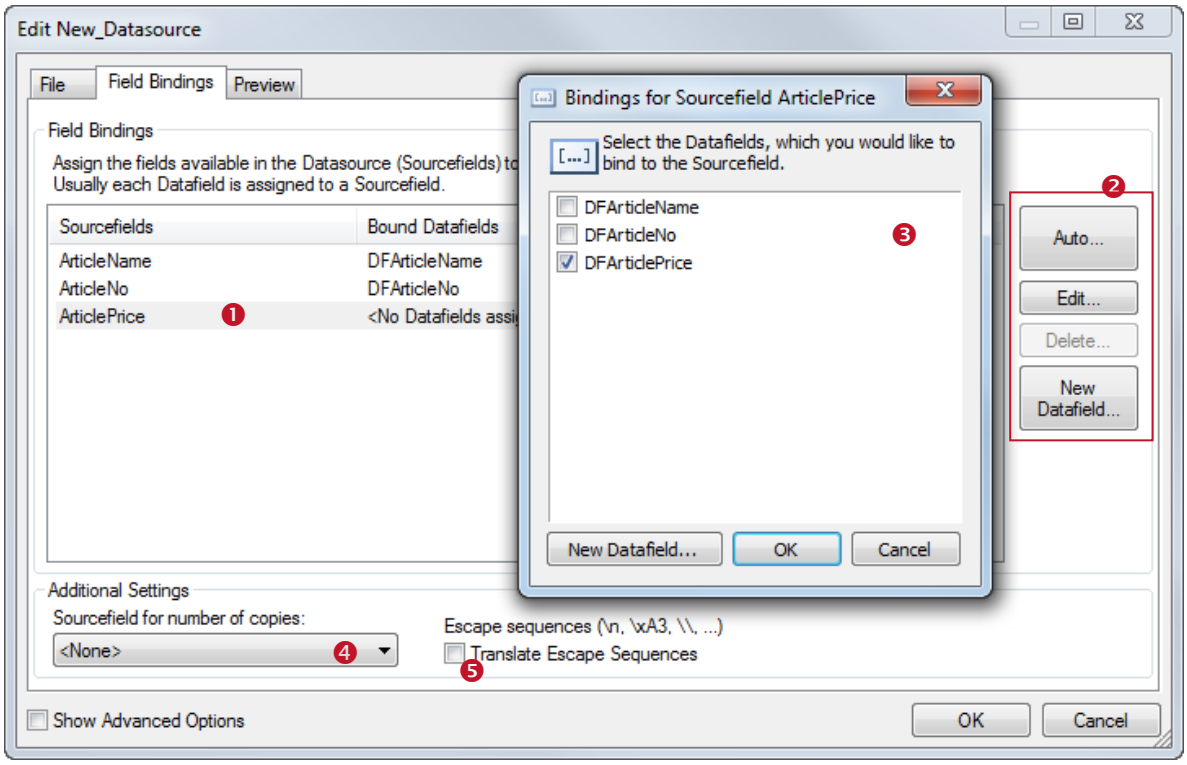


Figure 55: Field Bindings Settings

- *Sourcefields* ❶
  This list shows all available sourcefields and their assigned datafields. One sourcefield can be associated with one or more datafields. Using the buttons in ❷ the field bindings can be edited or deleted:
  - *Edit*
    This button opens dialog ❸. In this dialog select one or more datafields, which you want to bind to the selected sourcefield.
  - *Delete*
    This button deletes the bindings for the selected sourcefield. Alternatively press the *Del* key on the keyboard.
  - *Auto*
    This button is used to create field bindings automatically. It binds all sourcefields to existing, equally named datafields. If no appropriate datafield exists, **TFORMer** prompts you, if you want to create a new datafield with a suitable name.
  - *New Datafield*
    This button creates a new datafield. A newly created datafield is automatically bound to the currently selected sourcefield.

- *Sourcefield for number of copies* ❹
  The drop-down list ❹ specifies which sourcefield is used as *"Copies"* column. This column specifies how often each record is printed. If no sourcefield is specified, **TFORMer** prints each record once.
- *Escape sequences* ❺
  This option tells **TFORMer** whether escape sequences should be translated or not. An escape sequence is an in-text placeholder for special characters. It always starts with a backslash ("\") followed by a character or character code.
  Example: The escape sequence "\n" is a placeholder for a newline character.

► Please note: If escape sequences are activated you must use the sequence "\\" in the data-source to encode a single backslash "\"!

13.5 Advanced Options

When enabling the *Show Advanced Options* checkbox at bottom of the *"Edit Datasource"* dialog the following additional tabs will be shown:

- Computed Fields (see section 13.5.1)
- Source-Parameters (see section 13.5.2)

13.5.1 Computed Fields

A *computed field* adds an additional column to a datasource. The content of this column is either specified as a constant value or it is computed using an arbitrary expression. In the expression you may refer to other sourcefields and computed fields as basis for calculation.

The usual purpose of a computed field is to compute values which are based on sourcefields or other computed fields. For example, you might convert the content of an existing sourcefield to uppercase, or remove leading and trailing spaces. You can also concatenate multiple sourcefields into a single sourcefield. Or you can perform numerical computations (e.g., add the Value Added Tax, VAT). For examples, see section 13.5.1.1.1.

In addition, computed fields support aggregation functions. Thus it is pretty simple to calculate running sums and averages which are not directly available as fields in the datasource. However, please note: If the aggregation value is not necessarily required as sourcefield, you may also calculate it via a computed datafield in the layout (see section 10.3.2).

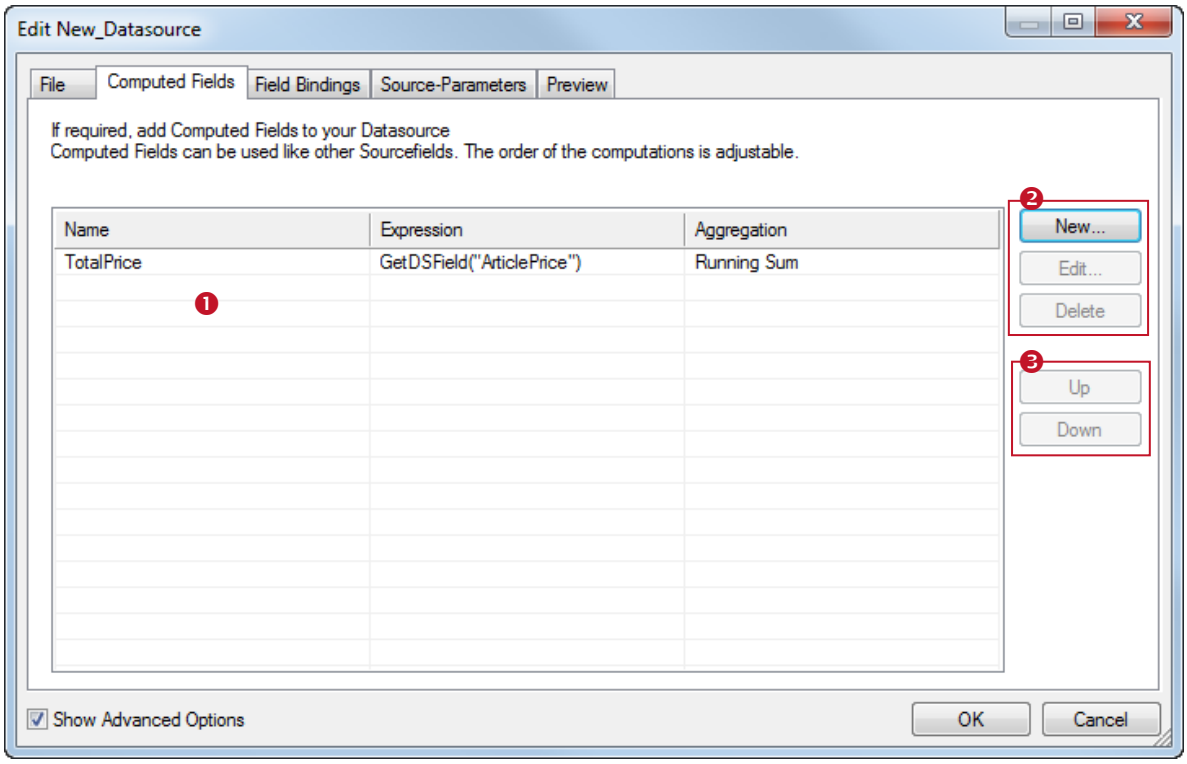


Figure 56: Computed Fields

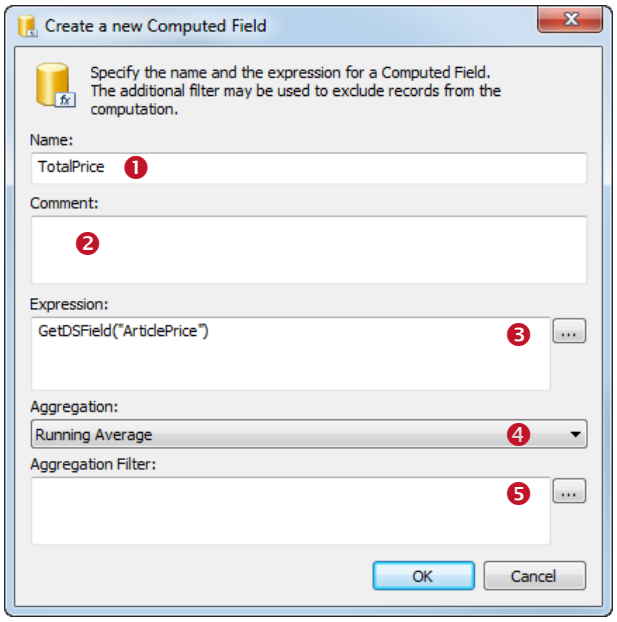
- *Computed Fields* ❶
  The list of computed fields is empty by default. Use the buttons in ❷ to create, edit and delete computed fields:
  - *New…*
    Create a new computed field (see next section).
  - *Edit…*
    Edit the computed field which is selected in ❶.
  - *Delete*
    Delete the computed field which is selected in ❶
- *Computation Order* ❸
  TFORMer computes the fields from top to bottom (as listed in ❶). To change this computation order use the buttons in ❸. Setting the correct computation order is essential if computed fields depend on each other.
  - *Up*
    Move the selected item up one position.
  - *Down*
    Move the selected item down one position.

► After a computed field was created, it is treated exactly like every other sourcefield. It has to be bound to a datafield before it can be used in the layout (see section 13.4).

► Computed fields are not available in the manual datasource.

13.5.1.1 Create a Computed Field

When clicking the *New…* button the following dialog will appear:



In ❶ you specify a name for the computed field. The name is used to identify the field. It has to be unique within the datasource.

In ❷ an optional comment can be entered.

The expression in ❸ provides the values for the computed field. You can enter the expression directly or you can open the expression builder by pressing button ... . For details see section 13.5.1.1.1.

The aggregation function ❹ allows you to build running sums and running averages. For details see section 13.5.1.1.2.

The filter expression ❺ can be used to exclude records from the computation. For details see section 13.5.1.1.3.

13.5.1.1.1 Expression

The expression in ❸ specifies the content of the computed field. This expression may return a constant value (e.g., to simulate a sourcefield which is not available in the current datasource), or it can perform arbitrary computations.

The expression can be entered directly in ❸, or you can open the expression builder by pressing the ... button. For more information about the expression builder, please refer to section 11.2.

Common applications are:

- Modify a sourcefield (convert it to uppercase, remove leading and trailing spaces, …).
  Example: *Trim (GetDSField("ArticleName"))*
- Concatenate multiple sourcefields into a single sourcefield.
  Example: *"Group:" + GetDSField("ProductGroup") + "Desc: " + GetDSField("Description")*
- Perform computations based on a sourcefield (like evaluating the Value Added Tax, VAT).
  Example: *GetDSField("UnitPrice") * 0.2*

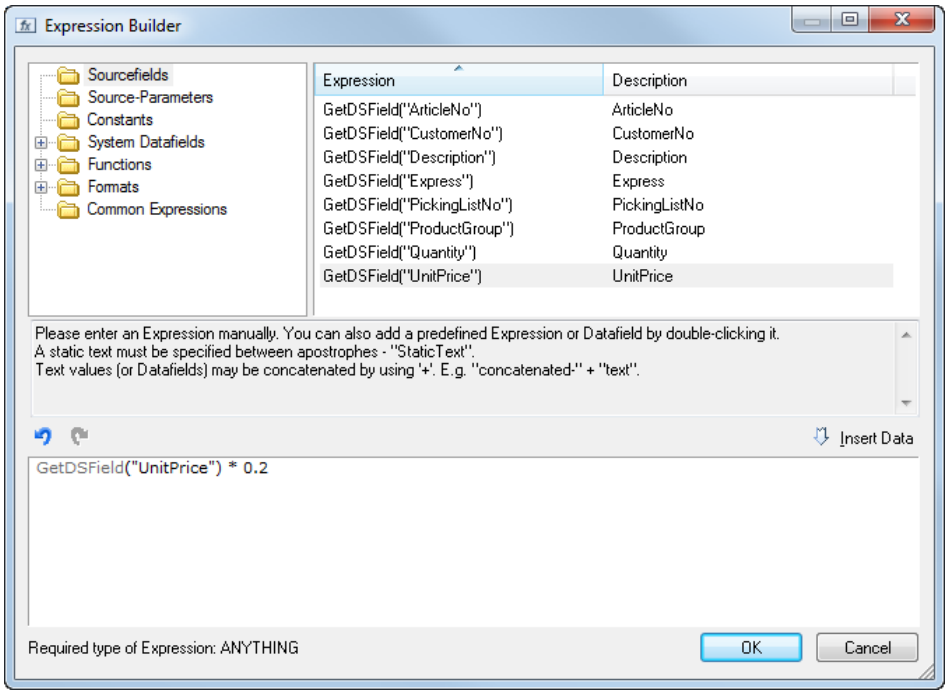


Figure 57: Example of an expression for a computed field

- To access a sourcefield from within an expression use the function *GetDSField("Sourcefield-Name")*.
- Accessing datafields is not possible.

13.5.1.1.2 Aggregation

For more advanced applications (like computing totals for the whole report) **TFORMer** supports aggregation functions. The following aggregation methods are available. You can choose from the drop-down list ❹.

▪ None (default)	No aggregation function is used.
▪ Running Average	**TFORMer** computes the mean value of all expression results up to the current record. For the first record the running average is the value itself. For the fifth record the running average is the average of the first five records. Example: If you want to provide the running average for the sourcefield "ArticlePrice", use Expression: *GetDSField("ArticlePrice")* Aggregation: *Running Average*
▪ Running Sum	**TFORMer** computes the sum of all expression results up to the current record. For the first record the running sum is the value itself. For all subsequent records the values are added. Example: If you want to provide a serial number which is incremented by "1" for each record, use Expression: *1* Aggregation: *Running Sum*

13.5.1.1.3 Aggregation Filter

When using a filter in field ❺, single values will only be considered if the expression returns true. Thus you can define, which values are considered for the aggregation, and which not.

Example:

```
GetDSField("ArticlePrice") > 100
```

This filter instructs **TFORMer** to consider only records where the "ArticlePrice" is greater than 100.

13.5.2 Source-Parameters

Source-parameters provide a possibility to implement dynamic datasource definitions. Thus you do not have to change the datasource definition each time for fetching different data. Source-parameters can be used to parameterize ODBC datasources and file-based datasources. Besides, they can be used in computed fields.

One good example for the use of source-parameters is to insert them in the SQL SELECT statement of an ODBC datasource. Thus the user (or developer) has the possibility to instruct **TFORMer** to fetch only specific records or to change the sorting order per parameter.

Using source-parameters requires the following steps:

- Create the source-parameter (see section 13.5.2.1)
- Assign the source-parameter (see section 13.5.2.2)
- Set a value for the source-parameter (see section 13.5.2.3)

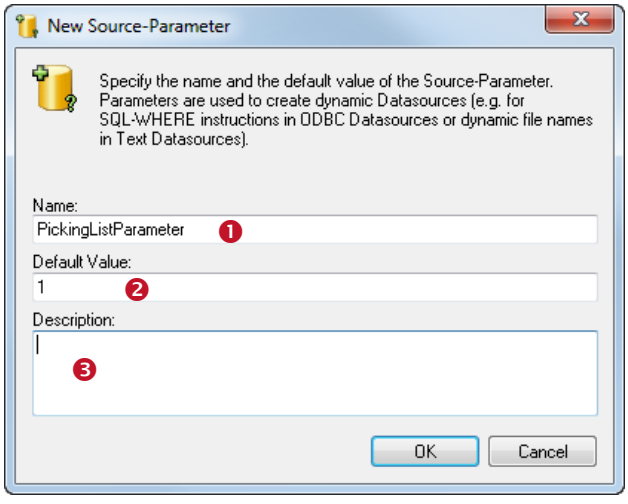
In the following sections we demonstrate the use of source-parameters by means of the picking list example as included in the **TFORMer** setup. Please note: This sample already includes all the adjustments as described below! To open the sample select *File* ► *New Form…* from the menu. Then open the folder *"(6) Samples"* and select *"Picking List"*.

13.5.2.1 Create a Source-Parameter

To create a new source-parameter use one of the following methods:

Select the datasource in the design tree and click on the *"New Source-Parameter"* icon in the data toolbar or select *Data* ► *New Source-Parameter…* from the menu. Alternatively right-click on datasource in the design tree and select *New Source-Parameter…* from the pop-up menu. A third method is to open the *"Edit Datasource"* dialog and then click on the *New…* button in the source-parameter tab.

The following dialog will appear:



In ❶ enter the name for the source parameter.

The *Default Value* ❷ is used to initialize the source-parameter after the layout was loaded.

In ❸ you may enter an optional description.

13.5.2.2 Assign the Source-Parameter

Now that you have created a source-parameter it can be used

- as placeholder in an ODBC datasource (in the DSN, username, password and SQL SELECT statement),
- as placeholder in a file-based datasource (in the file name and file path),
- in computed fields.

In this example we use the source-parameter to parameterize the *SELECT Query* in an ODBC datasource. Assuming the ODBC datasource is already created (see also sections 13.3.2 and 13.3.3), open the *"Edit Datasource"* dialog (e.g., by double clicking the datasource in the design tree) and switch to the *SQL-Query* tab.

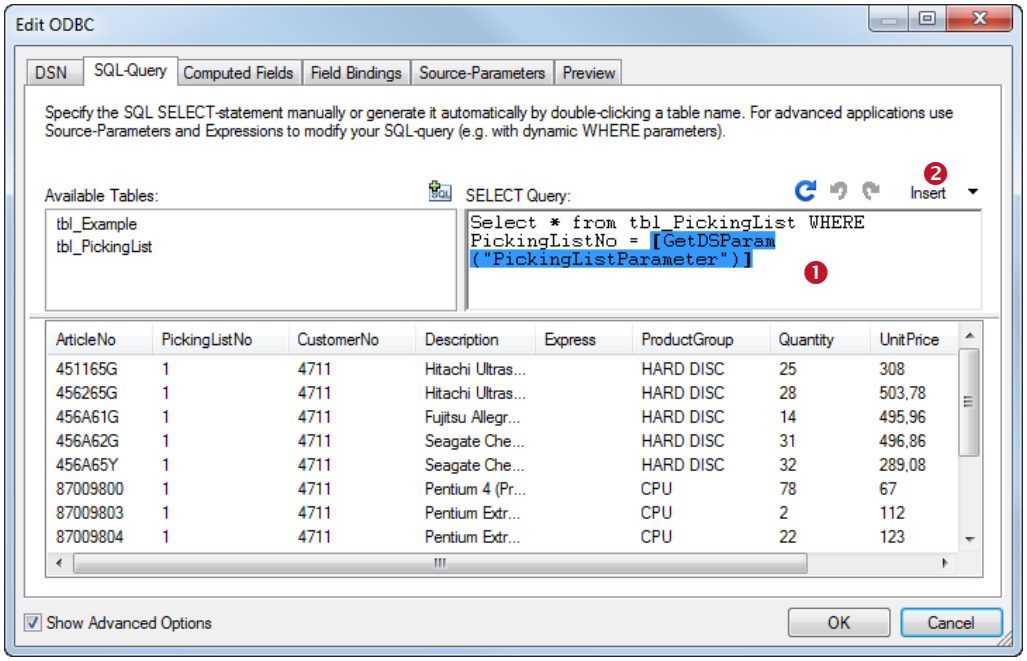


Figure 58: SQL-Query tab

To insert a source-parameter in the SQL query place the cursor on the required position in ❶ and then click on ❷ *Insert* ► *Source-Parameter…*. The following dialog will appear:

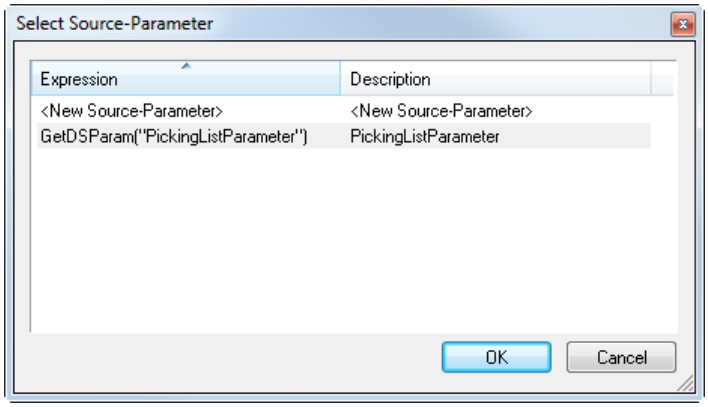


Figure 59: Select a Source-Parameter

Select the required source-parameter and confirm with *OK.*

In this case the source-parameter "PickingListParameter" will be used in the WHERE clause of the SQL query to retrieve the data for a specific picking list only. When **TFORMer** is fetching data from the datasource the expression *[GetDSParam("PickingListParameter")]* is substituted with the actual value of the source-parameter. Thus, the resulting SQL SELECT fetches only records from the database which match the specified picking list number.

E.g., if the "PickingListParameter" is set to 1, the SQL Query

```
SELECT * FROM tbl_PickingList WHERE PickingListNo = [GetDSParam("PickingListParameter")]
```

will internally be evaluated as:

```
SELECT * FROM tbl_PickingList WHERE PickingListNo = 1
```

13.5.2.3 Set a Value for the Source-Parameter

For correct substitution of the source-parameters you have to assign the required parameter values.

Therefore click on the *"Edit Source-Parameter"* icon in the toolbar or select *Data ► Edit Source-Parameter…* from the menu. Alternatively you can also click on the button *Parameter…* in the design view or right-click on the datasource entry in the design tree and select *Edit Source-Parameter…* from the pop-up menu. The following dialog will appear:

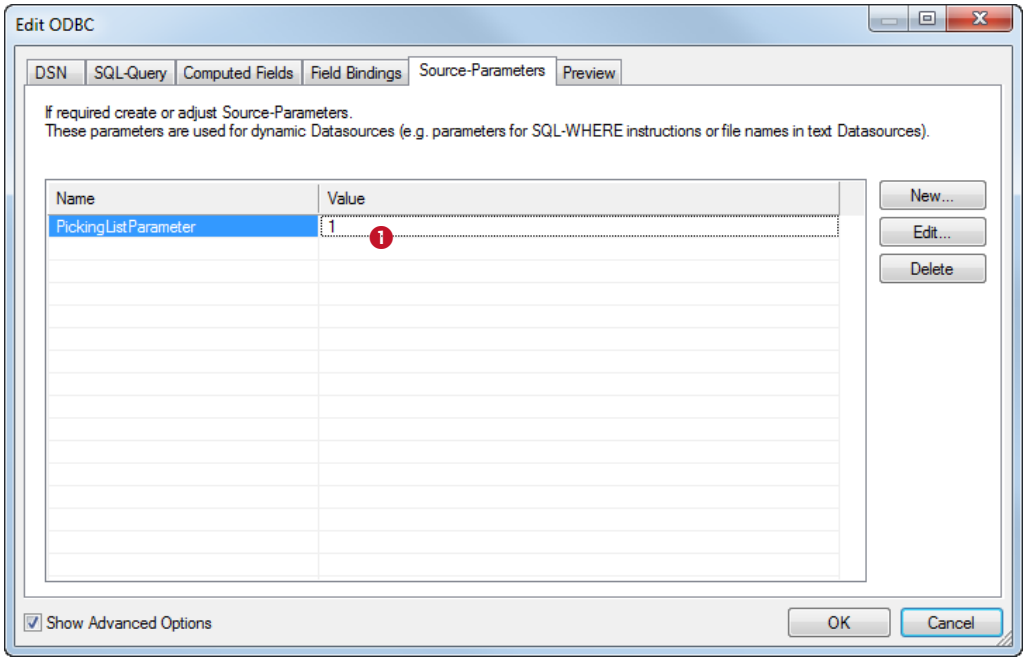


Figure 60: Edit a Source-Parameter Value

Enter the required value for the source-parameter in ❶. Then confirm with *OK*.

Using the settings as shown above, **TFORMer** will only fetch records where the *PickingListNo* is equal to *'1'*.

#	Copies	PickingListNo	ArticleNo	CustomerNo	Description
1	1	1	451165G	4711	Hitachi Ultrast...
2	1	1	456265G	4711	Hitachi Ultrast...
3	1	1	456A61G	4711	Fujitsu Allegro...
4	1	1	456A62G	4711	Seagate Cheet...
5	1	1	456A65Y	4711	Seagate Cheet...
6	1	1	87009800	4711	Pentium 4 (Pr...
7	1	1	87009803	4711	Pentium Extre...

Figure 61: Data for PickingListNo = 1

Changing the source-parameter to *'2'* results in different data:

#	Copies	PickingListNo	ArticleNo	CustomerNo	Description
1	1	2	456365G	4711	Hitachi Ultrast...
2	1	2	456A64G	4711	Fujitsu Allegro...
3	1	2	456A65G	4711	Seagate Cheet...
4	1	2	456A65S	4711	Fujitsu Allegro...
5	1	2	456A66G	4711	Fujitsu Allegro...
6	1	2	87009801	4711	Pentium 4 Extr...
7	1	2	87009806	4711	Pentium D (Pr...

Figure 62: Data for PickingListNo = 2

► Source-parameters can also be used by software developers via the API or with the command line utility **TFPrint**. Please refer to the appropriate documentation for details.

13.6 Filter

The filter is used to limit the records in the active datasource. The filter criterion is specified with a Boolean expression (see also chapter 11): Only records for which the filter expression returns *true* remain visible in the data view. All other records are masked out. Masked out records are neither used for the preview nor for printing.

The filter functions are available in the data view and in the preview.

This section covers the following topics:

- Create a filter (see section 13.6.1)
- Apply a filter (see section 13.6.2)
- Print with a filter (see section 13.6.3)

13.6.1 Create a Filter

To create a filter select *Data* ► *Edit Filter…* from the menu or click on the icon in the data toolbar. Alternatively use the button *Filter…* in the data view.

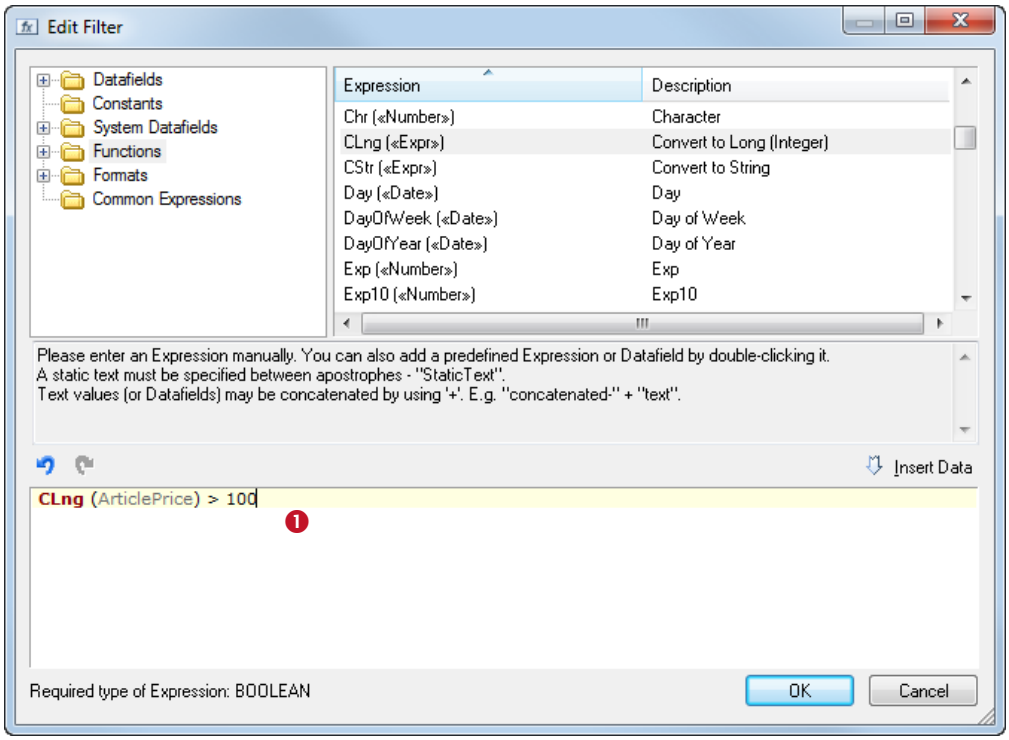


Figure 63: Edit a Filter

In the appearing dialog enter the required expression in ❶. Then confirm with *OK.* The filter will be applied to the current datasource automatically. An active filter can be identified by the pressed filter icon .

Example:
When using the filter expression below, only records with an *ArticlePrice* greater than 100 will be printed.

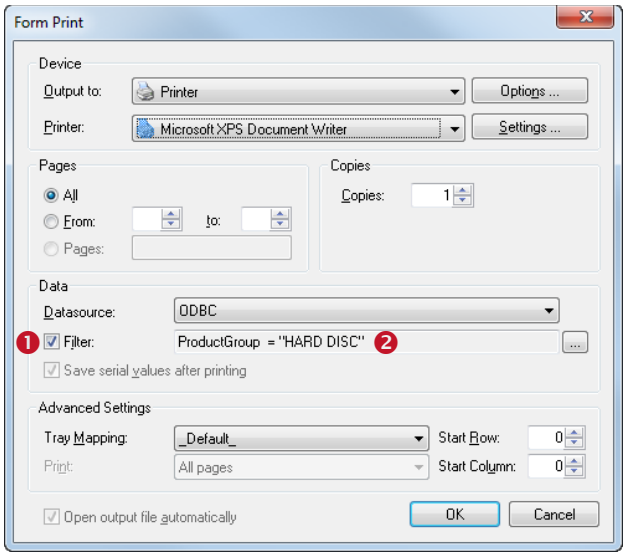
```
CLng(ArticlePrice) > 100
```

13.6.2 Apply a Filter

To apply or to cancel a previously defined filter click on the icon in the data toolbar or select *Data ► Apply Filter* from the menu.

13.6.3 Print with a Filter

To start the printing process press *Ctrl+P* or select *File ► Print…* from the menu. The dialog below will appear. On the printing dialog select the required printer, the datasource, etc.



If you have already adjusted the filter in the data view or in the print preview, no additional adjustments are required in ❷. Anyway, you still have the possibility to

- enable/disable (❶) or to
- modify (❷)

the current filter expression. To edit the filter expression in ❷ click on the button ... . The expression builder (see section 11.2) will open.

Upon confirming the printing dialog with *OK* the printing process will start.

For general information on the printing dialog, please refer to chapter 15.

Example:

For the following screenshot we use the picking list example *(File ► New Form… ► (6) Samples ► Picking List).* To print only items in the product group “HARD DISC” use the following filter expression:

```
ProductGroup = "HARD DISC"
```

TFORMer will only print data records where the content of the field *ProductGroup* is equal to the string “HARD DISC”:

Customer 4711

Picking List

TEC-IT
WWW.TEC-IT.COM

Picking List Number: 1

Article Group: HARD DISC

Article-No	Description	Quantity	Unit Price	Barcode
451165G	Hitachi Ultrastar 15K147 147GB SAS (HUS151414VLS300)	25	308,00	
456265G	Hitachi Ultrastar 15K300 300GB SAS (HUS153030VLS300)	28	503,78	
456A61G	Fujitsu Allegro 10LX, 300GB SAS (MBA3300RC)	14	495,96	
456A62G	Seagate Cheetah 15K.5 300GB SAS (ST3300655SS)	31	496,86	
456A65Y	Seagate Cheetah 15K.4 147GB SAS (ST3146854SS)	32	289,08	
Group Total		**130**	**53402.50**	

Figure 64: Picking List, filtered for ProductGroup = ”HARD DISC”

14 Preview

14.1 Introduction

TFORMer offers a *preview* which instantly renders the resulting output using the current layout and the active datasource. This view offers functions for page navigation, filtering the data, and more.

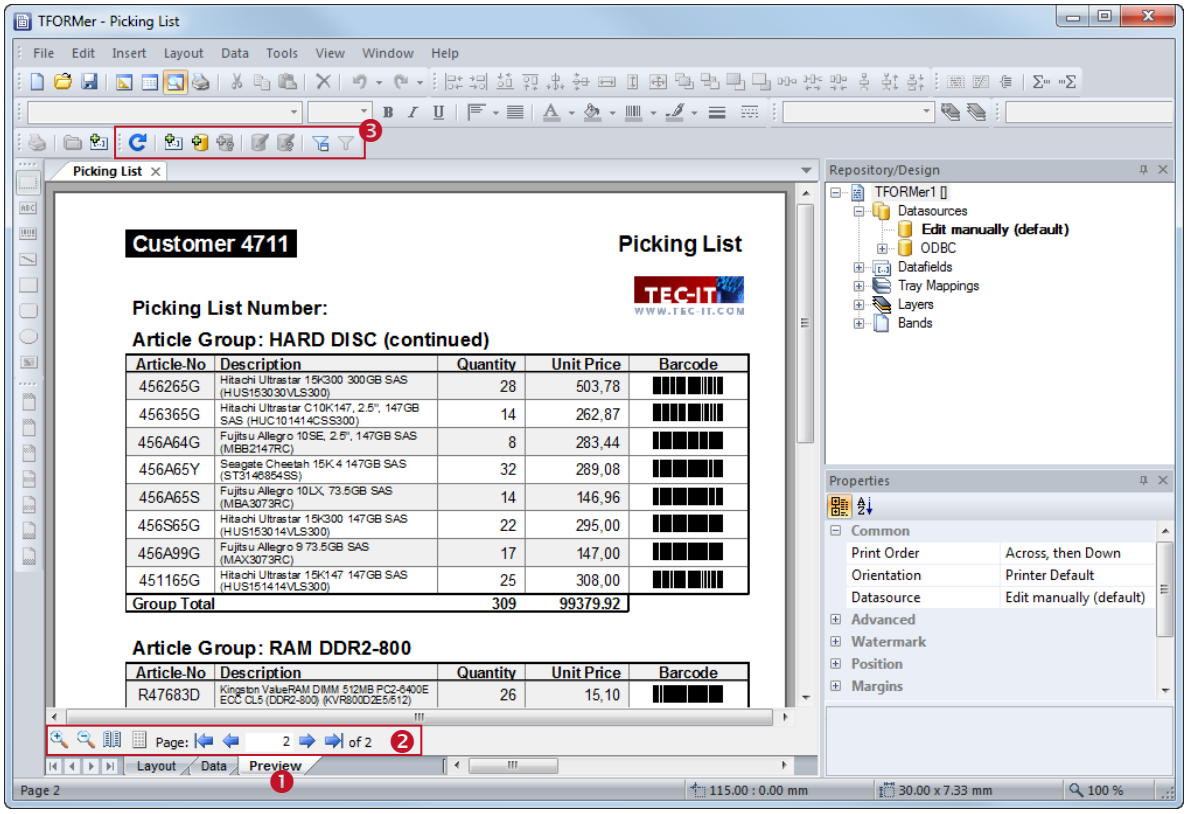


Figure 65: Preview

To switch to the preview click on the *Preview* tab ❶. Alternatively press *Ctrl+Space* or select *File* ► *Preview* respectively *View* ► *Preview* from the menu.

14.2 Page Navigation

The page navigation ❷ allows you to zoom, to switch between single page or double-page preview, to show/hide label boundaries, to navigate between pages and to directly jump to a user-defined page.

14.3 Additional Functions

Most functions, which are available in the data view (apply a filter, change a source-parameter, switch the datasource) can also be applied in the preview (see ❸). The result is displayed immediately.

► Please note: When applying a filter (see section 13.6) or when switching between datasources (see section 13.3.2.6), **TFORMer** reloads the data. This ensures that the most actual data is used.

15 Printing

15.1 Introduction

The term printing in the context of this document incorporates generating output in general.

Besides direct printing (via a printer-driver in Microsoft Windows), **TFORMer** also supports the creation of PDF documents, HTML files, PostScript files, images, ZPL-II output (for ZEBRA printers) and ASCII output.

All output formats are generated directly. There is no need for third party software. This might particularly be interesting for PostScript or ZEBRA output (see below).

15.2 Printing Manually

To open the print dialog press *Ctrl+P* or select *File* ► *Print…* from the menu. The following dialog will appear:

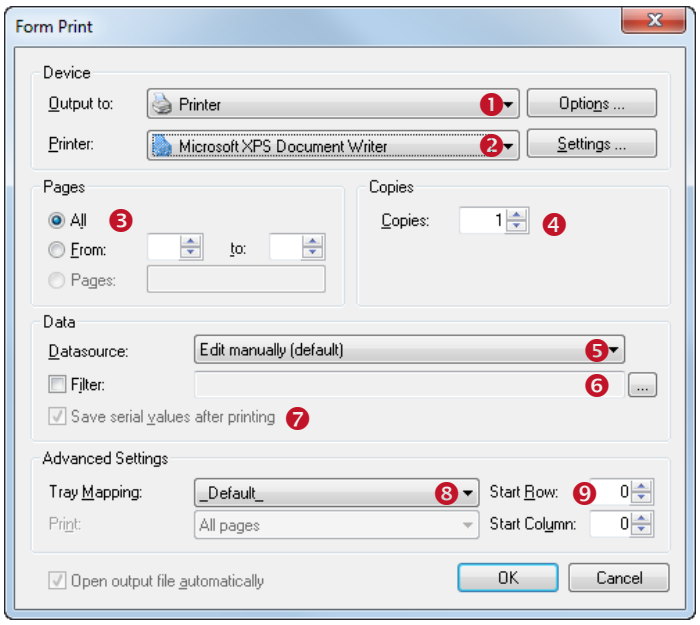


Figure 66: Print Dialog

In this dialog specify all output parameters as discussed below. When finished click *OK* to start the print-job (or to create the output file).

15.2.1 Output Format and Device

In ❶ select the output device respectively format:

- Printer (via a printer driver)
- PDF
- PostScript
- HTML
- Text (pure ASCII)
- BMP, GIF, JPG, PCX, PNG, TGA, TIF or multipage TIF
- ZEBRA (ZPL-II)

Depending on your selection in ❶ different target devices are available in ❷.

15.2.1.1 Printing to PostScript and ZEBRA printers

TFORMer generates PostScript or ZEBRA output directly. This means that you can use such printers without any proprietary printer driver – you have two options:

- Print via a generic ASCII printer driver (dumb driver)
- Print via a PostScript or ZEBRA driver

In both cases the printer driver is used to send the data which was generated by **TFORMer** to the required device (comparable to pass through mode). No driver functionality is used.

15.2.2 Pages and Copies

In ❸ you specify the range of output pages (e.g., from “1” to “1” prints the first page only). In ❹ you specify the number of copies per page.

15.2.3 Data

In ❺ select the datasource to be used for printing.

> ► Please note: Before printing **TFORMer** always reloads the data from the adjusted data-source. This guarantees that the actual data is used.

In ❻ an additional filter can be applied. If a filter is set, only the records for which the filter expression returns *true* are printed. All other records are ignored. In order to apply the filter the checkbox “Filter” must be activated.

Examples for filter expressions:

Filter Expression	Description
`Record <= 3`	Print the first three data records only.
`ProductGroup="HARD DISC"`	Print only data records, where the value of the datafield “ProductGroup” is “HARD DISC”.

An example for printing with a filter is described in section 13.6.3.

If you are generating serial numbers with **TFORMer** (see section 10.3.3), you can specify whether the last printed value of the serial number should be stored or not (see ❼). If enabled, the next print job will continue with the next unused number as start value.

15.2.4 Advanced Settings

15.2.4.1 Tray Mappings

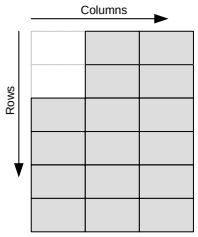
In ❽ select one of the defined tray mappings (see also section 12.6). If you want to print on the default printer tray, this selection is obsolete.

15.2.4.2 Start Row and Start Column

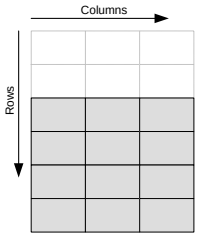
For label printing the *“Start Row”* and the *“Start Column“* (❾) can be specified. Use this setting if you want to leave the first labels on the sheet blank.

Example:

Start Row=2
Start Column=0



Print order *“Down, then Across:.* The first two labels in the left column will be left empty.



Print order *“Across, then Down”:* The first two rows will be left empty.

15.3 Printing Programmatically

Software developers can print layouts created with **TFORMer Designer** as part of their applications using **TFORMer SDK.**

TFORMer SDK is available for Microsoft® Windows® and for almost all Linux® and UNIX® platforms. The following kinds of integration are possible:

- Command line based printing application (TFPrint)
- Software component (DLL, COM component, .NET component and shared library)

For details, please refer to the Developer Manual or to the API References of the **TFORMer SDK.**

16 Repositories

16.1 Introduction

TFORMer offers two different possibilities for organizing layouts:

16.1.1 Stand-Alone Forms

> ► Stand-alone forms are the simplest way to create and to print layouts. Each stand-alone form designed with **TFORMer** can be used on its own.

A stand-alone form contains all the necessary information which is required for printing. Images are not embedded but stored as file references. As long as no repository is used, **TFORMer** always creates stand-alone forms when selecting *File* ► *New Form…* from the menu.

A stand-alone form is using the file extension **.tff*. Organizing multiple stand-alone forms in the file system is completely up to the user.

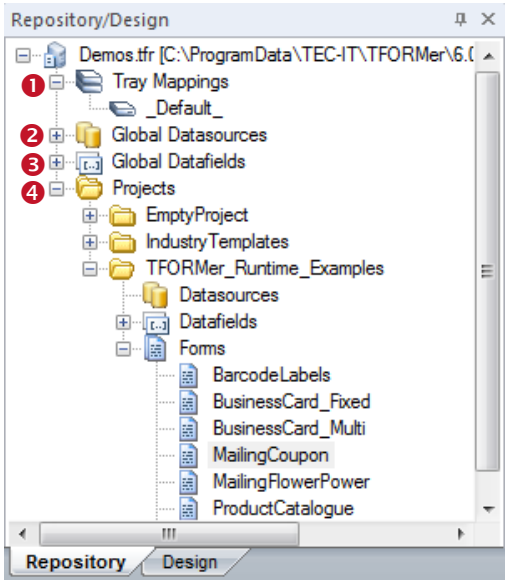
16.1.2 Repositories

If you prefer to organize your layouts and datafields in a structured way or if you plan to create multiple layouts which share the same data basis (same datafields) the use of a so-called *repository* is recommended. A repository is a central database for layouts, datasources, datafield definitions and tray mappings.

Layouts and datafield definitions are stored within a repository on a per *"Project"* base. A project defines datafields and holds (better: references) layouts. Each of the datafields defined in a project is accessible within every layout contained in that project. A repository can contain multiple projects and one special *global project*. Datasources and datafields which are defined within the global project (*global datasources and datafields*) may be used by all layouts in all projects.

> ► A repository is used for organizing multiple layouts within a structured data base. A repository holds *tray mappings* and is divided into *projects*. A project stores *datasources, datafield* definitions and references *layouts.*

A repository is stored with the file extension *.tfr*. All projects are stored in subdirectories. Please, take care to maintain this directory structure. Do not change it manually!



In a repository you will find the following entries:

1. *Tray Mappings*
   These tray mappings are available for all layouts in all projects.
2. *Global Datasources*
   These datasoruces are available for all layouts in all projects.
3. *Global Datafields*
   These datafields are available for all layouts in all projects.
4. *Projects*
   A collection of layouts and datafields. Datafield definitions are only valid within layouts of the project they are defined in.

16.2 Basic Operations

16.2.1 Open an existing Repository

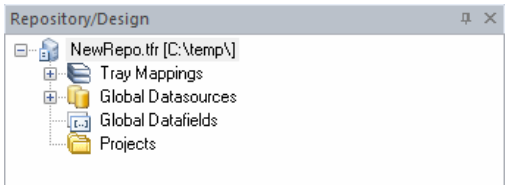
To open a repository select *File ► Repository ► Open Repository…* from the menu. Then select the repository file and confirm with *OK.* Alternatively use the menu *File ► Open…*. **TFORMer** allows you to open both, .tff-files (stand-alone forms) and .tfr-files (repositories). Once opened, the repository is displayed in the design tree in the *"Repository"* tab – see figure above.

16.2.1.1 Demo Repository

The **TFORMer** installation includes a demo repository. This repository is named "Demos.tfr" and can be opened by selecting *All Programs ► TEC-IT TFORMer 6.0 ► Examples ► Demo Repository* from the *Windows Start Menu.*

16.2.2 Create a New Repository

Select *File ► Repository ► New Repository…* from the menu. A file dialog will appear. Enter a suitable filename. Then click *Save.*



The new repository will be displayed in the design tree.

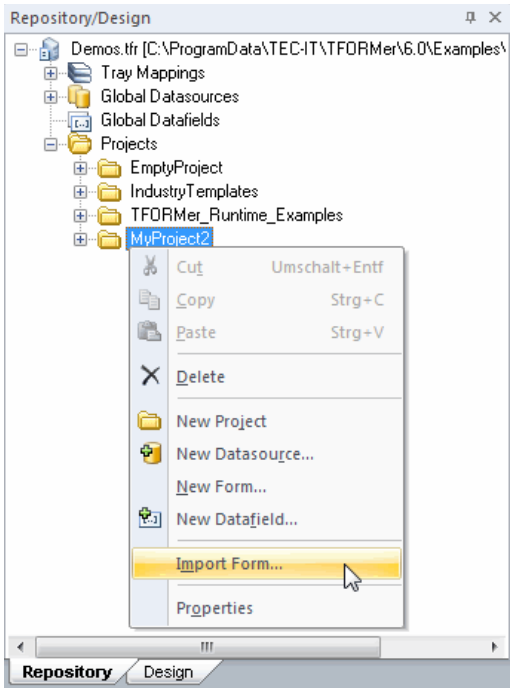
► Important: Each repository has to be stored in a separate folder! Otherwise it will lead to problems. So, when creating a new repository, you should always create a folder first, then save the repository there.

16.2.3 Save a Repository

Select the repository inside the design tree and press *Ctrl+S* (or select *File ► Save* from the menu).

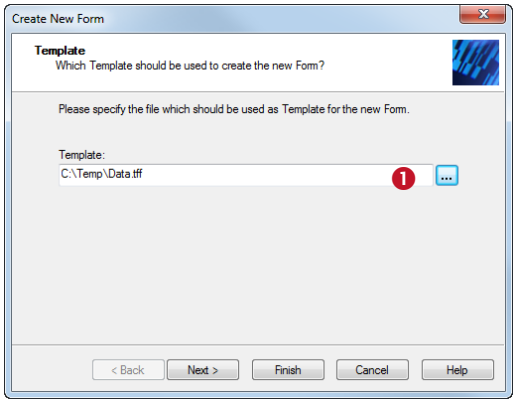
16.2.4 Import a Stand-alone Layout into a Repository

You can import existing stand-alone layouts into the repository by following these steps:



In the design tree locate the project, in which you want to import the layout.
Right-click on the project and select *Import Form…* from the pop-up menu.

The dialog below will appear.



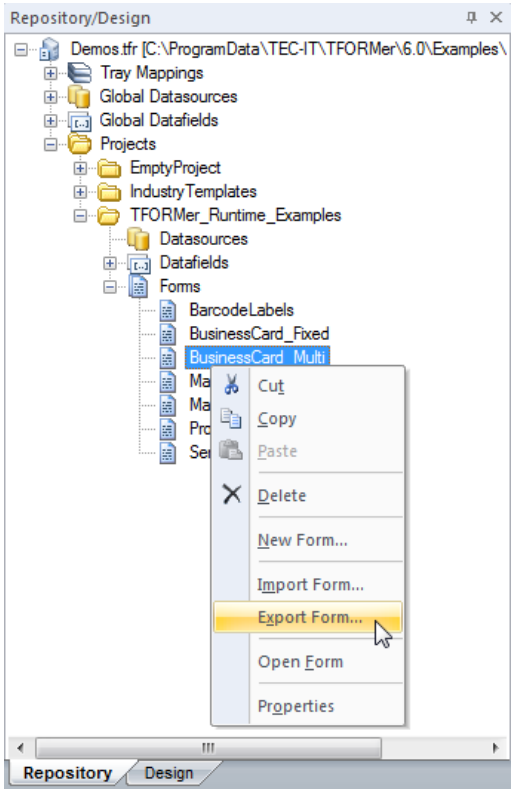
In ❶ select the file that you want to import.

Then click *Next* or *Finish.*

If you click *Next*, you can assign a name and a description for the layout in the repository.
By default the name of the imported file will be used as layout name.

16.2.5 Export a Layout from the Repository

Exporting a layout from a repository creates a stand-alone layout.



In the design tree locate the project, from which you want to export the layout. Then right-click on the layout and select *Export Form…* from the pop-up menu.

An alternative method is to open the layout by double-clicking it in the design tree. Thereafter select *File ► Save as…*.

16.2.6 Close a Repository

Mark the repository in the design tree (the root entry). Then right-click on it and select *Close Repository* from the pop-up menu. Alternatively you can also use the menu *File ► Repository ► Close Repository.*

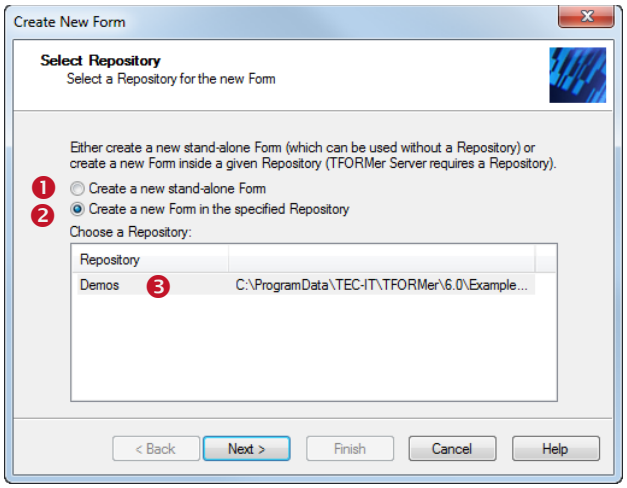
16.3 Working with a Repository

16.3.1 Projects

Before you can add layouts and datafield definitions, you first have to create a project: Right-click on the repository in the design tree. Then select *New Project* from the pop-up window. Alternatively use the menu: *Insert ► Project.* The name and the description for the project can be edited in the properties window or using the *F2* key.

16.3.2 Insert a Layout

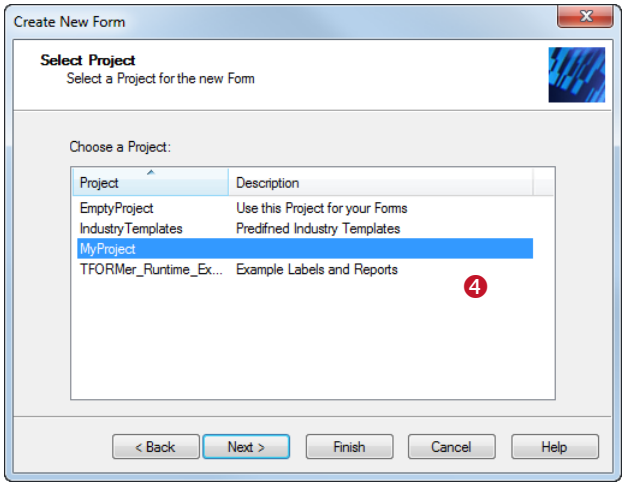
Select *File ► New…* from the menu (or right-click on any item in the *"Layouts"* branch in the design tree and select *New Form…* from the pop-up menu) to open the following wizard:



In this dialog you select, whether you want to create a stand-alone form (❶), or if you want to add a form to a repository (❷).

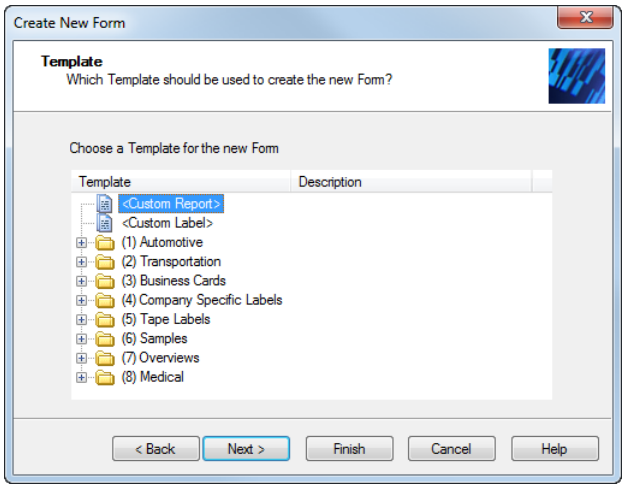
To insert the new layout into the repository select option ❷. Then select the repository in ❸.

Click *Next.*
The following dialog will appear:



Select the project in ❹.

Then click *Next.*



In this dialog you can finally select one of the available templates (analogous to section 6.2.2).

Click *Finish* to accept.

The selected layout template will be added to the selected project in the repository.

If you click *Next*, you can assign a name and a description for the layout in the repository.
By default the name of the imported file will be used as layout name.

17 General Settings

17.1 Options Dialog

In the options dialog you can customize common **TFORMer** settings. To open the dialog select *Tools* ► *Options…* from the menu.

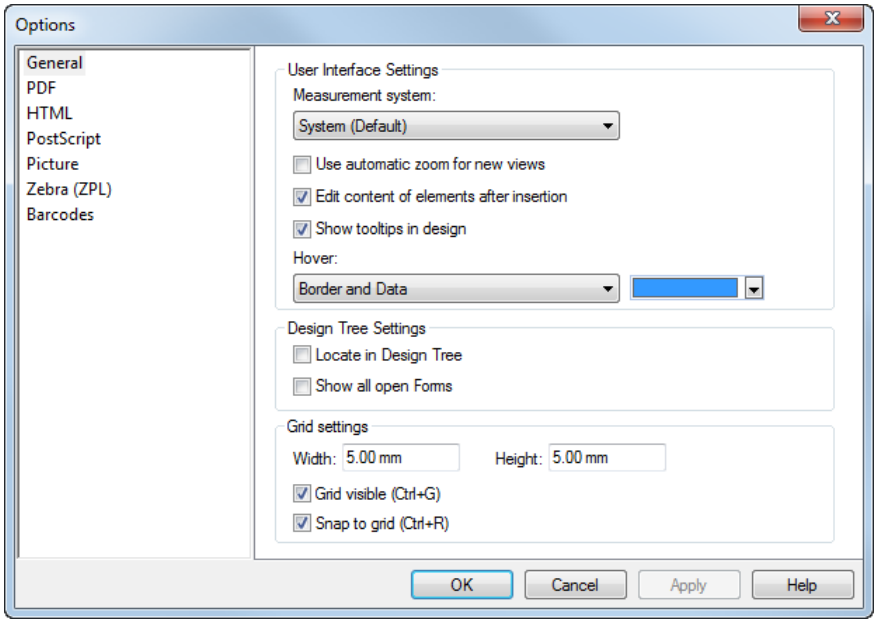


Figure 67: Options Dialog

Please note:

- **TFORMer** internally only stores the values of the "Gerenal" settings page (see section 17.2) permanently. All other settings (sections 17.3 to 17.8) are output parameters, which are shared by **TFORMer Designer** and other components like **TFPrint** and **TFORMer SDK.** Each time a new instance of **TFORMer Designer** is started they are reset to the values from the configuration file *"TFORMer.xml"* (see also Appendix E.2).
- In order to change the output parameters (sections 17.3 to 17.8) for single print-outs, use either this dialog or the *Options…* button in the print dialog.
- In order to change the output parameters (sections 17.3 to 17.8) permanently, please edit the configuration file *"TFORMer.xml"* in an appropriate text or XML editor.

17.2 General

17.2.1 User Interface Settings

Measurement system	The measurement system can be set to: ▪ System (Default) Uses the measurement system that is specified in the regional options of your operating system. ▪ Metric (mm) ▪ U.S.-System (inch)
Use automatic zoom for new views (y/n)	If enabled, all files that are opened will be set to *"Automatic"* zoom level (= fit the width of the layout window). By default this option is disabled.

Edit content of elements after insertion (y/n)	If enabled, upon insertion of a new text, barcode or picture element the dialog for editing the content will be opened automatically (see chapter 9). By default this option is enabled.
Show tooltips in design (y/n)	If enabled, **TFORMer** will display tooltips when hovering the mouse cursor over design elements or bands. These tooltips show information about assigned data, comments and printing conditions. By default this option is enabled.
Highlight on mouse over	Specifies the type of highlighting and the highlighting color for elements which have the mouse focus in the design view. The following options are available: ▪ Border ▪ Border and Data (Default)

17.2.2 Design Tree Settings

Locate in Design-Tree (y/n)	If enabled, **TFORMer** will automatically locate each element in the design tree which was selected it in the layout view. By default, this option is disabled.
Show all open Forms (y/n)	If enabled, the design tree shows the contents of all open documents. Otherwise it is populated with the contents of the current layout only. By default, this option is disabled.

17.2.3 Grid Settings

Width	Horizontal resolution of the grid.
Height	Vertical resolution of the grid.
Grid visible (y/n)	Show or hide the grid in the layout view. Keyboard shortcut: *Ctrl+G*
Snap to grid (y/n)	If enabled, every graphical element that is inserted or moved will be positioned on the grid. Keyboard shortcut: *Ctrl+R* Hint: If this option is enabled you may position elements off the grid as well: Just press the *Alt* key during a drag and drop operation.

17.3 PDF

17.3.1 Properties

Page Layout	The page size (Letter, A4, …) for PDF output.
Orientation	The orientation (portrait or landscape) for PDF output.
Color mode	The color mode for PDF output: “Color”, “Grayscale” or “Black & White”.

17.3.2 Fonts

Embed as ___ (y/n)	If this option is enabled, all used fonts will be embedded within the PDF file. This option ensures, that the PDF document is always displayed correctly, even if the required fonts are not installed on the system where the document is viewed. Embedding fronts increases the size of the generated PDF document. By default this option is enabled. You can choose the method for font-embedding: ▪ *TrueType:* The complete TrueType font is embedded in the PDF file. ▪ *TrueType-Subgroups:* Only the required parts of a TrueType font are embedded in the PDF file (smallest output size). ▪ *Type3:* Fonts are embedded as type 3 fonts in the PDF file. Type 3 fonts should only be used when printing the resulting PDF on high resolution devices.
Simulate bold fonts if not available (y/n)	If this option is enabled, **TFORMer** will simulate the bold font in the resulting PDF file if the bold typeface is missing. By default this option is enabled.

17.3.3 Image Resolution

Reduce to ___ (y/n)	If enabled, all images of which the resolution is larger than the specified value will be resampled. Smaller images are stored in their original resolution. If disabled, all images are embedded in their original size.

17.3.4 Image Compression

Method	The compression method that is used for storing images in the PDF output: ▪ Auto (All images which are provided as JPEG are embedded as JPEG. All other images (.bmp, .tif, …) are stored as Zip compressed bitmap data.) ▪ Zip (All images are stored as Zip compressed bitmap data.) ▪ Jpeg (All images are stored as JPEG images using the specified compression quality.)
Quality	The compression factor which is used for storing embedded JPEG images.

17.4 HTML

Page Layout	The page size (Letter, A4, …) for HTML output.
Orientation	The orientation (portrait or landscape) for HTML output.
Output Resolution	The output resolution (in dpi) for rendering HTML pages. The pre-set value is the screen resolution 96 dpi.
Optimal Barcode Resolution (y/n)	If enabled, the module width of barcodes will be optimized for the (rather small) output resolution of HTML pages. This option guarantees optimal barcode readability. Please note: As a result the size of the barcode may be reduced!

17.5 Postscript

Page Layout	The page size (Letter, A4, …) for PostScript output.
Orientation	The orientation (portrait or landscape) for Postscript output.
Color mode	The color mode for PostScript output: “Color”, “Grayscale” or “Black & White”.

17.6 Picture

Page Layout	The page size (Letter, A4, …) for image output.
Orientation	The orientation (portrait or landscape) for image output.
Output Resolution	The output resolution (in dpi) for rendering images. The pre-set value is the screen resolution 96 dpi.
Color Mode	The default color mode for image output: “Color”, “Grayscale” or one of various “Black & White” modes (e.g., Scattered Dither, Ordered Dither, Treshold).
Font Anti-aliasing (y/n)	If enabled, all fonts are anti-aliased. By default this option is enabled.
TIFF Compression	The compression method which is used for creating TIFF output.
JPEG Compression	The compression factor which is used for creating JPEG output.

17.7 Zebra (ZPL)

Page Layout	The page size (Letter, A4, …) and the orientation (portrait or landscape) for ZPL-II output.
Orientation	The orientation (portrait or landscape) for ZPL-II output.
Resolution	The resolution (in dpi) of the used printer. The pre-set value is “8 dots/millimeter (203 dpi)”.
Scaling (^JM)	Selects the scaling factor for the output. Switch to *Doubled* if the resulting output is half of the required size. ▪ Normal (default) ▪ Doubled Please check out your ZPL-II manual for the ^JMA and ^JMB commands for details.

Compression	Selects the compression method that is used for compressing the bitmap which is sent to the ZEBRA printer. The following options are available: ▪ None (The bitmap will not be compressed. This kind of transfer may be very slow but it is supported by all ZEBRA models.) ▪ RLE (The bitmap will be RLE compressed. This option is supported by most ZEBRA models. This is the default option.) ▪ PNG (The sent bitmap is a PNG image. This option is supported by newer ZEBRA models respectively newer ZEBRA firmware versions.)
Dither Mode	Selects the method used for simulating color output on a black & white printer. The following options are available: ▪ Scatter (grayscale simulation method 1, this is the default) ▪ Ordered (grayscale simulation method 2) ▪ Threshold (no grayscale pattern, just black or white)
Do not dither Text (y/n)	If enabled, colored text is never dithered. By default this option is enabled.

17.8 Barcodes

Barcode Printing on Windows-Printers	For the output on Windows-Printers **TFORMer** offers the following barcode rendering methods: ▪ Default method (This method is given through the barcode DLL. Currently the default method is the quality mode – please, see below. However, it may change in later versions.) ▪ Compatibility mode, supported by all printers (This method may result in suboptimal output quality, but it is compatible to nearly all printers. It uses the GDI rectangle functions for drawing the barcode symbols.) ▪ Quality mode, supported by most printers (This method results in the best quality but may not be supported by all printers. It uses advanced GDI drawing methods.) ▪ Dual, a combination of above modes (This method produces correct output on most printers but might decrease drawing speed.)

18 Licensing

18.1 License Types

Please check out http://www.tec-it.com/order for available license types and pricing.

18.2 Entering your License Data

To enter the license data select *Help ► License…* from the menu. The dialog below will appear. **TFORMer** provides two methods for entering the license data:

- Online Activation using the Activation Key (see 18.2.1).
  This is the default method. Internet access is required.
- Manual Licensing (see 18.2.2).
  Please get in touch us with us if your system has no internet access or if you prefer to use the manual activation for any other reason. We will be glad to send you the license data which is suitable for manual licensing

18.2.1 Online Activation using the Activation Key

If you have received an *activation key* from TEC-IT, please use the online activation ❶.

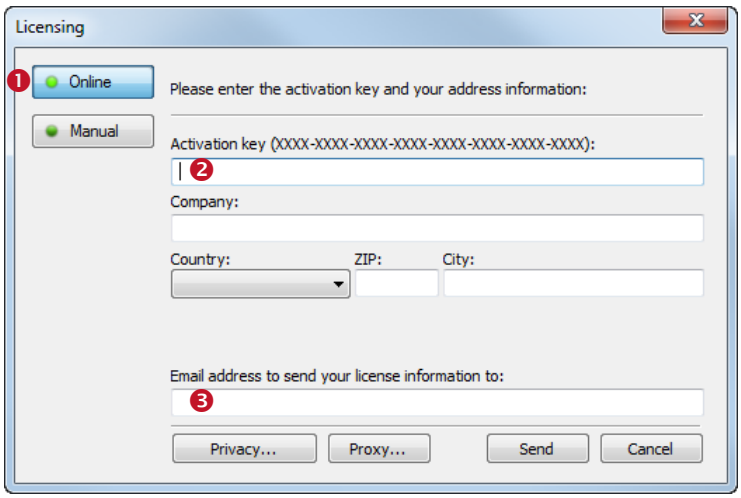


Figure 68: License Dialog – Online Activation

The following steps are required to license the product:

1. In field ❷ enter the activation key.
2. In field ❸ enter your email address. A license certification will be sent to this address automatically.
3. Complete all remaining data (Company, Country, ZIP Code, City). For a successful activation all fields are required.
4. Confirm the entered data by clicking *Send.*

A message will inform you about the successful activation. In case of problems or errors, please contact TEC-IT.

18.2.2 Manual Licensing

Manual licensing is the alternative method for licensing if your system has no Internet connection. Select manual licensing ❹ and enter the license data as provided by TEC-IT.

- ► Please enter the license data exactly as you received it from TEC-IT!
  Spacing and upper/lower case letters are to be considered. To avoid typing errors, it is recommended to use “copy and paste” whenever possible: Copy the data from the license email which you have received from TEC-IT and paste it in the license dialog.
- ► Single licenses
  If you are interested in a single (workstation) license, please tell us the so-called *"System ID”* of the target computer. You can find the *System ID* in the licensing dialog (see below).

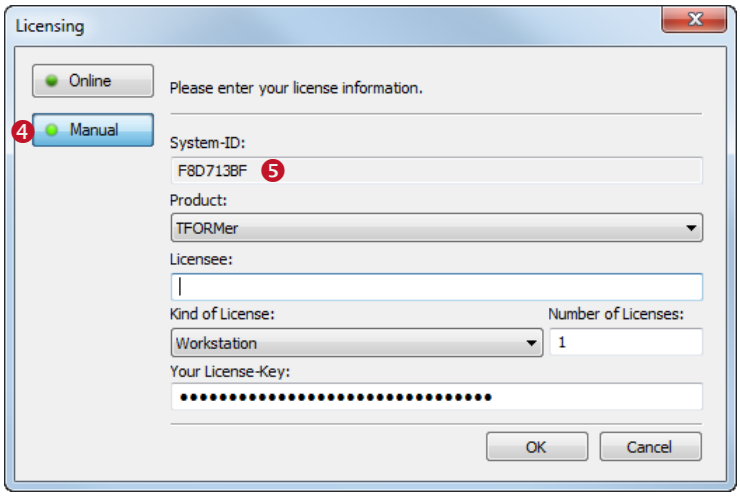


Figure 69: License Dialog – Manual Licensing

The following steps are required to license the product:

1. On the top of the dialog you see the System ID (❺) of your computer.
   For single licenses we will ask you to send us the System ID of the target computer.
2. In the field *“Product”* select **“TFORMer”.**
3. In the field *“Licensee”* enter the name of the license holder.
4. In the field *“Kind of License”* select the kind of license that you have purchased. Possible license types are:
   - Workstation
   - Site
5. In the field *“Number of Licenses”* enter the number of licenses that you have purchased.
6. In the field *“Your License-Key”* enter the license key exactly as received from TEC-IT.
7. Confirm the dialog with *OK.*

A message will inform you about the successful licensing. In case of problems or errors, please contact TEC-IT.

18.3 Notes

TFORMer stores the license data in the registry of the current user (HKEY_CURRENT_USER\Software\TEC-IT Datenverarbeitung GmbH\TFORMer\6.0). For workstation wide licensing copy this license data to the corresponding HKEY_LOCAL_MACHINE registry path.

Another possibility to license **TFORMer** is to use a license file named “TFORMer.ini”. This file must be stored in the same directory as “TFORMer.exe”. For details, please contact TEC-IT.

19 Contact and Support Information

TEC-IT Datenverarbeitung GmbH

Address: Wagnerstr. 6
AT-4400 Steyr
Austria/Europe
Phone: +43 / (0)7252 / 72 72 0
Fax: +43 / (0)7252 / 72 72 0 – 77
Email: mailto:support@tec-it.com
Web: http://www.tec-it.com

Appendix A: Properties

A.1 Form Properties

A.1.1 Common

Print Order	The print order for the *detail band.* You can choose between the following settings: ▪ Across, then Down ▪ Down, then Across The printing order will be applied only, if the number of *Columns,* the number of *Rows,* the *Column Width* or the *Row Height* are adjusted to produce a printout with multiple rows and columns (label printing). For reports the number of columns and the number of rows are set to 1 by default. In this case the print order is irrelevant.
Orientation	The orientation for the printout: ▪ Portrait ▪ Landscape ▪ Printer Default (uses the orientation, which is adjusted in the printer driver)
Datasource	Select the datasource which is used for printing. By default the manual datasource is selected. Whenever you change the datasource (e.g., in the data view), this setting is changed.

A.1.2 Advanced

Columns	The number of columns per printed page (labels). 0 = auto If set to *auto*, the number of possible columns will be calculated automatically, based on *paper width, margins, column spacing* and the *column width.* If set to any other value than *auto,* **TFORMer** will print the specified number of columns – no matter, if the width of the paper is sufficient or not.
Rows	The number of rows per printed page (labels). 0 = auto If set to *auto*, the number of possible rows will be calculated automatically, based on *paper height, margins, row spacing* and the *height of the bands.* Use this setting only if the label contains exactly one band *(detail band)* or if the row height is given. If set to any other value than *auto,* **TFORMer** will print the specified number of rows – no matter, if the height of the paper is sufficient or not.
Column Width	The column width (label width). 0 = auto If set to *auto*, the width of the worksheet (see A.1.4) will be used as column width. If set to any other value than *auto,* **TFORMer** will use the specified column width – no matter, if the width is sufficient for the label or not. When using label templates (e.g., Zweckform® or Avery®) you can input the manufacturer's label specifications here. Please note: The *column width* denotes the space requirement of a label. Besides that the *layout width* (see A.2.4) can be used for printing an additional *page header* over the entire width of the page (see Figure 24, and section D.5).
Row Height	The row height (label height). 0 = auto If the row height is set to *auto,* **TFORMer** automatically calculates the height of the labels according to the height of the *detail band* (and the group headers/footers). Use this setting only if the label contains exactly one band *(detail band)* or if the number of rows is given. If set to any other value than *auto,* **TFORMer** will use the specified row height – no matter, if the height is sufficient for the label or not. When using label templates (e.g., Zweckform® or Avery®) you can input the manufacturer's label specifications here.
Column Spacing	The column spacing (labels). When using label templates (e.g., Zweckform) you can input the manufacturer's label specifications here.
Row Spacing	The row spacing (labels).

	When using label templates (e.g., Zweckform) you can enter the manufacturer's label specifications here.
Document Name	This expression will be used as name of the generated output or spool file. If left blank, the file name is used.
Rotation	Output rotation (0, 90, 180 or 270 degrees, counter clockwise).

A.1.3 Watermark

The watermark is a background picture which is printed per output page.

File Name	The file name of the picture. See section A.3.1.4.
Picture-Mode	The display mode of the picture. See section A.3.1.4.
Picture Alignment	The alignment of the picture on the printed page. See section A.3.1.4.

A.1.4 Position

Width	The layout width.

A.1.5 Margins

Margin Left	The page margins are used to confine the printing area on the output page.
Margin Top	
Margin Right	
Margin Bottom	

A.1.6 Documentation

Comment	An arbitrary comment. For documentation purposes.

A.2 Band Properties

A.2.1 Common

Group By[2]	The records of a datasource will be grouped based upon this expression: Every time, the computed value of the expression changes, a new group will be started. Group-by expressions do not change the order of the records in the datasource. Group headers are printed when a group starts, group footers are printed after a group ends.
Page Break[3]	Determines whether a page break will be inserted before and/or after this band. Possible values: ▪ None ▪ Before this Band ▪ After this Band ▪ Before and After this Band
Row/Column Break[3]	Determines whether a row/column break will be inserted before and/or after this band. Possible values: ▪ None ▪ Before this Band ▪ After this Band ▪ Before and After this Band
Pre-Evaluation	This expression will be evaluated before the band is printed (see section 12.4).
Post-Evaluation	This expression will be evaluated after the band is printed (see section 12.4).

A.2.2 Advanced

Tray	For every single page of the output you can dynamically select a tray on the target printer. E.g., this might be useful for printing the first page of an invoice on a letterhead. The tray selection is performed by the first band on a new page (e.g., by the *page header)*. The tray that is entered in the property of this band will be selected: ▪ Tray 0 is the default tray (uses the setting of the current printer driver). ▪ Tray 1 to 10 can be pre-configured *(Tray Mappings ► Trays).* For more details, please refer to section 12.6.
Type	The band type (Report Header, Page Header, …).
Output Area[4]	The area, in which the band shall be printed: ▪ Page ▪ Label
Print at Bottom[5]	By default, the report footer will be printed directly after the last *detail band.* By setting this value to *True,* the report footer will be printed at the bottom of the page (before the page footer).

A.2.3 Control

Printing Condition	The printing condition decides at print-time, whether the band is printed or not (see section 12.2).

A.2.4 Position

Height	The band height.
Width	The band width. Same as the layout width (see section A.1.4).
Can Grow	Specifies, whether the height of the band is increased if required. This is useful if the band contains elements with dynamic size (possible for text and picture elements). Allowed values are: ▪ No ▪ Height

[2] For *group header* and *group footer.*
[3] Not for *page header* and *page footer*
[4] For *report header, page header, page footer* and *report footer.*
[5] For *report footer.*

Can Shrink	Specifies, whether the height of the band is decreased if possible. This is useful if the band contains elements with dynamic size (possible for text and picture elements). Allowed values are: ▪ No ▪ Height

A.2.5 Documentation

Name	The name of the band (used for display only).
Comment	An arbitrary comment. For documentation purposes.

A.2.6 Columns

The group *Columns* is available for *detail bands* only.

Columns	Specifies, how many columns are used for printing the detail bands. If activated, each new record (respectively each record copy) is printed in a subsequent column. When designing the layout you have to consider, that only the respective percentage of the column width may actually be used for design elements (e.g., for 2 columns you may only cover 50% of the width). The Row/Column Break has to be removed.
Gap	The gap between the columns.
Width	The width of a column.

For the alternating output of two or more detail bands you have to consider, that each involved detail band must have the same column settings!

A.3 Element Properties

A.3.1 Common

A.3.1.1 Text Elements

Text	The text to be printed.
Font	Selects the font style and size.
Fore Color	The color of the text.
Fill Color	The color and the pattern of the text background.
Line Color	The color and the pattern of the bounding rectangle.
Line Width	The width of the bounding rectangle.
Line Style	The line style for the bounding rectangle (solid, dashed, …).
Text Alignment	The alignment of the text within the bounding rectangle. The following values are possible: Top, Left | Top, Center | Top, Right Center, Left | Center | Center, Right Bottom, Left | Bottom, Center | Bottom, Right
Justify	Turn justify on/off.
Word Wrap	If enabled, line-breaks will be ignored.
Text Rotation	Text rotation (0, 90, 180 or 270 degrees, counter clockwise).

A.3.1.2 Barcode Elements

Barcode-Data	The data which is encoded as barcode symbol. The barcode data may contain static data and/or dynamic data (e.g., datafields).
Barcode Type	The barcode type (symbology).
Check Digit	Selects the check digit computation.
Barcode Color	The color of the bars.
Font	Selects the font style and size.
Fore Color	The color of the text.
Fill Color	The color and the pattern of the background.
Human Readable Text	The position of the human readable text (Above, Below, No).
Human Readable Text Distance	The distance between the barcode and the human readable text.
Rotation	Barcode rotation (0, 90, 180 or 270 degrees, counter clockwise).

A.3.1.3 Line, Rectangle, Ellipse

Fill Color	The color and the pattern of the text background. (This property is not available for line elements.)
Line Color	Line color and pattern.
Line Width	The line width.
Line Style	The line style (solid, dashed, …).
Radius	The radius of the rounded corner. (This property is only available for rounded rectangles.)

A.3.1.4 Picture

File Name	The image file name. The following file formats can be used: .bmp, .gif, .jpg, .pcx, .png, .tga, .tif. File names can be specified using one of the following formats: ▪ Windows file format (e.g., “C:\temp\sample.png”) ▪ URL format (only available under windows!) (e.g., “http://www.tec-it.com/pics/sample.png”)

	You can use absolute file names (e.g., "C:\sample.jpg") and relative file names (e.g., "sample.jpg" or "img\sample.jpg"). As base for relative path specifications the path of the **TFORMer** file (.tff or .xmd) is used.
Alignment	The alignment of the picture within the bounding rectangle. The following values are possible: Top, Left | Top, Center | Top, Right Center, Left | Center | Center, Right Bottom, Left | Bottom, Center | Bottom, Right
Rotation	Picture rotation (0, 90, 180 or 270 degrees, counter clockwise).
Mode	The display mode: ▪ Clip (The picture will be printed in its original size and might be clipped by the bounding rectangle.) ▪ Stretch (The picture will be stretched to fit the bounding rectangle, regardless of its aspect-ratio.) ▪ Zoom (The picture will be enlarged to its maximum possible size within the bounding rectangle, keeping its aspect-ratio.) ▪ Tile (The picture will be replicated row- and column-wise inside the bounding rectangle.)
Transparent Color	All regions in the image that are filled with the selected color will be printed transparently.

A.3.2 Advanced

A.3.2.1 Text Elements

Auto Font-Size	If set to *True,* **TFORMer** automatically selects the biggest possible font size, for which the text to fits into the bounding rectangle without being cropped.
Auto Font-Size Minimum	Specifies the lower font size limit (in percent), if *Auto Font-Size* is enabled.
Auto Font-Size Maximum	Specifies the upper font size limit (in percent), if *Auto Font-Size* is enabled.
Optimize Spaces	If enabled, multiple consecutive spaces will be printed as one single space character. Sample: "**TFORMer Designer** " will be printed as "**TFORMer Designer**".
Suppress Blank Lines	If enabled, empty lines in a multi line text will not be printed. Sample: "**TFORMer** **Designer**" will be printed as "**TFORMer** **Designer**"
Line-Spacing	The spacing between text lines.

A.3.2.2 Barcode Elements

For more information on the following properties, please refer to the Barcode Reference, which is available for download on www.tec-it.com.

Escape Sequences	Turn the translation of escape sequences on/off.
Hex-Data	Enable this property if your barcode data is provided as hex-string. **TFORMer** will convert it to ASCII characters automatically. Example: The hex-string "34373131" will be converted into the barcode data "4711".
Format	The format property is used for formatting the utilizable data of the barcode. It is specified via a string that operates with substitute symbols to indicate how the data shall be structu-red. Specifying a format string, you can: ▪ Select subsets for Code 128 and UCC/EAN-128. ▪ Define the desired Start/Stop characters for CODABAR. ▪ Reposition the Check-Digit (in special cases). ▪ Encode the date, preamble, service class, postal and country code into the barcode data (MaxiCode).
Bearer Width	Adjust the horizontal lines above and below the barcode: 0 inch (or mm) … no line x inch (or mm) … use the specified line width
Module Width	The module width is the width of the narrowest bar.

Notch Height	The notch height specifies how far the synchronization bars should jut out from the remaining barcode. (E.g., the EAN code has synchronization bars on the left, in the middle and on the right.)
Bar:Space Ratio	This property is used to specify the relationship between the widths of the single bars and spaces of a barcode. The input format depends on the selected barcode type. – For Code 2OF5 Interleaved you could specify the ratio “1:2:1:2”, e.g.
Suppress Error Message	In case of an error during the printing of a barcode (e.g., wrong input data), an error message will be printed. This option allows you to suppress the error message. The space will be left blank instead.
Optimal Resolution	Automatically adjust the module width of the barcode (decrease it) to avoid aliasing problems on the output device. The module width is set to a multiple of the smallest available printing unit.
Bar-Width Reduction	When printing on inkjet printers, the ink that is absorbed by the paper tends to diffuse. Setting the bar width reduction allows you to work against this spreading of ink. The bar width reduction is specified in percent of the module width. E.g., when setting this value to 20 all bars will be narrowed by 20 percent of the module width.

A.3.3 Control

Printing Condition	The printing condition decides at print-time, whether the element will be printed or not (see section 12.2).
Layer	Assign the element to one of the predefined layers (see section 12.3).

A.3.4 Position

Left	The left most coordinate of the element.
Top	The top most coordinate of the element.
Right	The right most coordinate of the element.
Bottom	The bottom most coordinate of the element.
Width	The width of the element.
Height	The height of the element.
Can Grow[6]	Specifies, whether the dimensions of the element are adjusted according to its content. If a text is clipped inside the frame (relevant for dynamic content only) you can use the property *Can Grow*. The size of the text element will be increased automatically as required. Allowed values are: ▪ No ▪ Height ▪ Width ▪ Height and Width If the height of the band which contains this element should grow accordingly, enable the “Can Grow” property of the band.
Can Shrink[6]	Specifies, whether the dimensions of the element are adjusted according to its content. Allowed values are: ▪ No ▪ Height ▪ Width ▪ Height and Width If the height of the band which contains this element should shrink accordingly, enable the “Can Shrink” property of the band.
Flush Top	Adjust the top border to the specified band edge. Useful for drawing column separator lines in tables. Allowed values are: ▪ No ▪ Last Printed Band
Flush Bottom	Adjust the bottom border to achieve a common baseline. Useful for aligning an element to the bottom position of the band or other elements. Allowed values are:

[6] For *text* and *picture* elements

	▪ No ▪ Elements ▪ Band
Flush Right	Adjust the right border to achieve a common vertical line to the right. Useful for aligning the right border of an element to the right position of the band or other elements. Allowed values are: ▪ No ▪ Elements ▪ Band
Shift Mode	This property is used to move elements relative to other growing or shrinking elements. “Always” considers size-changes of other elements immediately. “When overlapped” only avoids intersections. Allowed values are: ▪ Always ▪ Don’t shift ▪ When overlapped

A.3.5 Margins

These properties are available for *Text Elements* only:

Margin Left	These margins are used to confine the printing area within a text element.
Margin Top	
Margin Right	
Margin Bottom	

A.3.6 Documentation

Name	The name of the element.
Comment	An arbitrary comment. For documentation purposes.

A.4 Layer Properties

A.4.1 Common

Name	The name of the layer.
Display Color	If *Display Layer colors* (✎) is enabled, all elements, which are assigned to that layer, will be drawn in the specified color.
Visible	Here you can toggle all elements of one layer visible or invisible. Please note: This is for editing purposes only! – The output will not be affected.

A.4.2 Control

Printing Condition	The printing condition decides at print-time, whether the elements, which are assigned to that layer, will be printed or not (see section 12.2).

A.4.3 Documentation

Description	A simple description. Only for documentation purposes.
Comment	An arbitrary comment. For documentation purposes.

A.5 Datafield Properties

A.5.1 Common

A.5.1.1 Datafield

Name	The name of the datafield. This name can be used to address the datafield in expressions.
Data Type	The type of the datafield. Choose one of the following: ▪ Text ▪ Integer ▪ Floating-point We recommend the general use of the data-type *"Text"*.
Default Value	The default value for the datafield. This value will be returned, if no other value is given.

A.5.1.2 Computed

Name	The name of the datafield. This name can be used to address the datafield in expressions.
Data Type	The type of the datafield. Choose one of the following: ▪ Text ▪ Integer ▪ Floating-point
Expression	The expression which is used to compute the value for the datafield.
Compute Per	Choose if the datafield is re-computed for every: ▪ Record ▪ Record copy

A.5.1.3 Serial

Name	The name of the datafield. This name can be used to address the datafield in expressions.
Data Type	The type of the datafield. Choose one of the following: ▪ Integer ▪ Floating-point
Start Value	The start value for the serial number. If the “Store last value” property is enabled (see below), the start value will be updated automatically after each print-job.
Step Size	The step size in which the serial number is incremented.
Store Last Value (y/n)	If enabled **TFORMer** remembers the last printed value. The next unused value will be assigned as start value for the next print job automatically.
Update on	Choose the trigger, on which the serial number is incremented: ▪ Record ▪ Record copy ▪ Document ▪ Page ▪ Label
Update Condition	An optional condition controlling the increment of the serial number. If a condition was entered, the serial number will only be increased if the result of the expression is true.

A.5.2 Validation (Datafield)

The *Validation* group is only available for normal datafields.

Validation Rule	This rule is verified when importing data.
Validation Message	If the validation-rule is violated, this message will be prompted.

A.5.3 Aggregation (Computed)

The *Aggregation* group is only available for computed datafields.

Aggregation Type	Selects one of the available aggregation functions or disables aggregation. Available values are: ▪ None ▪ Running Average ▪ Running Sum
Aggregation Region	If aggregation is enabled, select the region for which the aggregation values are calculated: ▪ All (do one aggregation within the whole range of the print-job) ▪ Page (do a separate aggregation within the range of each new page) ▪ Label (do a separate aggregation within the range of each new label) ▪ Group (do a separate aggregation within the range of each new group)
Group	If the aggregation region is set to “Group”, use this field to enter the grouping condition. Enter the same condition as specified in the group header or group footer.
Aggregation Filter	An optional filter for aggregation values. If a filter was entered, single values will only be considered, if the expression returns true. Thus you can define, which values are considered for the aggregation, and which not.

A.5.4 Advanced

Caption	These properties may be queried with the **TFORMer SDK** API. They have no functionality for printing.
Display Order	This property is used by the data grid of **TFORMer Designer** and by the tool **QuickPrint** to determine the order in which datafields are displayed. By default **TFORMer** sorts the data-fields alphabetically.

A.5.5 Documentation

Description	A simple description. Only for documentation purposes.
Comment	An arbitrary comment. For documentation purposes.

A.6 Tray Mapping Properties

A.6.1 Common

Name	The name of the tray mapping.

A.6.2 Trays

Tray 1	For each of the 10 logical trays in **TFORMer** you can assign a device-specific printer tray. You can assign tray names or paper format names.
…	
Tray 10	

A.6.3 Documentation

Description	A simple description. Only for documentation purposes.
Comment	An arbitrary comment. For documentation purposes.

A.7 Repository

Description	A simple description. Only for documentation purposes.
Comment	An arbitrary comment. For documentation purposes.

A.8 Repository: Tray Mapping Properties

Same as the *Tray Mapping Properties* described in A.6.

A.9 Repository: Global Datafields

Same as the *Datafield Properties* described in 0.

A.10 Repository: Projects

A.10.1 Common

Name	The name of the project.
File	The file name of the form.

A.10.2 Documentation

Description	A simple description. Only for documentation purposes.
Comment	An arbitrary comment. For documentation purposes.

Appendix B: Supported HTML Tags and Entities

B.1 HTML Tags

Tag	Description
	Paragraph. starts a new paragraph. ends the paragraph.
 	Line break. Inserts a line break.
,	Bold text. Everything between the start tag and the end tag is rendered bold.
	Italic text. Everything between the start tag and the end tag is rendered italic.
<u>	Underlined text. Everything between the start tag <u> and the end tag </u> will be underlined.
<strike>	Strike through text. Everything between the start tag <strike> and the end tag </strike> will be striked through.
<font>	Font. The following attributes are supported: ▪ color Only hexadecimal values, no color names. E.g., <font color=”#ff0000”>. ▪ name Selects a typeface. E.g., <font name=”Helvetica”> ▪ size Use one of the predefined HTML font sizes within the range from 1 to 7. E.g., <font size=”2”> ▪ style Supports the “font-size” attribute only. E.g., <font style="font-size: 12pt">
<a>	Anchor. Used for embedding links. Only the href attribute is supported. E.g., Link to Google
<Expr>	Used by **TFORMer** internally. This tag embeds expressions.
<HtmlExpr>	Used by **TFORMer** internally. This tag embeds expressions, which are returning HTML.

B.2 Named Entities

Named Entity	Description
>	">" character (greater than).
<	"<" character (less than).
&	"&" character (ampersand).
	Non breaking space.

Appendix C: Function Reference

C.1 Functions

Return	Function	Description
long	Abs («Number»)	Returns the absolute value of a number.
date	AddDays ()	Adds or subtracts the specified number of days to/from a date value.
date	AddMonths ()	Adds or subtracts the specified number of months to/from a date value.
date	AddWeeks ()	Adds or subtracts the specified number of weeks to/from a date value.
date	AddYears ()	Adds or subtracts the specified number of years to/from a date value.
Long	Asc («Text»)	Returns the ASCII value of a given character or of the first character of «text».
date	CDate («Text»)	Converts the string «text» to a date. Provides an empty date if no conversion is possible. The date format in «Text» depends on the regional settings on your computer. This setting is configured in the *Control Panel ► Region and Language* under „Formats“. ▪ German: Format = “24.12.2004” ▪ English: Format = “12/24/2004”
date	CDateEx («Text», «Format»)	Converts a string into a date. The conversion format to be used is adjustable. For information on how to specify the format, please refer to the description of the function *Format («Date», «Format»)* below. Example CDateEx ("31.12.2000", "dd.MM.yyyy").
double	CDbl («Expr»)	Converts any value to a double value (floating-point notation). The result is 0.00 when a conversion is not possible.
string	CheckDigits («Method», «Text»)	Returns the check digit for the given string as text. Possible values for «Method» are: 2 (Modulo 10), 3 (Modulo 43), 4 (Modulo 47, 2 digits), 5 (DP Leitcode), 6 (DP Identcode), 7 (Code11, 1 digit), 8 (Code11, 2 digits), 9 (USPS PostNet), 10 (MSI, 1 digit), 11 (MSI, 2 digits), 12 (Plessey), 13 (EAN 8), 14 (EAN 13), 15 (UPC A), 16 (UPC E), 17 (EAN 128), 18 (Code 128), 19 (Royal Mail 4 State), 20 (Modulo-11, PZN), 21 (Modulo-11, W=7), 22 (EAN 14), 23 (Modulo 10, Korean PA), 24 (Modulo 10, Planet), 25 (Modulo 10, Italian Postal 2/5), 26 (Modulo 36, DPD Barcode), 27 (Modulo 16), 28 (Modulo 10, Luhn Algorithm).
long	CheckDP («Text»)	Returns the Deutsche Post check-digit (Leitcode, Identcode) of the specified value as long.
long	CheckMod10 («Text»)	Returns the Modulo 10 check-digit (weighted) as ASCII code. E.g., CheckMod10(”1203”) will return 52, which is the ASCII code for the character “4”. Please, use the function *Chr(CheckMod10(..))* to get the result as a character.
char	CheckMod36 («Text»)	Returns the Modulo 36 check-digit of the specified value as character.
char	CheckMod43 («Text»)	Returns the Modulo 43 check-digit of the specified value as character.
char	Chr («Number»)	Returns the corresponding character for the specified ASCII value «Number».
long	CLng («Expr»)	Converts any value into a whole number. If a conversion is not possible, the result is 0.
string	CStr («Expr»)	Converts a value into a text.
long	Day («Date»)	Determines the day of the month [1..31].
long	DayOfWeek («Date»)	Returns the day of the week of a specified date [1..7]. 1=Sunday, 2=Monday, ...
double	DayOfYear («Date»)	Returns the day of the year of a specified date [1..366].
double	Exp («Number»)	Returns the value $e^{«Number»}$, where e is the base of the natural logarithms.
double	Exp10 («Number»)	Returns the $10^{«Number»}$.
long	Find («Text», «SearchText», «nStart»)	Searches the string «Text» for «SearchText» starting from Position «nStart». Returns the position of the string or -1. The first character of a string is located at position 0.
long	FindReverse («Text», «SearchText», «nExclude»)	Searches the string «Text» for «SearchText» in reverse order excluding «nExclude» characters at the end. Returns the position of the string or -1. The first character of a string is found at position 0.

string	Format («Number», «Pattern»)	Formats «Number» according to the specified pattern string «Pattern». Format placeholders: # digit or no value, 0 '0' or digit . decimal point , comma + - sign
string	Format («Date», «Pattern»)	Formats «Date» according to the specified pattern string «Pattern». Format placeholders: dd day of month (01 – 31) MM month double-digit (01 – 12) MMM month three-digit (Jan – Dec) yy year double-digit (01) yyyy year four-digit (2001) Example: *Format (CDate("12/24/2009"), "MMM, dd yyyy")* returns *Dec, 24 2009*. See also section C.4 Formats.
double	Fract («Number»)	Returns the fractional unit of «Number».
long	Hour («Date»)	The hour of a specified date [00..23].
string	IIf («Condition», «TrueExpr», «FalseExpr»)	Returns the value of «TrueExpr» if «Condition» is evaluated as (TRUE or not equal to 0) . Returns the value of «FalseExpr» if «Condition» is evaluated as (FALSE or equals 0).
bool	IsDate («Text», «Format»)	Checks if the given string can be converted into a valid date using the specified conversion format. For information on how to specify the format, please refer to the description of the function *Format («Date», «Format»)* above.
long	IsEmpty («Text»)	Test whether the string «Text» is empty or not.
long	IsEven («Number»)	Returns TRUE if «Number» is even.
long	IsLastPage ()	Returns TRUE if the page being printed is the last page of the document.
bool	IsLeapYear («Date»)	Returns TRUE if the specified date occurs within a leap year.
long	IsOdd («Number»)	Returns TRUE if «Number» is uneven.
string	KeepChars («Text», «KeepChars»)	Removes all characters in «Text» which are NOT included in «KeepChars».
string	LastValue («VarName»)	Returns the last value of a given datafield (the value of the datafield from the previous record).
string	Left («Text», «nLength»)	Returns the first characters «nLength» of a string.
long	Len («Text»)	Returns the length of the given string.
double	Log («Number»)	Returns the natural logarithm of «Number».
double	Log10 («Number»)	Returns the logarithm of «Number».
string	Mid («Text», «nStart», «nLenght»)	Returns the substring of string «Text» starting at position «nStart» with length «nLength». The first character of a string is located at position 0.
long	Minute («Date»)	The minutes of a specified date/time [00..59].
long	«NumberA» % «NumberB»	Modulo operator: Remainder of the integer division «NumberA» / «NumberB».
long	Month («Date»)	The month of the specified date [1..12].
date	Now ()	The actual date and time.
double	Pow («Number», «Power»)	Returns the result of «Number» raised to the power of «Power».
string	RemoveChars («Text», «RemoveChars»)	Removes all characters in «Text» which are included in «RemoveChars».
string	Replace («Text», «SearchText», «ReplaceText»)	Replaces each occurrence of «SearchText» in string «Text» with «ReplaceText».
string	Right («Text», «nLength»)	Returns the last characters «nLength» of a string.
double	Round («Number», «Precision»)	Returns «Number» rounded using precision digits. If 0 is «Precision» the result will be rounded to a whole number.

long	Second («Date»)	The second of a specified date/time [00..59].
double	Sqrt («Number»)	Returns the square root of «Number».
long	SumOfDigits («Number»)	The sum of all digits of «Number».
long	SumOfDigits1 («Number»)	Returns the one digit sum of all digits of «Number».
string	ToLower («Text»)	Converts all character in the string «Text» to lower case.
string	ToUpper («Text»)	Converts all character in the string «Text» to upper case.
string	Trim («Text»)	Removes leading and trailing spaces.
string	TrimLeft («Text»)	Removes leading spaces.
string	TrimRight («Text»)	Removes trailing spaces.
double	Value («Text»)	Converts «Text» to a double value.
long	WeekOfYear («Date»)	The calendar week of a specified date/time [1..52].
long	Year («Date»)	The year of the specified date/time.

Table 1: Functions

C.2 System Datafields

Function	Description
BandName	The name of the current band.
BandType	The type of the current band.
Column	The current column number (label printing) [0..x].
ComputerName	The name of the computer (hostname).
Copy	The actual number of copies [1..NumCopies].
Device	The name of output device (Printer).
Device Type	The type of the output device (usually Winspool).
Form	The name of the current form.
LogPage	The number of the current *logical page*[7] [1...NumLogPages]. Several logical pages can be printed per physical page.
NumCopies	The count of all copies [1..x].
NumLogPages	The count of all *logical pages*[7] [1...x]. Several logical pages can be printed per physical page.
NumPages	The count of all physical pages (sheets) [1..x].
NumPrintItems	Total number of items to be printed. Sum of the copy count of all records.
NumRecordCopies	The number of copies of the actual record.
NumRecords	The count of all input-records.
Page	The actual page number [1..NumPages]. Every printed physical page (sheet) increases the page number by 1.
PrintedItems	Number of currently printed items, including the current item.
Project	The name of the current project.
ProjectDir	The directory of the current project.
Record	The actual record number [1..NumRecords].
RecordCopy	The actual number of record copy.
Repository	The name of the current repository.
RepositoryDir	The directory of the current repository.
Row	The current row number (by Label print) [0..x].
tfDocumentName	This name will be used for the spool-job under Microsoft Windows. Can be set via an

[7] A *logical page* is usually the area of a label on the output page. Every *Row/Column Break* starts a new logical page.

	expression.
tfServerJobFile	The job file of **TFORMer Server**.
tfServerJobID	The job ID of **TFORMer Server**.
tfServerJobTimesPrinted	For **TFORMer Server** only: How often has this server job already been printed.
XResolution	Horizontal device resolution in dots per inch (25.4 mm).
YResolution	Vertical device resolution in dots per inch (25.4 mm).

Table 2: System Datafields

C.3 Common Expressions

Expression	Description
"- " + CStr (Page) + " -"	Inserts the page number into the layout.
"EUR " + Format («Number», "########.00")	Formats a value into a EUR price.
"Page " + CStr (Page)	Inserts a formatted page number into the layout.
"Page " + CStr (Page) + " of " + CStr(NumPages)	Inserts the page number and the count of all pages into the layout.
"USD " + Format («Number», "########.00")	Formats a value into a USD price.
CStr (NumPages)	Inserts the count of all pages into the layout.
Format (Now (), "hh':'mm':'ss tt")	Inserts the actual time (12 hours).
Format (Now), "HH':'mm':'ss")	Inserts the actual time (24 hours).
Now ()	Inserts the current date into the layout.

Table 3: Common Expressions

C.4 Formats

C.4.1 Numbers

Expression	Description
Format («Number», "#,###,##0.00+")	Formats a number with a trailing sign (+ or -) (e.g., 1.299,20+).
Format («Number», "#,###,##0.00-")	Formats a number with an optional trailing sign (only if negative) (e.g., 1.299,20-).
Format («Number», "*,***,**0.00-")	Formats a number with leading starts and an optional trailing sign (only if negative) (e.g., ***1.299,20-).
Format («Number», "+#,###,##0.00")	Formats a number with a leading sign (+ or -) (e.g., +1.299,20).
Format («Number», "-#,###,##0.00")	Formats a number with an optional leading sign (only if negative) (e.g., -1.299,20).
Format («Number», "-0.000.000.00")	Formats a number with an optional leading sign (only if negative) (e.g., -0.001.299,20).

Table 4: Number Formats

C.4.2 Date

Expression	Description
Format (Now (), "dd'-'MM'-'yy")	Formats the actual date (e.g., 29-01-07).
Format (Now (), "dd'-'MM'-'yyyy")	Formats the actual date (e.g., 29-01-2007).
Format (Now (), "dd'.'MM'-'yyyy")	Formats the actual date (e.g., 29. Jan '07').
Format (Now (), "dd'.'MM'.'yyyy")	Formats the actual date (e.g., 29.01.2007).
Format (Now (), "ddd','d'.'MMM'.'yyyy")	Formats the actual date (e.g., Mon., 29. Jan. 2007).
Format (Now (), "dddd','d'.'MMM'.'yyyy")	Formats the actual date (e.g., Monday, 29. Jan. 2007).
Format (Now (), "dddd','d'.'MMMM yyyy")	Formats the actual date (e.g., Monday, 29. January 2007).
Format (Now (), "h'.'m'.'s tt")	Formats the current system time and appends am or pm (according to the language settings of your system) (e.g., 9:6:5 am/pm).
Format (Now (), "H'.'m'.'s")	Formats the current system time (e.g., 21:6:5).

Format (Now (), “h’.’mm’.’ss tt”)	Formats the current system time and appends am or pm (according to the language settings of your system) (e.g., 9:06:05 am/pm).
Format (Now (), “H’.’mm’.’ss”)	Formats the current system time (e.g., 7:06:05).
Format (Now (), “hh’.’mm’.’ss tt”)	Formats the current system time and appends am or pm (according to the language settings of your system) (e.g., 09:06:05 am/pm).
Format (Now (), “HH’.’mm’.’ss”)	Formats the current system time (e.g., 21:06:05).

Table 5: Date Formats

C.5 Constants

Constant	Description
"\n"	Linefeed.
False	Logical value FALSE. This value is usually the result of a condition. If the condition is not fulfilled the resulting value is FALSE.
True	Logical value TRUE. This value is usually the result of a condition. If the condition is fulfilled the resulting value is TRUE.

Table 6: Constants

Appendix D: Layout Schemes

D.1 General

When printing, **TFORMer** fills each physical page with page headers, detail bands, page footers, etc. If no space is left a new physical page is started.

TFORMer offers the possibility to divide a physical page into multiple logical areas (= labels). When doing this, each logical area (label) is treated like a physical page. The division into logical areas is done using the *"row"* and *"column"* properties of the layout:

- Label-style output is achieved by dividing the physical page into multiple rows and columns.
- Report-style output is achieved otherwise.

Based on this rules various layout schemes can be created. The following examples demonstrate the most common page layouts.

Figure 70 to Figure 76 use the following color keys for page areas and bands:

- The red frame marks the page area, which may, on demand, be divided into multiple labels.
- The dashed frame marks one label (one logical area).
- Depending on the settings for this band, it is either printed inside or outside of the label area.
- This band is always printed inside of the label area.

D.2 Reports

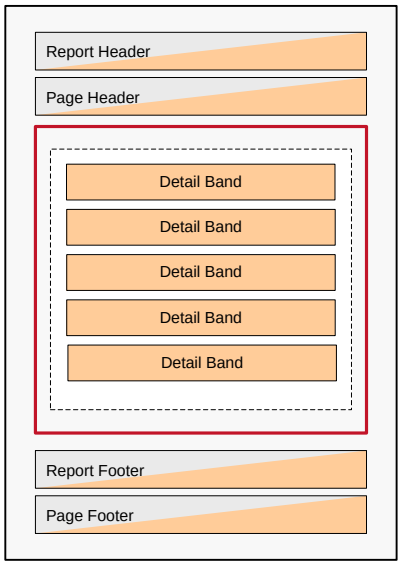


Figure 70: Default Report Configuration

The *"Custom Report"* template (see section 6.2.2.1) adjusts the form and band properties as follows:

Form:

Advanced	
Columns	1
Rows	1
Column Width	(auto)
Row Height	(auto)
Column Spacing	0.000 mm
Row Spacing	0.000 mm

Detail Band:

Common	
Page Break	None
Row/Column Break	None

The settings for rows and columns are both set to one. Therefore the physical page is not divided into sub-areas. The detail bands are printed below each other without any intermediate page or column breaks.

Example:
File ► New Form… ► <Custom Report>

D.3 Labels on Normal Printers

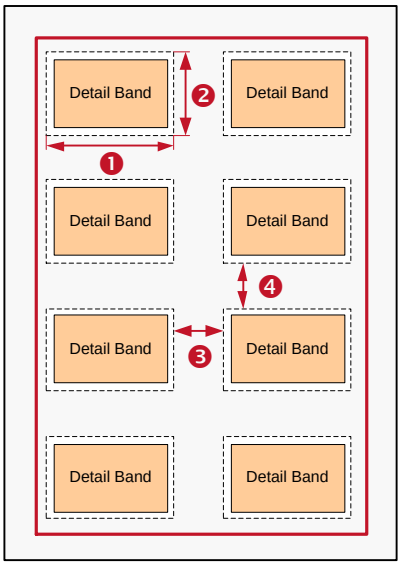


Figure 71: Default Label Configuration (Normal Printer)

The *"Custom Label – Normal Printer"* template (see section 6.2.2.2) adjusts the form and band properties as follows:

Form:

Advanced	
Columns	2
Rows	5
Column Width ❶	50.000 mm
Row Height ❷	50.000 mm
Column Spacing ❸	0.500 mm
Row Spacing ❹	0.500 mm

Detail Band:

Common	
Page Break	None
Row/Column Break	After this Band

The label dimensions (❶ and ❷) and the number of labels per page (❸ and ❹) are specified.

If you want to print the labels on different paper sizes you can set the number of rows and columns to "auto". In this case **TFORMer** will calculate the number of labels that fit on the output page.

Example:
File ► New Form… ► <Custom Label> ► Normal Printer

D.4 Labels on Label Printers

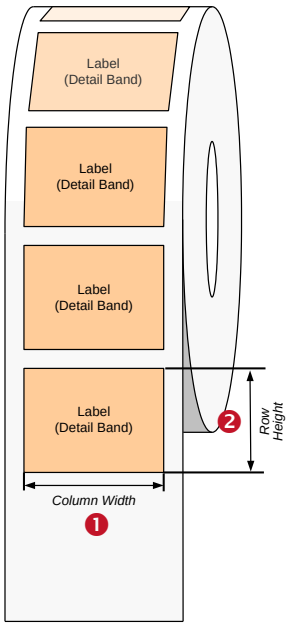


Figure 72: Default Label Configuration (Label Printer

The *"Custom Label – Label Printer"* template (see section 6.2.2.3) adjusts the form and band properties as follows:

Form:

Advanced	
Columns	1
Rows	1
Column Width ❶	50.000 mm
Row Height ❷	50.000 mm
Column Spacing	0.000 mm
Row Spacing	0.000 mm

Detail Band:

Common	
Page Break	None
Row/Column Break	After this Band

Rows and *Columns* are both set to one (one label is one output page). The column width (❶) and the row height (❷) correspond to the label size which is the same as the size of the output page.

Example:
File ► New Form… ► <Custom Label> ► Label Printer

D.5 Labels with Headers and Footers

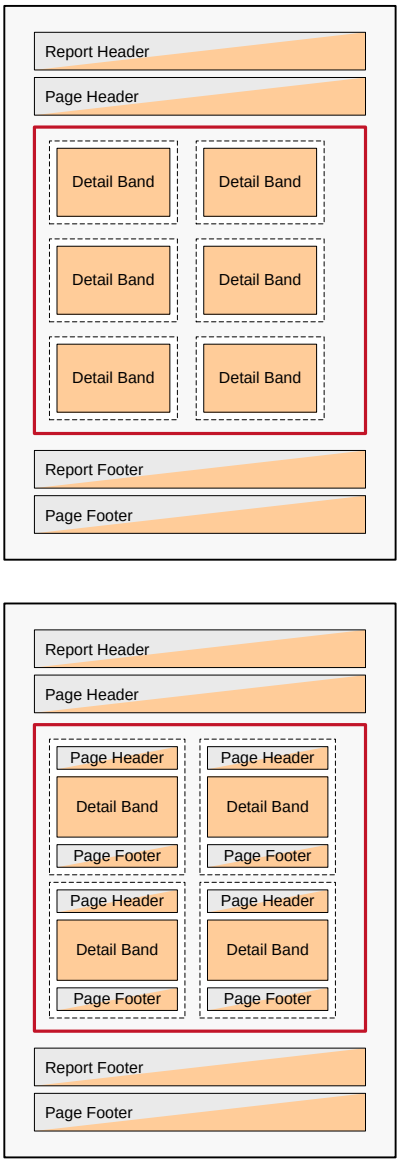
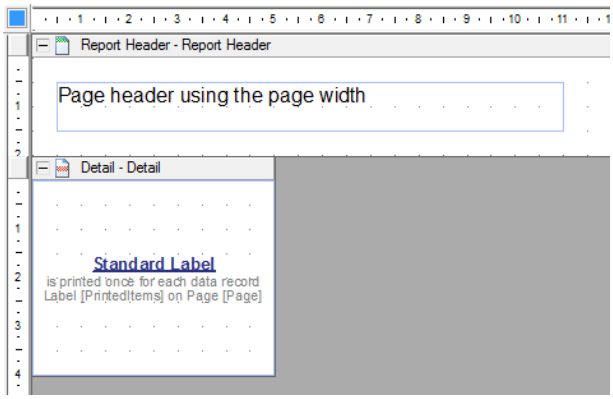


Figure 73: Labels with Headers and Footers

This is a variant of the *"Custom Label"* template with additional headers and footers.

Header/ Footer:	Advanced	
	Output-Area	Page

Setting the *Output-Area* to *"Page"* (= default) instructs **TFORMer** to print the header or footer bands using the whole physical page width.



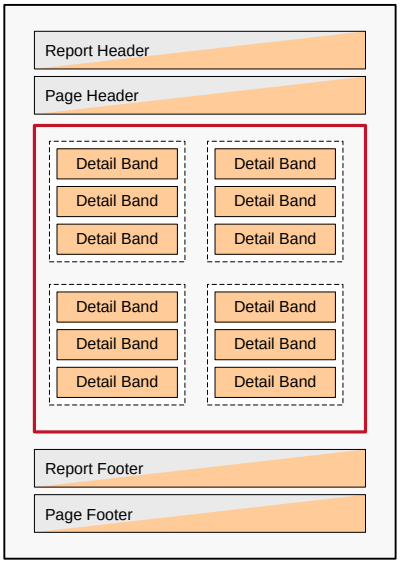
When setting the *Output-Area* to *"Label"* **TFORMer** will print the header or footer bands within the labels.

Header/ Footer:	Advanced	
	Output-Area	Label

Please note:
You can add more than one header (or footer) to your design. For example, one page header which is printed per page and one page header which is printed per label.

Example:
File ► New Form… ► (6) Samples ► Label_2

D.6 Reports in Labels



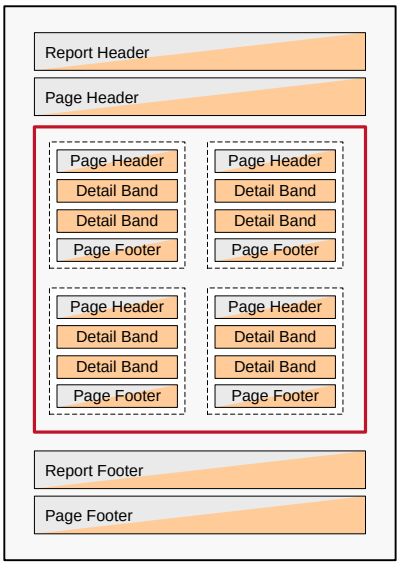


Figure 74: Reports in Labels

To print multiple detail bands inside one label you have to do the following: Beginning with the *"Custom Label"* remove the *Row/Column Break* for all detail bands.

This will make **TFORMer** print as many detail bands as possible for the given row height.

Form:

Advanced	
Columns	2
Rows	5
Column Width	50.000 mm
Row Height	60.000 mm
Column Spacing	0.500 mm
Row Spacing	0.500 mm

Detail Band:

Common	
Page Break	None
Row/Column Break	None
Position	
Height	20.000 mm

Make sure to use reasonable row and detail band heights! In the example above three detail bands will fit into one label (60 / 20 = 3).

Please note:
Headers and footers may be added as required (see section D.5).

Examples:
File ► New Form… ► (6) Samples ► Report_3
File ► New Form… ► (6) Samples ► Report_4

D.7 Report with Multiple Columns

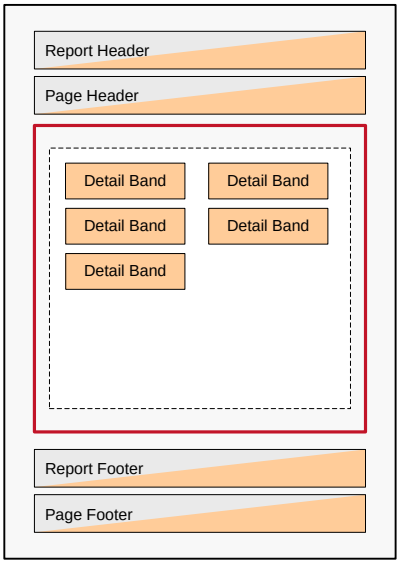


Figure 75: Report with Multiple Columns

Apart from dividing the page into multiple rows and columns, you can also create multiple columns within a detail band.

This layout is based on the *"Custom Report"* template:
The number of columns is adjusted in the in the *Columns* group in the detail band properties. In addition, make sure to remove all page and row/column breaks in the detail band. The print order is always *"across, then down".*

Detail Band:

Common	
Page Break	None
Row/Column Break	None
Position	
Height	30.000 mm
Width	200.000 mm
Columns	
Columns	2
Gap	0.000 mm
Width	(auto)

Please note:
Since the number of columns is set to 2, the design on the detail band must not cover more than 50% of the detail band width (in this case 100.000 mm)!

Hint:
When using a fixed number of columns and *Width="auto",* the available horizontal space is divided equally. Otherwise all bands are printed from left to right, without any additional horizontal spacing.

Example:
File ► New Form… ► (6) Samples ► Report_5

D.8 Labels with Multiple Columns

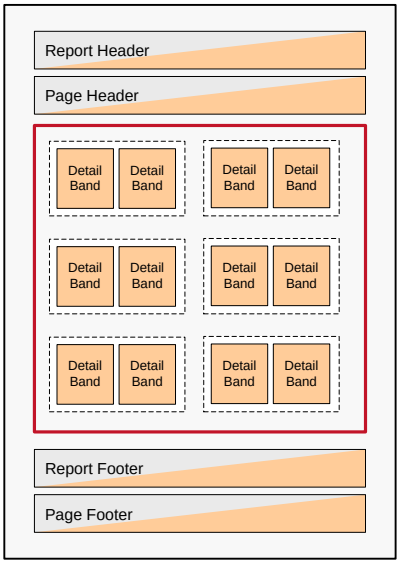


Figure 76: Labels with Multiple Columns

To print multiple columns per labels, use the following settings:

This layout is based on the *"Custom Label"* template. In the layout properties specify the row height or the number of rows (or both). In addition, make sure to remove all page and row/column breaks in the detail band.

Form:

Advanced	
Columns	(auto)
Rows	(auto)
Column Width	80.000 mm
Row Height	30.000 mm
Column Spacing	0.500 mm
Row Spacing	0.500 mm

Detail Band:

Common	
Page Break	None
Row/Column Break	None
Position	
Width	80.000 mm
Height	30.000 mm
Columns	
Columns	2
Gap	0.000 mm
Width	(auto)

Please note:
Since the number of columns is set to 2, the design on the detail band must not cover more than 50% of the detail band width (in this case 40.000 mm)!

Appendix E: Advanced Configuration

E.1 Template File Path

The file path for the template files can be set in the windows registry. You can find the appropriate settings either under HKEY_CURRENT_USER or under HKEY_LOCAL_MACHINE:

```
[HKEY_CURRENT_USER\Software\TEC-IT Datenverarbeitung GmbH\TFORMer\6.0\Directories]
TemplateDir="Some directory where the templates are stored"
```

When storing the setting under HKEY_LOCAL_MACHINE then it is valid for *all users* on the computer:

```
[HKEY_LOCAL_MACHINE\Software\TEC-IT Datenverarbeitung GmbH\TFORMer\6.0\Directories]
TemplateDir="Some directory where the templates are stored"
```

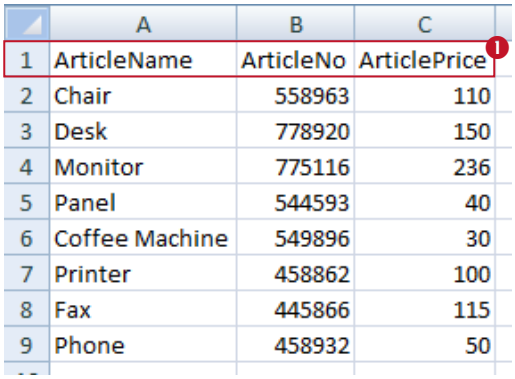
E.2 Configuration File TFORMer.xml

The **TFORMer SDK** uses a configuration file which holds basic output settings suitable for most requirements. This configuration file is named *TFORMer.xml* and is installed automatically. After installation it can be adjusted to meet customized output needs.

For details, please refer to the Developer Manual.

Appendix F: Creating CSV Files with Excel®

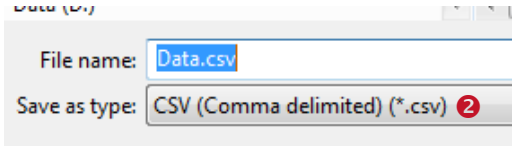
In order to create a CSV file with Microsoft® Excel®, please follow these steps:



	A	B	C
1	ArticleName	ArticleNo	ArticlePrice ❶
2	Chair	558963	110
3	Desk	778920	150
4	Monitor	775116	236
5	Panel	544593	40
6	Coffee Machine	549896	30
7	Printer	458862	100
8	Fax	445866	115
9	Phone	458932	50

First create a table with the desired data. Make sure that every column has a column name (see ❶).

The column names should match the names of the datafields used in the layout.



Then select *Office Button* ►*Save as* ►*Other Formats* from the menu.

Navigate to the desired output folder and enter a file name (e.g., “Data”). Under ❷ choose *“CSV (Comma delimited) (*.csv)”.* Then click *Save.*

```
ArticleName;ArticleNo;ArticlePrice
Chair;558963;110
Desk;778920;150
Monitor;775116;236
Panel;544593;40
Coffee Machine;549896;30
Printer;458862;100
Fax;445866;115
Phone;458932;50
```

When opened in a text editor the CSV file should look like this. All columns must be separated by semicolons.

Files

New Form ..... *Ctrl+N*
Open ..... *Ctrl+O*
Save ..... *Ctrl+S*
Close Form ..... *Ctrl+W, Ctrl+F4*

Switch between open files

Select the file from the *"Window" menu* respectively using the *layout picker*, or use the following shortcuts:

Next File ..... *Ctrl+Tab*
Previous File ..... *Ctrl+Shift+Tab*

Layout and Printing

Layout View ..... *Ctrl+L*
Data View ..... *Ctrl+D*
Preview ..... *Ctrl+Space*
Print ..... *Ctrl+P*

Editor Customization

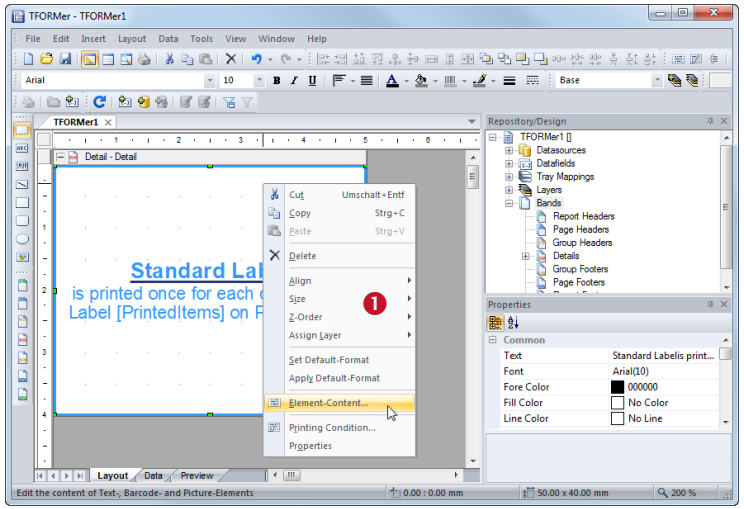
Grid Visible (on/off) ..... *Ctrl+G*
Snap To Grid (on/off) ..... *Ctrl+R*
Zoom In ..... *Ctrl++*
Zoom Out ..... *Ctrl+-*
Zoom In/Out ..... *Ctrl+Mousewheel*

Window Selection

Layout View, Data View, Preview ..... *Alt+1*
Property Window ..... *Alt+2*
Design Tree ..... *Alt+3*

Miscellaneous

TFORMer Help *F1*
Cancel operation (e.g., cancel a mouse drag and drop) ..... *ESC*



Right Click - Context Menu

If you are unsure which operations can be performed on a selected object, open the context menu by clicking the right mouse button – see ❶.

Renaming

To change the name of a *datafield, tray mapping, layer, band* or *design element,* select the entry in the tree view. Then, again, click on it with the left mouse button (or press the *F2* key). The name will become editable.

Layers

Double-click on the layer icon in the design tree to toggle the visibility.

Default Element Properties

You can save the currently used format styles (font style, color, line width, …). These styles will be applied automatically when inserting a new element. Use the context menu (right mouse button) to:

- *Set Default-Format* (based on current selection)
- *Apply Default-Format* (to current selection)

Editing

Undo ..... *Ctrl+Z, Alt+Backspace*
Redo ..... *Ctrl+Y*
Cut ..... *Ctrl+X*
Copy ..... *Ctrl+C*
Paste ..... *Ctrl+V*
Delete ..... *Del*
Select all Elements ..... *Ctrl+A*

Element content

Edit the content of selected text, barcode and picture elements *F2*

Text properties

Bold ..... *Ctrl+B*
Italic ..... *Ctrl+I*
Underline ..... *Ctrl+U*

Alignment (if two or more elements are selected)

Align Left ..... *Ctrl+←*
Align Right ..... *Ctrl+→*
Align Top ..... *Ctrl+↑*
Align Bottom ..... *Ctrl+↓*

Band Order (if a band is selected)

Move Up ..... *Alt+↑*
Move Down ..... *Alt+↓*

Drag Modifiers

While dragging elements with the mouse you can use the following modifier keys:

Snap to horizontal and vertical axis ..... *Shift*
Copy Mode ..... *Ctrl*
Fine Positioning ..... *Alt*