# Lesson 18: LSMW

## LSMW – Introduction

- *Legacy System Migration Workbench*
- *An R/3-based tool that supports when transferring data from non-SAP systems ("Legacy Systems") to SAP systems once or periodically*
- *The tool supports conversion of data of the legacy system in a convenient way.*
- *The data can then be imported into the SAP system via batch input, direct input, BAPIs or IDocs.*
- *The LSM Workbench provides a recording function that allows generating a "data migration object" in an entry or changing transaction*

## Basic Principles of the LSMW

- *The LSM Workbench was developed on the basis of the R/2- R/3 Migration Workbench*
- *LSMW was developed on the following principles*
    - *Most of the functions should reside in the SAP system. No collection of individual programs on different platforms.*
    - *The quality and consistence of the data imported into the SAP system should be more important than speed and performance of data migration.*
    - *Existing knowledge and coding should be used.*
    - *The developed "mapping" and rules should be reusable and thus be used repeatedly in projects.*

## Advantages of LSMW

- *It is a part of the SAP system and thus independent of individual platforms*
- *A variety of technical possibilities of data conversion:*
- *Data consistency due to standard import techniques*
- *Generation of the conversion program on the basis of defined rules*
- *Clear interactive process guide*
- *Interface for data in spreadsheet format*
- *Creation of data migration objects on the basis of recorded transactions*

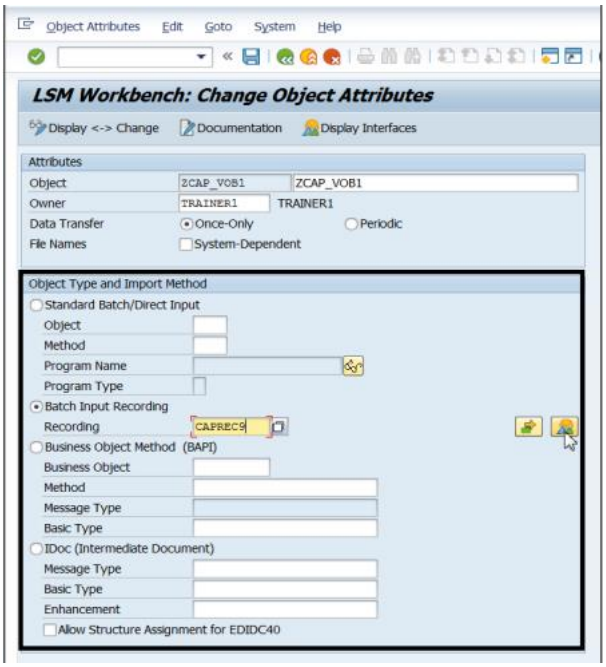
# LSMW Import Methods

***IDOC's***

***BAPI's***

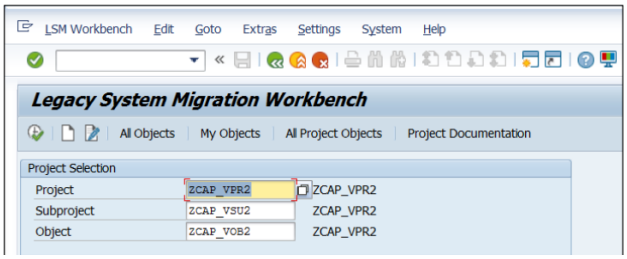
***Standard/Direct Input***

***Batch Input***



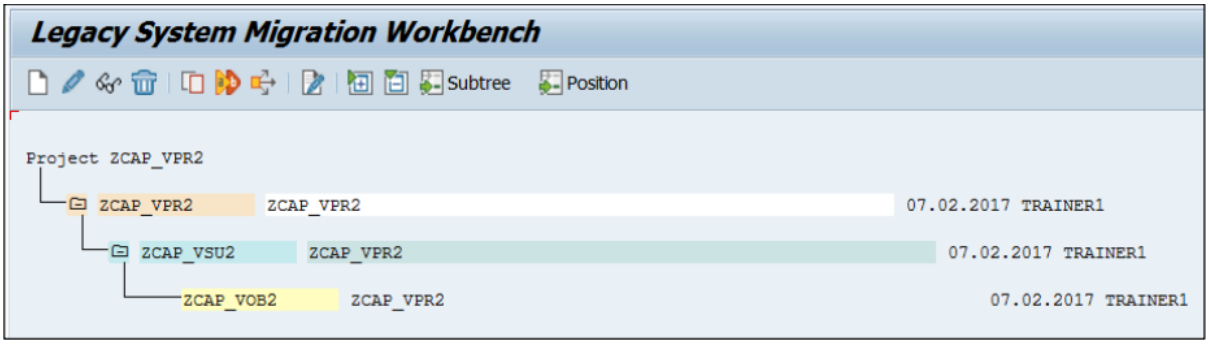

# Steps involved in LSMW

***Start Transaction Code LSMW***

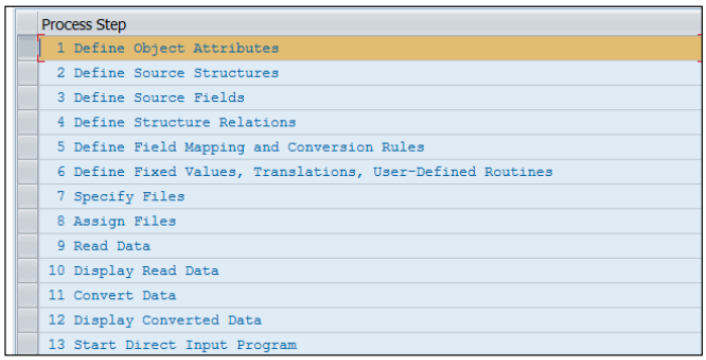



# LSMW – Procedure

- ***Project***
  - ***An ID to name the data transfer project***
- ***Sub Project***
  - ***An ID used as a further structuring attribute***
- ***Object***
  - ***An ID to name the business object***

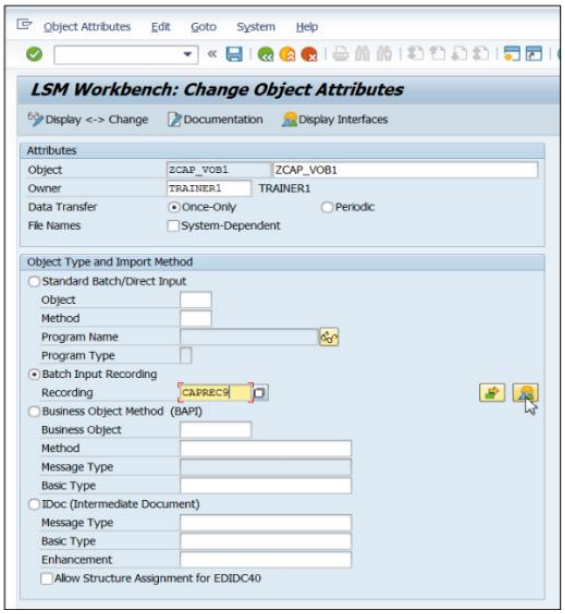


***Upon creating the project, subproject, and Objects, execute and the process steps appear as follows:***



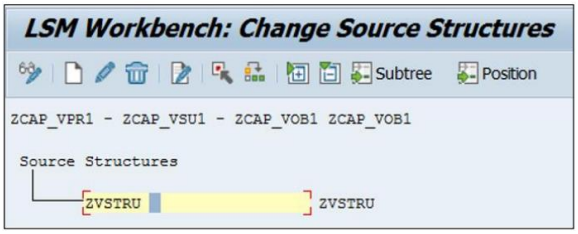

## *Define Object Attributes*

***The object type and import technique are selected***




## *Define Source Structure*

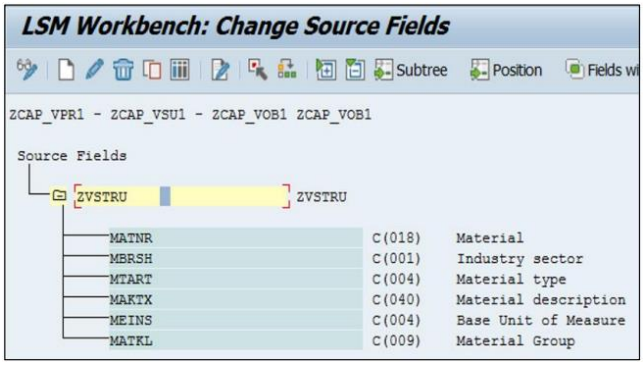
***Define the structures of the object with name, description.***




## *Define Source Fields*

***In the step 'Maintain Source Fields', fields are created and maintained for the source structure defined in the preceding step***

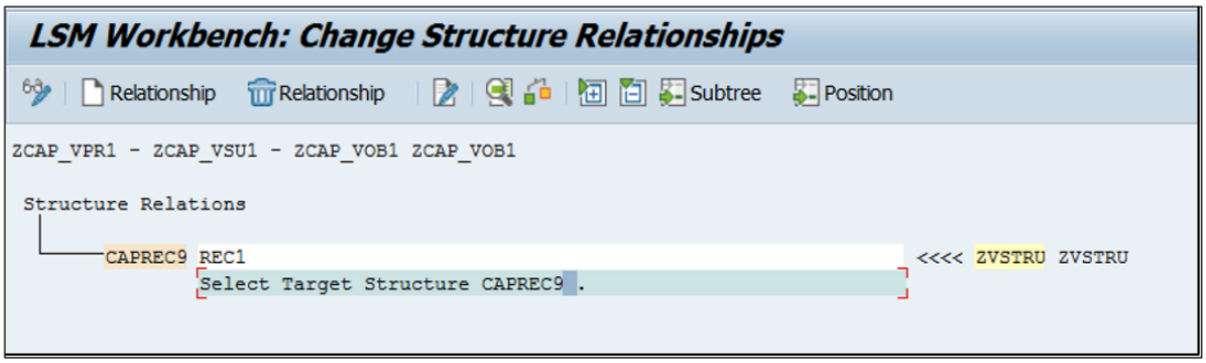
***Use source fieldnames with the same names as the target fieldnames as much as possible, because it allows you to use the 'auto - field mapping' function in step***



*'Maintain field mapping and conversion rules'.*

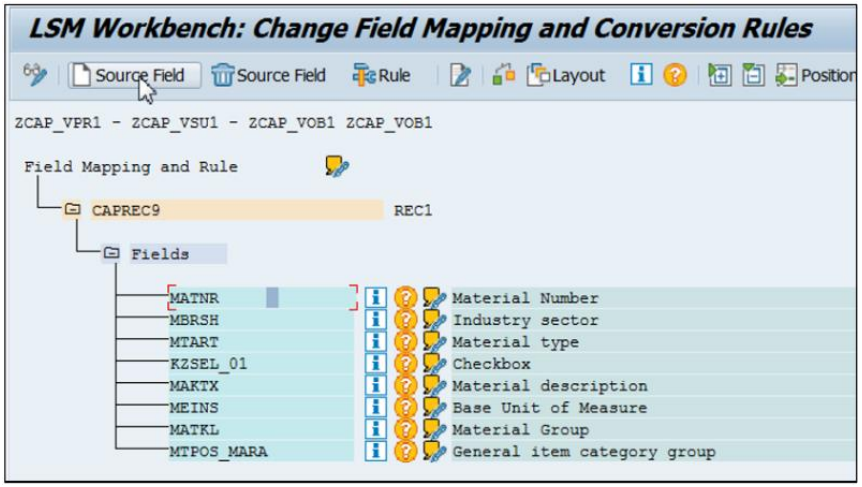
## *Define Structure Relationships*

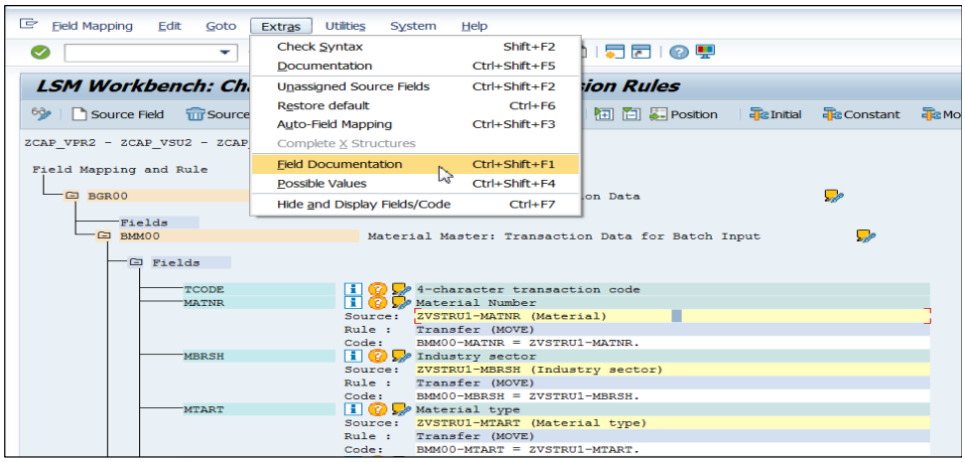
- ***The structural relationships define the relationships between source and target structures.***
- ***Since there is only one source and target, the relationship is maintained***




## *Maintain Field Mapping and Conversion Rules*

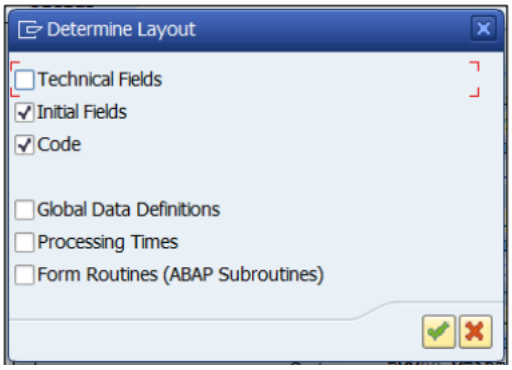
- ***Assign source fields to target fields and define how the field contents will be converted.***
- ***In the step 'Maintain Field Mapping and Conversion Rules', you assign source fields to target fields and define how the field contents will be converted***
- ***All fields of target structure, which you selected in the previous step, will be displayed.***
- ***To assign a source field, position the cursor on a target field in the tree structure and select Assign source field***
- ***This displays a list of all available source fields for selection. You can assign the fields by double-clicking on them as well***






***Layout determination***

❖ 

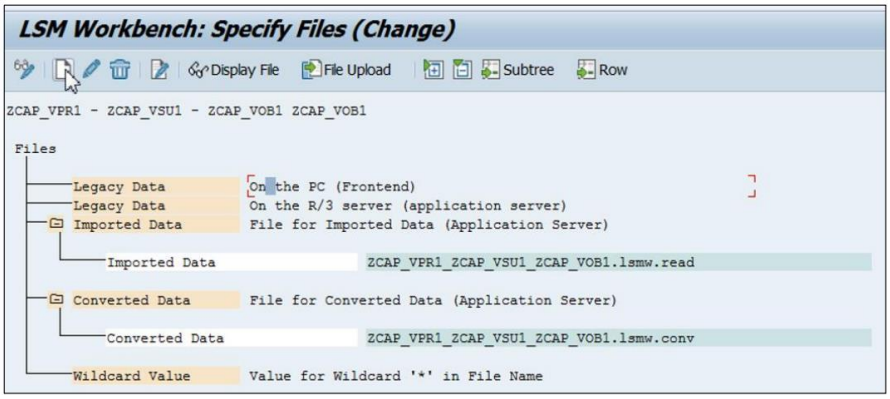




## *Maintain Fixed Values, Translations and User written Routines*

- ***Fixed value: Here you can specify length, type, flag for lowercase/uppercase and value in addition to the description.***
- ***Translation values: Here you specify the value table to be used during translation. The values can be uploaded from a PC file.***
- ***Process the reusable rules of a project***

## *Specify Files*

***This step describes all files to be used in the following steps:***




## *Assign Files*

***Assign defined files to the source structures***

## *Read Data*

- ***Can display all or a part of the read data in table form.***
- ***To process all data belonging to an object, click on Execute.***
- ***To migrate a part of the data only, limit the number of data to be migrated in field "General selection parameters". Make the selection in field "Transaction number" from "... to ...". Multiple selection is possible.***
- ***In addition, two check boxes are offered:***
  - ***Amount field: Amount fields are converted into calculation format (with decimal point).***
  - ***Date field: Date fields are converted into internal format (YYYYMMDD).***

## *Display Read Data*

- ***Display all or a part of the read data in table form.***
- ***Clicking on a line displays all information for this line in a clear way.***
- ***Change display allows to select either a one-line or multi-line view.***
- ***Display colour palette displays the colours for the individual hierarchy levels.***

## *Convert Data*

- ***With regard to operation, this work step corresponds to work step "Read Data".***
- ***If data selection is not to be made, confirm the process by clicking on Execute. Otherwise, make the selection in field "Transaction number" from "...to...". Here, multiple selection of transaction numbers is possible as well.***
- ***If one or several source fields are marked as selection parameters when defining the source fields, these fields are also offered as selection parameters.***

## *Display Converted Data*

***The display the data that is converted.***

## *Generate Batch Input Session*

- ***The standard batch input program belonging to the object is directly called.***

- ***The name of the file with the converted data is already proposed.***
- ***The batch input sessions to be generated are named after the LSMW object.***

## *LSMW – Procedure*

- ***Run Batch Input Session***
  - ***The program goes to SAP standard transaction SM35.***
  - ***Follow the procedure to run the session (which is already discussed in the Session method)***
- ***Import Data with Direct Input***
  - ***Depending on the object type, either the standard direct input program belonging to the object is called or select a direct input program or a direct input transaction.***
- ***Start Direct Input Session***
  - ***Depending on the object type, either the standard direct input program belonging to the object is called or a direct input program can be selected or a direct input transaction.***

## *Review Question*

***Question 1: In the specify file step of LSMW, files can only be selected from the Application Server.***
- ***False***

***Question 2: In Standard/Direct Input method of LSMW all fields of a transaction are always available for reprocessing.***

---

# *Lesson 19: Smart Forms*

---

## *Smart Forms-Introduction*

- ***Tool to create and maintain forms with minimal programming effort***
- ***Easy to create the form without much programming knowledge***
- ***Allows to execute simple modifications to the form and in the form logic by using graphical tools***

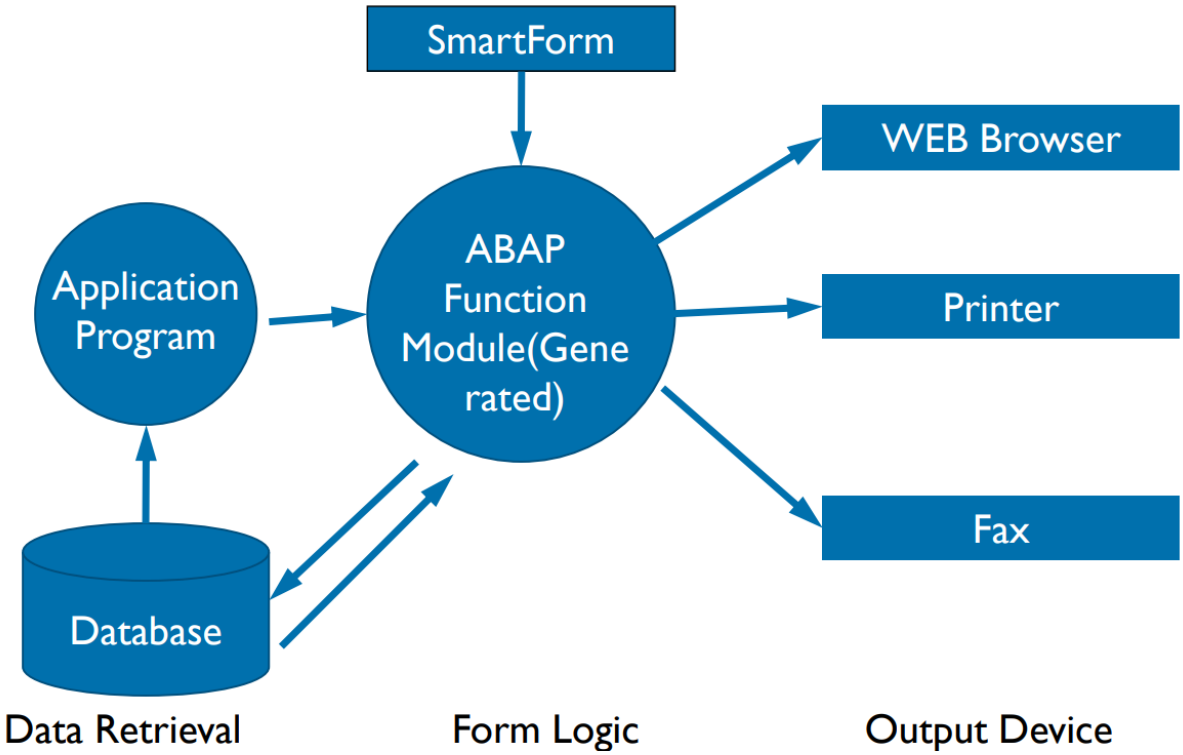
## *Basic Features of Smart Forms*

- ***Data retrieval and form logic are separated from each other.***
- ***Application program passes data to Smart forms through Function module interface which is generated automatically on Smart forms activation.***
- ***Reduces the implementation cost.***

## *Key Benefits of Smart Forms*

- ***Less Programming Efforts***
- ***Output of background graphics, for form design***
- ***Coloured output of texts***
- ***User-friendly and integrated Form Painter for the graphical design of forms.***
- ***Graphical Table Painter for drawing tables***

## *Smart Forms Architecture*



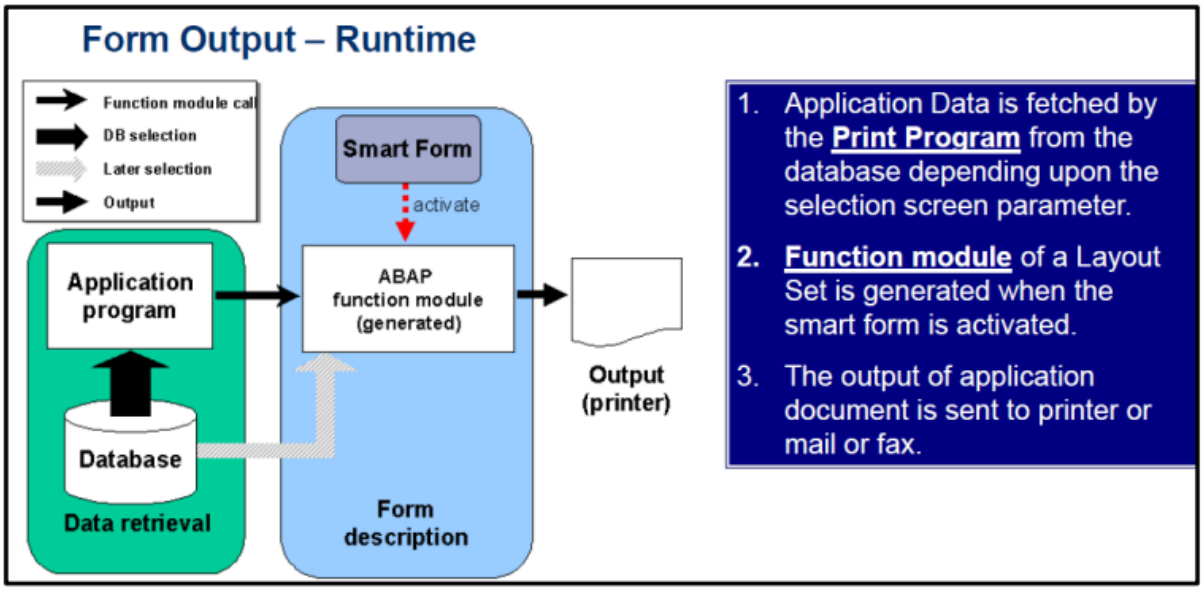

## *Smart forms Architecture - Structure*

- ***A Smart form has the following attributes:***
  - ***Layout: In the layout, you define how the output data is positioned, its appearance in graphics, and the design of the pages.***
  - ***Form logic: control the flow of the form output.***
  - ***Form interface to transfer application data to the form definition***
- ***Transactions***

| SMARTFORMS | Create Smartform |
|---|---|
| SMARTSTYLES | Create style |
| SO10 | Create standard Text |
| SE38 | Create print program |
| SE78 | Upload Logo |

## *Architecture - Form Output Runtime*

***The following graphics show you the architecture that is implemented when you create and print a Smart form.***

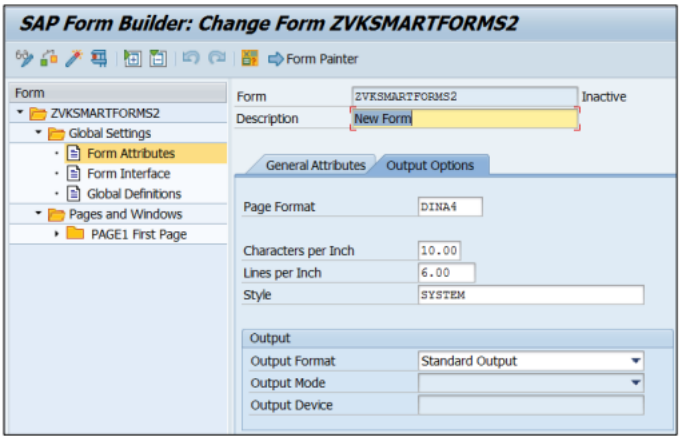



## *Structure of Smart Forms*

- ***SMARTFORMS provides a graphical user interface which is divided into three different parts:***
  - ***Navigation Panel***
  - ***PC Editor***
  - ***Form Painter***

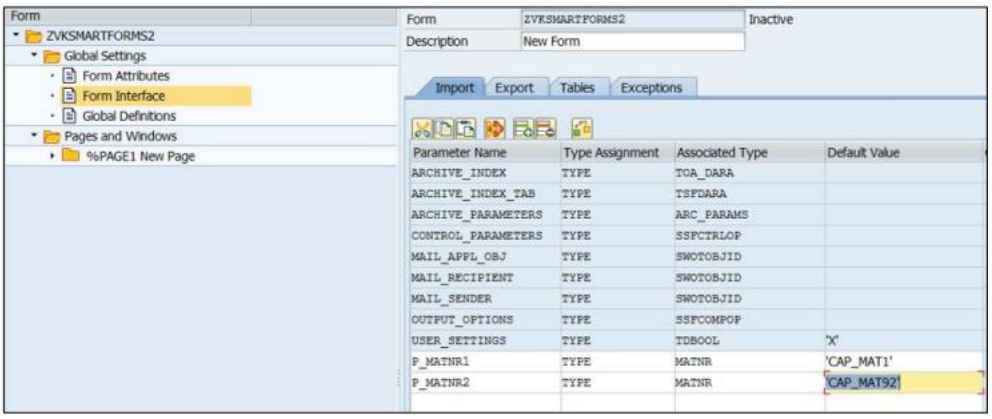
## *Smart Forms - Navigation Panel*

- ***Global Settings Has 3 sections***
  - ***Form Attributes***
    - ***Define settings necessary for printing like page format, characters per inch, flag for XSF output etc***



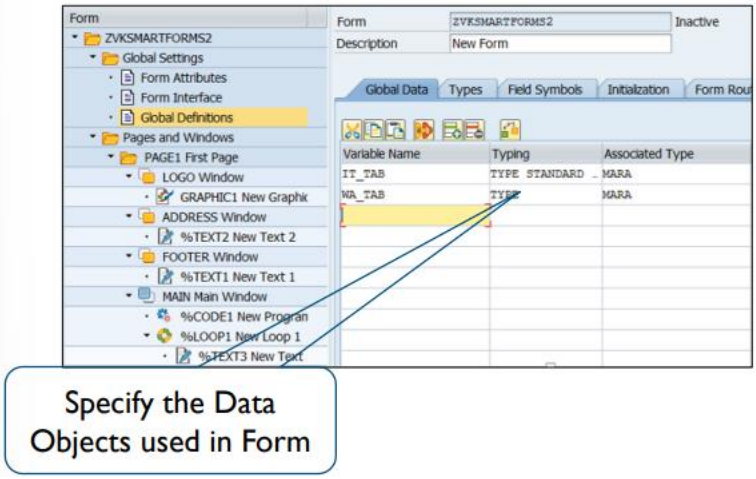

## *Navigation Panel - Global Settings*

- ***Form Interface***
  - ***To transfer application data to form definition***
  - ***The IMPORT EXPORT parameters and INTERNAL TABLES are declared***
  - ***Exception handling is also taken care.***




## Navigation Panel - Global Settings (Contd.).

- *Global Definition*
- *Allows user to declare variables which can be used on global scope*




## Pages and Windows

- *Provides list of all components of form*
- *All the basic elements are maintained under this node*
- *Pages*
- *Windows*
- *Graphic*
- *Address*

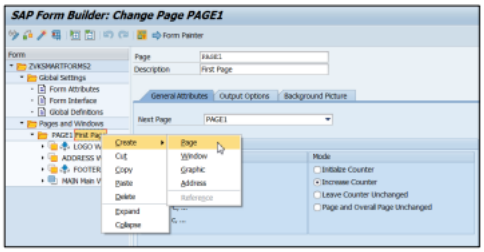
## Pages

- *Each form consists of one or more pages*
- *The first page in the tree structure is the start page*
- *The page layout includes the page format and the position of windows on a page*

## Procedure for Creating Page

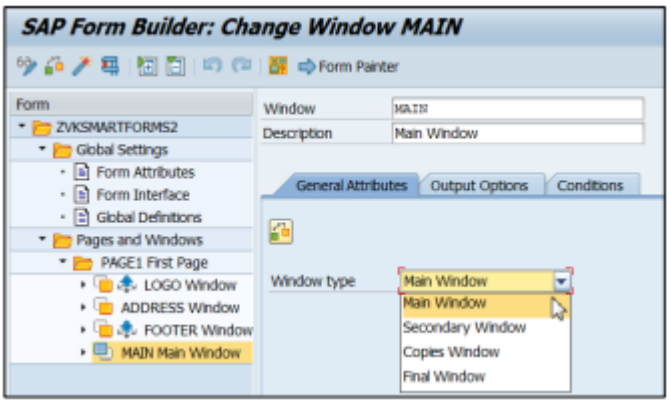
- *Select an existing page node to position the new page node*

- *Create a new page node in the navigation tree of the Form Builder*
- *Name and description has to be specified*
- *The format and mode of the page counter has to be specified on General Attributes tab*

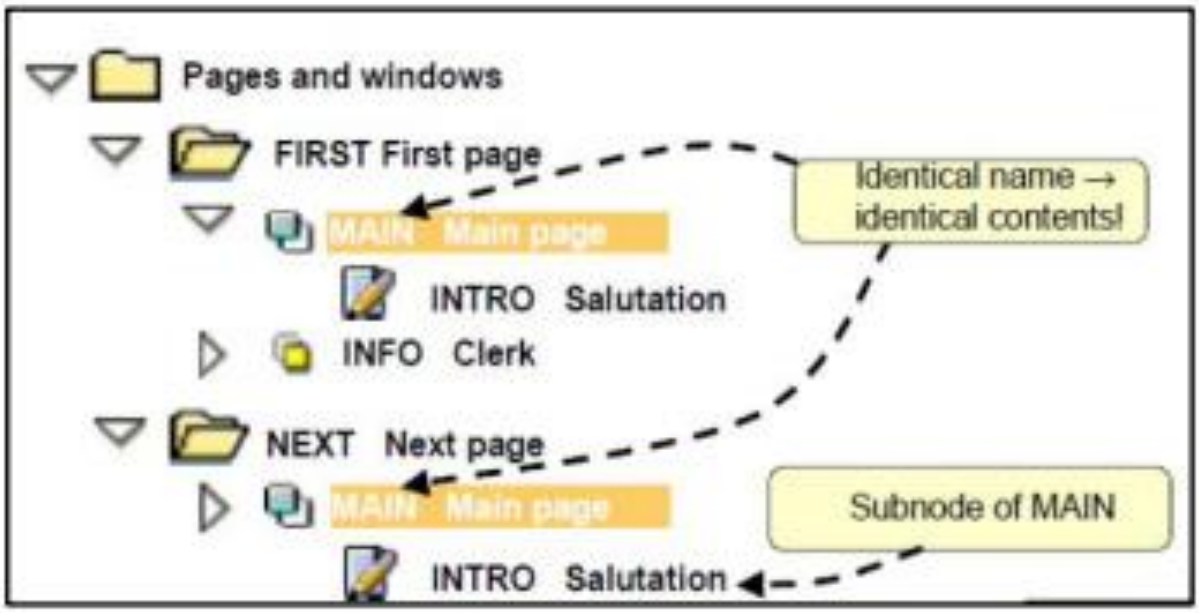



## Windows

- *Output areas for all output data*
- *Size and position are set in the Form Painter*
- *Following are the Window Types in Smart forms*
- *Main window*
- *Secondary window*
- *Copies Window*
- *Final Window*




## Form Builder - Structuring Pages



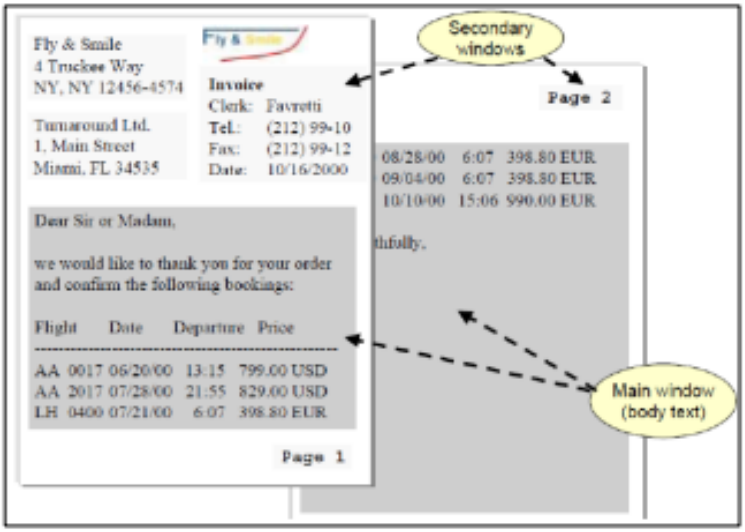

## Main Window

- *Display the text and data that runs in to several pages*
- *It automatically triggers the page break*
- *Only one window in a form is main window*
- *The main window must have the same width on each page*
- *A page without main window must not call itself as next page, since this would trigger an endless loop*

## Secondary-Windows

- ***Text and data displayed in a predetermined output area.***
- ***Text and data that do not fit into the secondary window are truncated and not displayed.***

## *Form Builder - Main and Secondary Windows*




## *Copies-Window*

- ***The content will appear either in the copy form or original form***
- ***This is used for printing the copies of the form***

## *Final-Window*

- ***Final window is used to display values which are processed in the initial pages***

## *Texts and Data in a Form*

- ***Texts and data are entered using PC Editor***
- ***Various Operations performed in the PC Editor***
- ***Tables or templates can be used to display texts and data in table format***

## *Positioning of Texts on the Form*

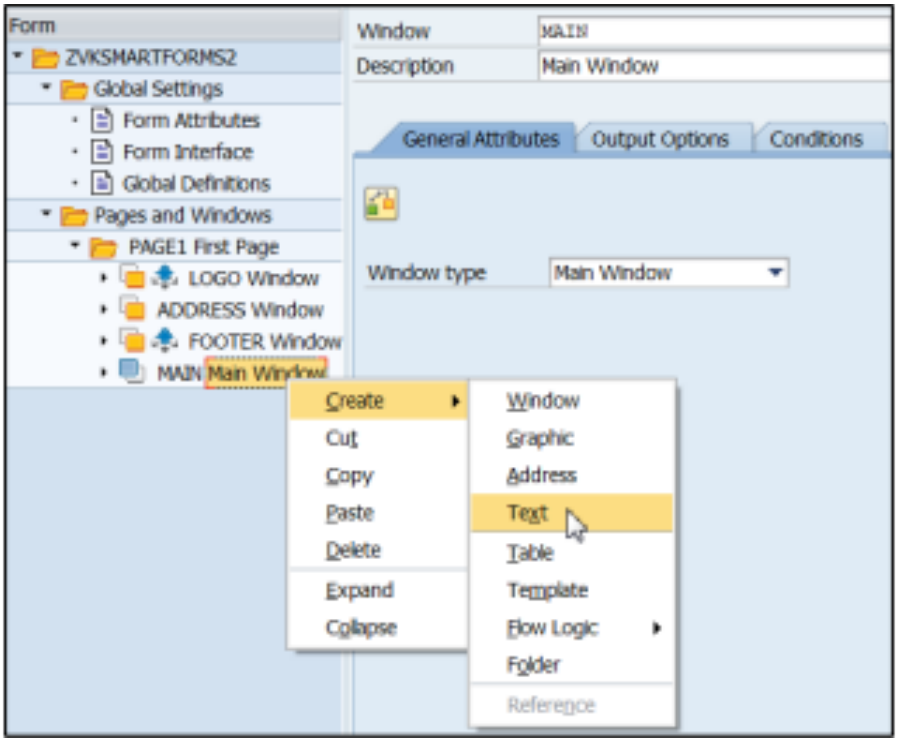
- ***All the texts in the form are displayed using text nodes***
- ***The only exception is addresses, which are displayed using their own node***

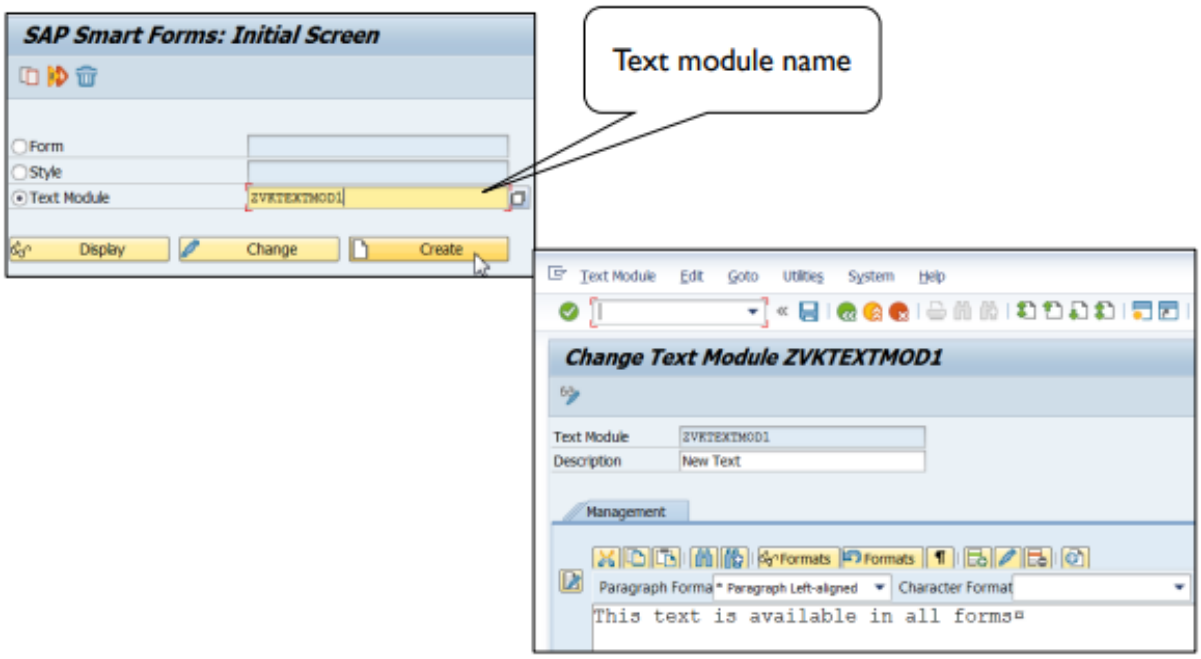
## *Entering Texts in PC Editor*

- ***New texts are entered in PC Editor***
- ***The system fields and the user-defined fields are used to include data from form interface***
- ***These fields are replaced with values when the form is processed***

## *Creating Text*

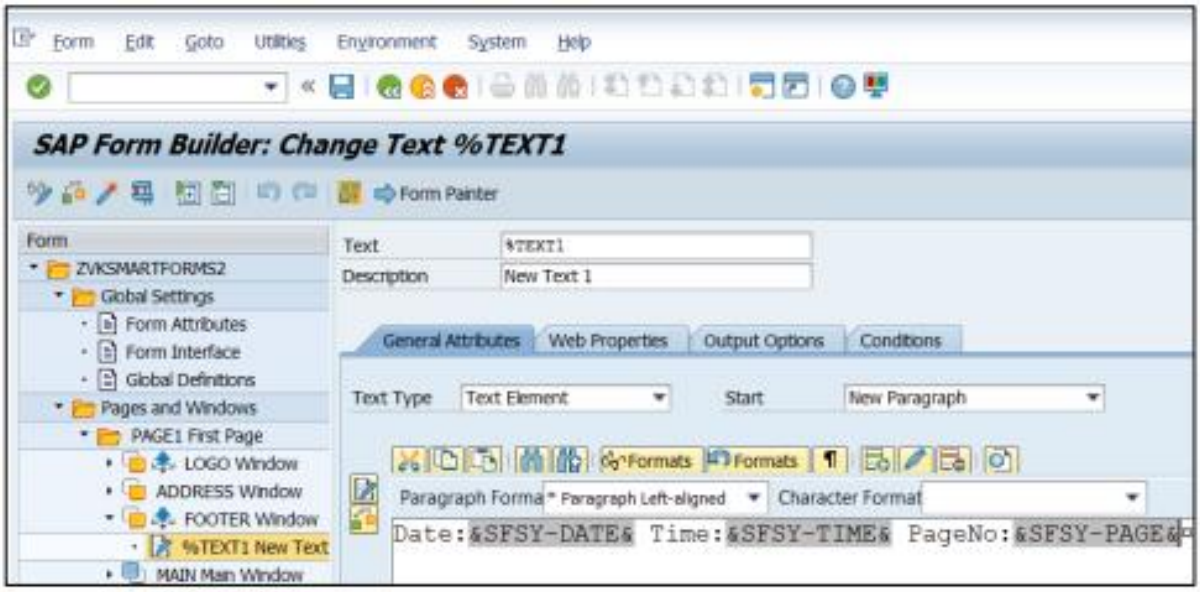
- ***Create a text node in the navigation tree of the Form Builder.***




## *Text Element in Form*

- ***Enter a unique name for the node and a node description.***
- ***On the General Attributes tab choose Text Element as text.***

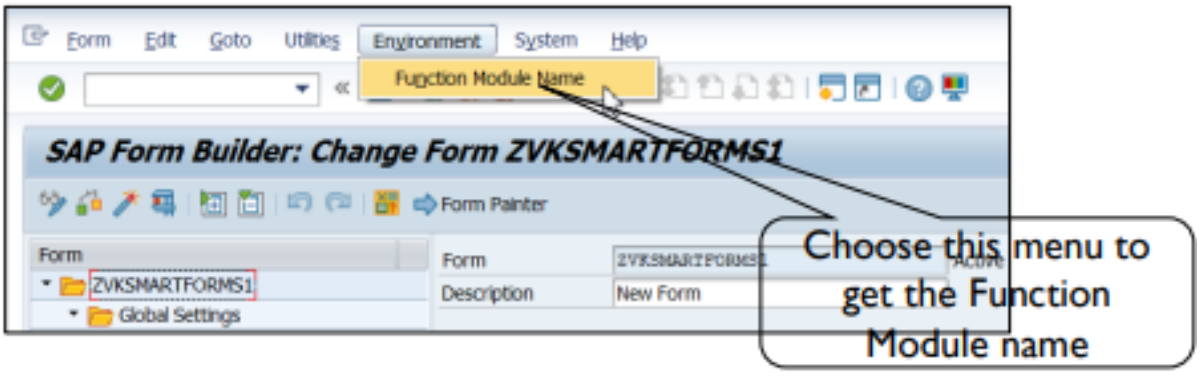



## *Integrating the Smart Form into the Application*

- ***Form printing triggered by calling function modules***
- ***Name of the form determines the name of the generated function module.***
- ***The name of the generated function module is unique only within one system.***

## *Procedure*

- ***In the Form Builder call the function Environment Name of the function module and copy its name.***

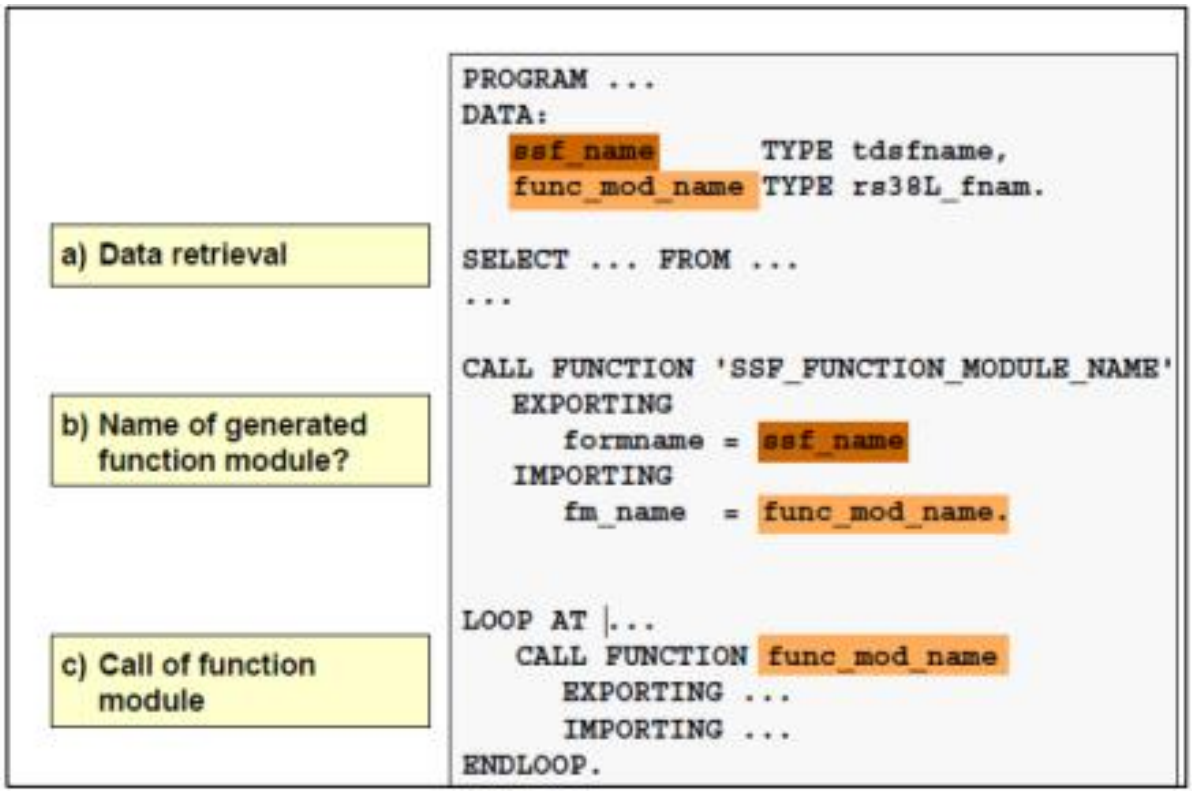




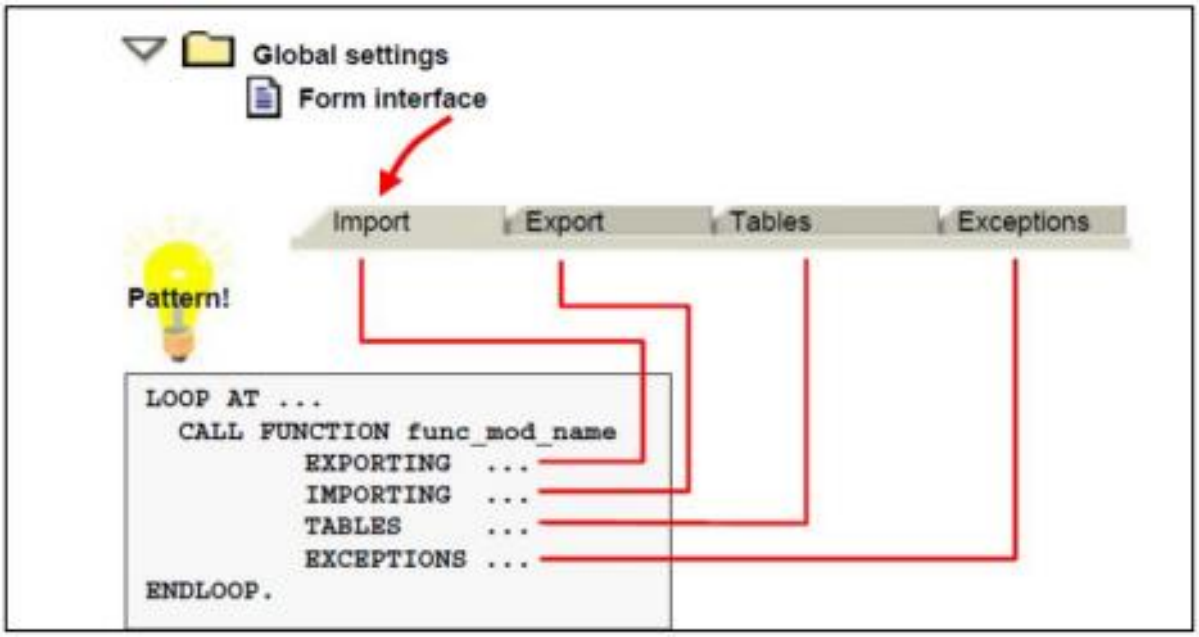
- ***In the application program define a variable of type RS38L_FNAM for the name of the generated function module:***
  - ***Data fm_name type RS38L_FNAM.***
- ***The Smart Form can be called in other parts of the application program as well.***
- ***Function module SSF_FIELD_LIST - to list form parameters***

❖ *Function module SSF_FUNCTION_MODULE_NAME - returns the name of generated function module*

## *Integration in ABAP programs*




## *Integration into Application Programs – Generated Function Module*




## *Integration into Application Programs – Control Structure CONTROL_PARAMETERS*

**CONTROL_PARAMETERS**
**(Export parameters of the function module generated)**
**Type:** ssfctrlop

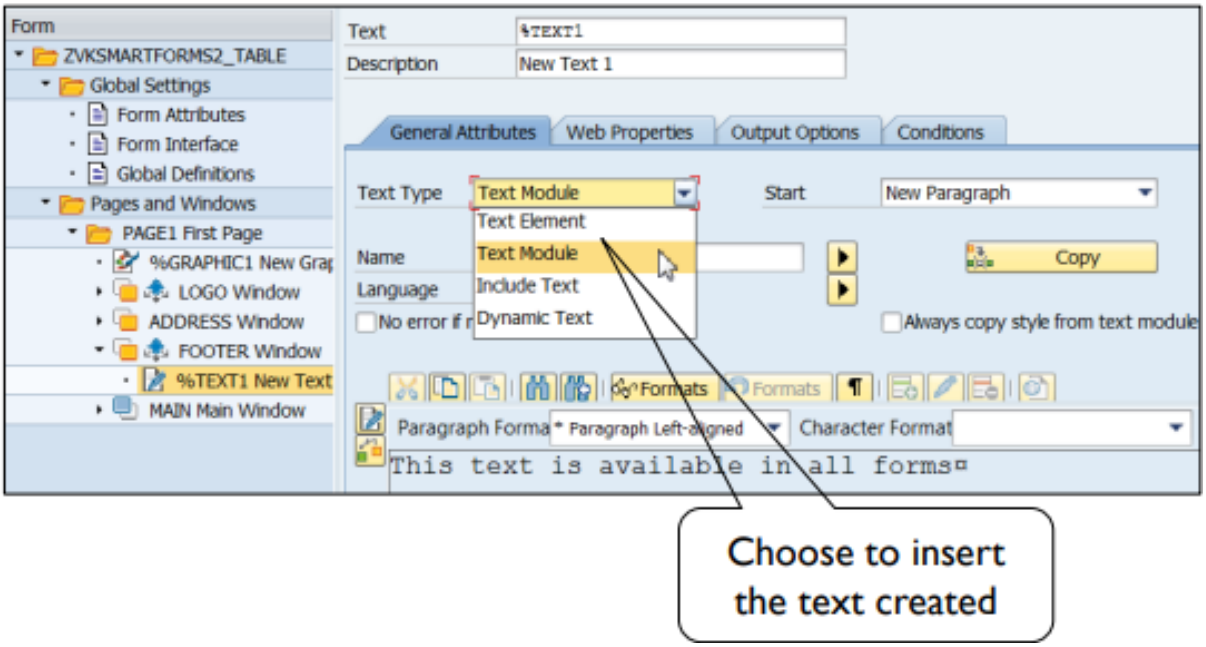
| | |
|---|---|
| no_open | No new spool request |
| no_close | Do not close spool request |
| device | Output device ('PRINTER', 'TELEFAX', 'MAIL') |
| no_dialog | No dialog box for output |
| preview | Print preview |
| langu | Language |
| startpage | Start page ≠ default |

## *Text Modules*

- ***Text modules are used to centrally store texts that are used frequently in forms in the system.***
- ***Text modules are included in forms using texts nodes***
- ***Allows easy use of text from a text module in several forms***
- ***It Can be used across clients***

## *Include Text Module in Form*

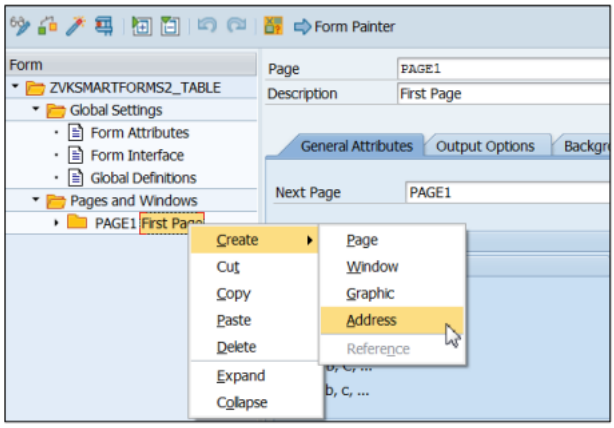
- ***Create a text node in the navigation tree***
- ***In General attributes tab change the type as 'TEXT MODULE'***
- ***Change text name to the name of the text module***



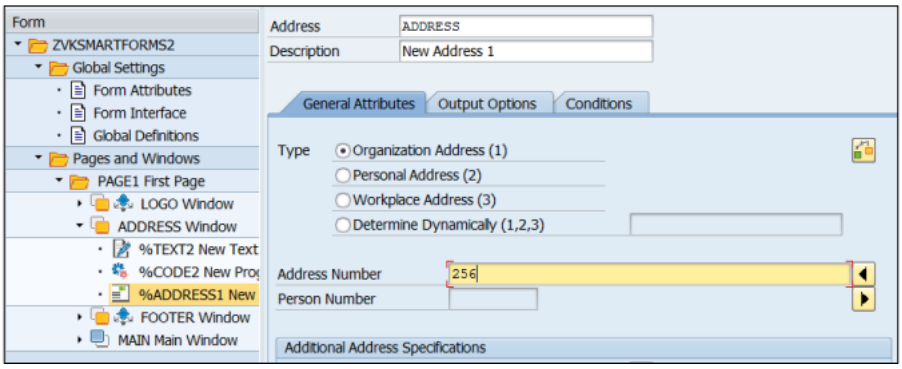

## *Inserting Addresses*

- ***Administered using the Business Address Services (BAS)***
- ***According to the postal regulations of the sender country, the address is formatted***
- ***Three address types***
    - ***Company addresses (address type 1)***
    - ***Personal addresses (address type 2)***
    - ***Workplace addresses (address type 3)***

## *Creating Address-type*




## *Address*

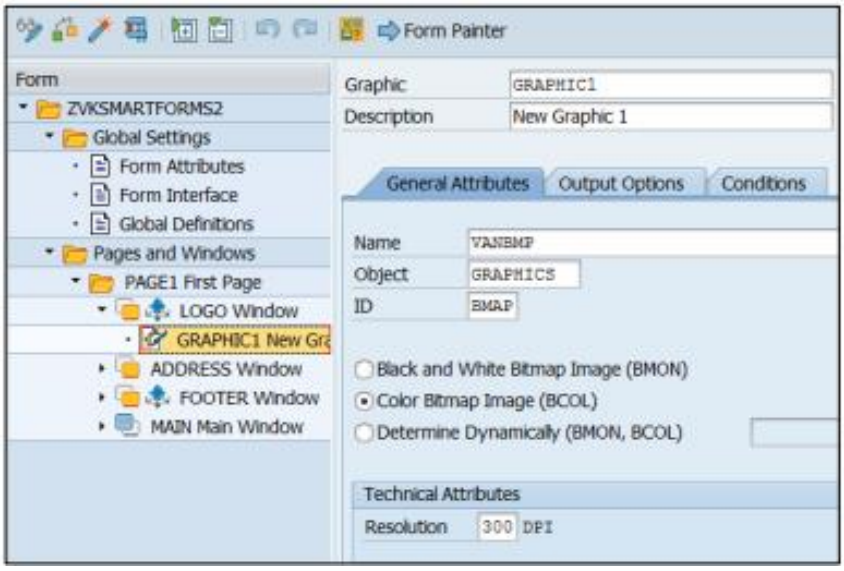



## *Graphics In Smartform*

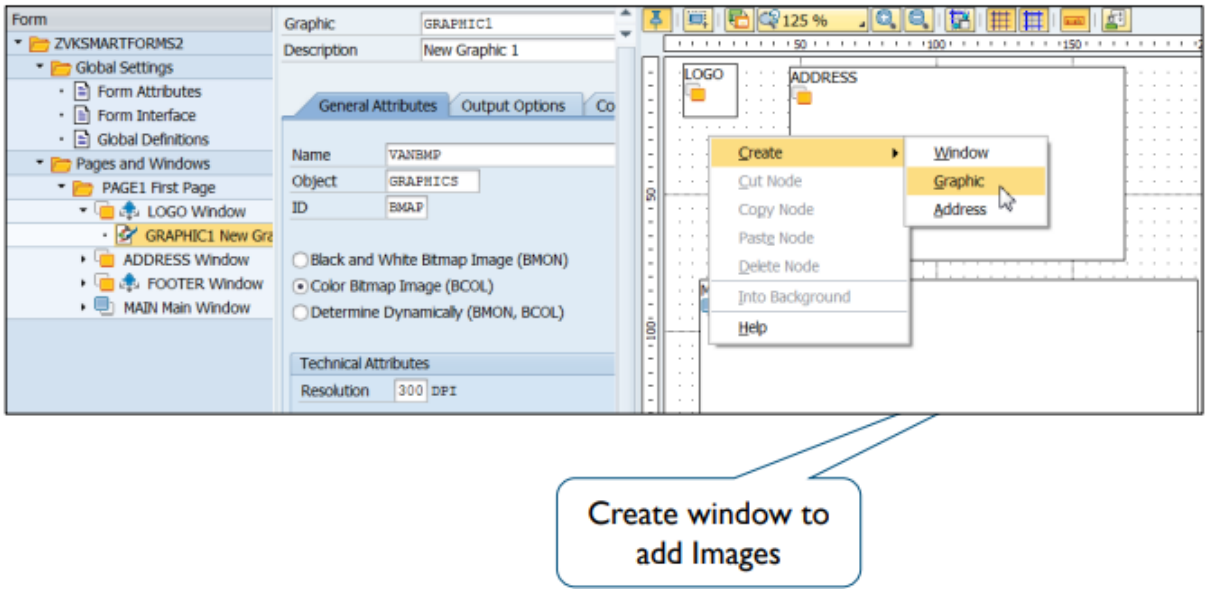
- ***To import, administer and transport images or graphics***
- ***They can be incorporated statically into a form or include them dynamically using an appropriate field***
- ***Images can be included in background as well***

## *Graphics*

- ****.BMP and *.TIF files can be imported and used in forms.***
- ***SE78 - Graphic administration.***




## *Creating Graphics in Smart Forms*




## *Working with Tables*

- ***To display or print contents in tabular form***
- ***Node types***
  - ***Template node***
    - ***Static - The number of columns and lines are determined before the actual output***
  - ***Table node***
    - ***Dynamic - Table size depends on the amount of data selected at runtime***
- ***Line Types Specifies***
  - ***Width of table line***
  - ***The layout of both node types***
  - ***Also, the width of the individual cells within the table line***

## *Printing tables*

- ***The table can be designed independent of the number of lines***
- ***The size of the table depends on how much data the application program passes to the form at runtime***
- ***Tables cannot be nested.***
- ***Output of a table can be divided into***
    - ***Header***
    - ***main area***
    - ***footer***

## *Accessing Application Data*

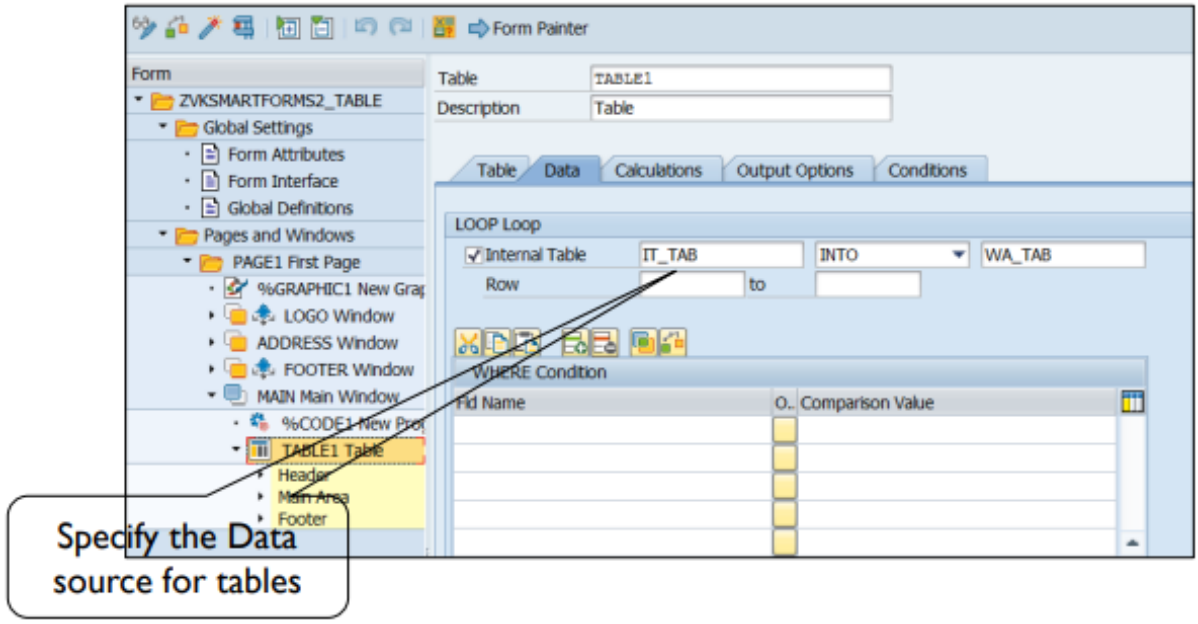
- ***The application program reads the data to be displayed on the form as a table from the database and writes it into an internal table***
- ***When calling the Smart Form, this internal table is passed to the form interface to access it within the form description***
- ***Access the internal table to display it on the form line by line***
- ***To accessing several internal table for table output whose entries are interdependent combine loop and table nodes***

## *Reading Internal-Tables*

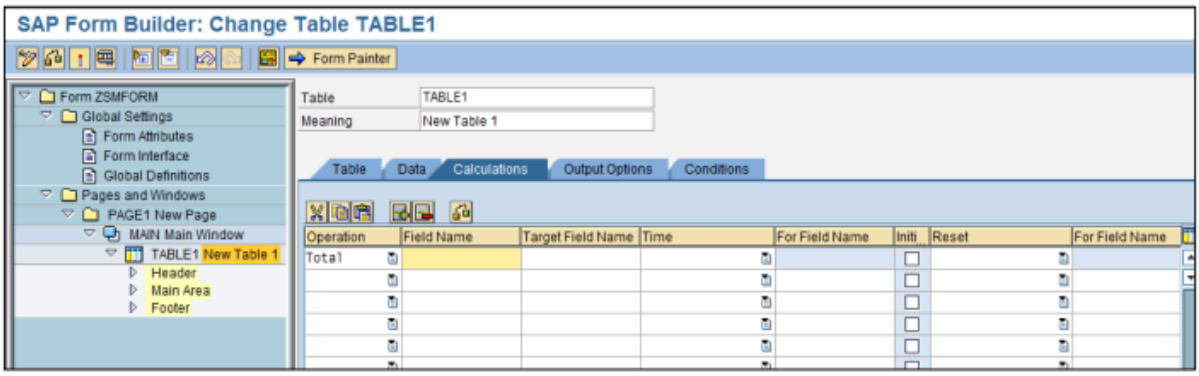
- ***The table is printed in the form line by line as the number of selected entries in the internal table differs***
- ***The table node defines a table layout***
- ***The Data tab is used to access internal tables, which exists for loop nodes and for table nodes***

## *Procedure to read Internal Tables*

- ***Create a work area for the internal table in the global definitions***
- ***Go to Data tab of the loop or table node and mark Internal Table else the loop is deactivated***
- ***Enter the name of the internal table that is passed at the form interface***
- ***Enter the assignment type (INTO or ASSIGNING) and a work area (structure with the same type as the table line or the field symbol)***
    - ***If a table with header line is used as work area specify the internal table name again***
    - ***If desired, use the input fields Line and To to limit the lines of the internal table that is to be read***
- ***Use the group box WHERE Condition to select a particular part of the data in the internal table***

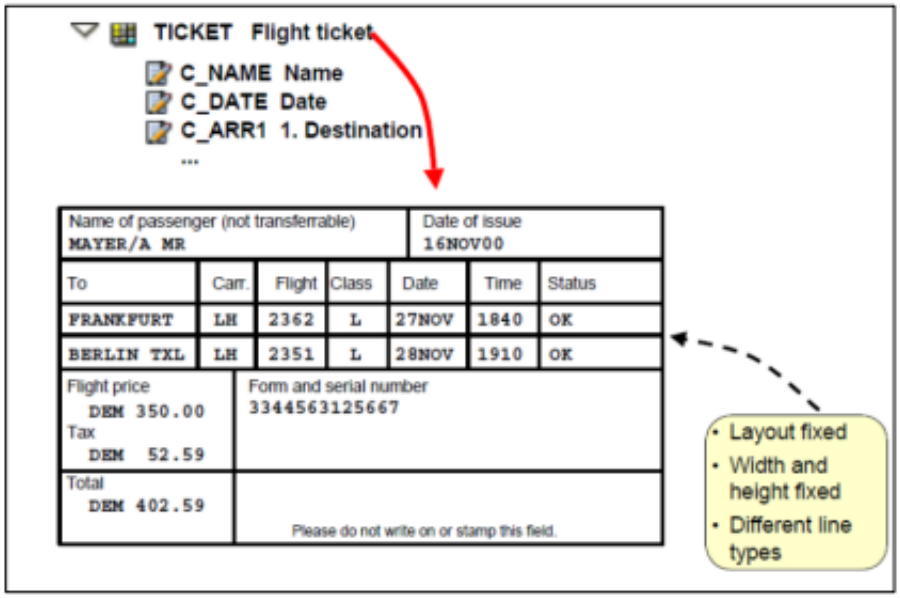



## *Table Calculations – Procedure*

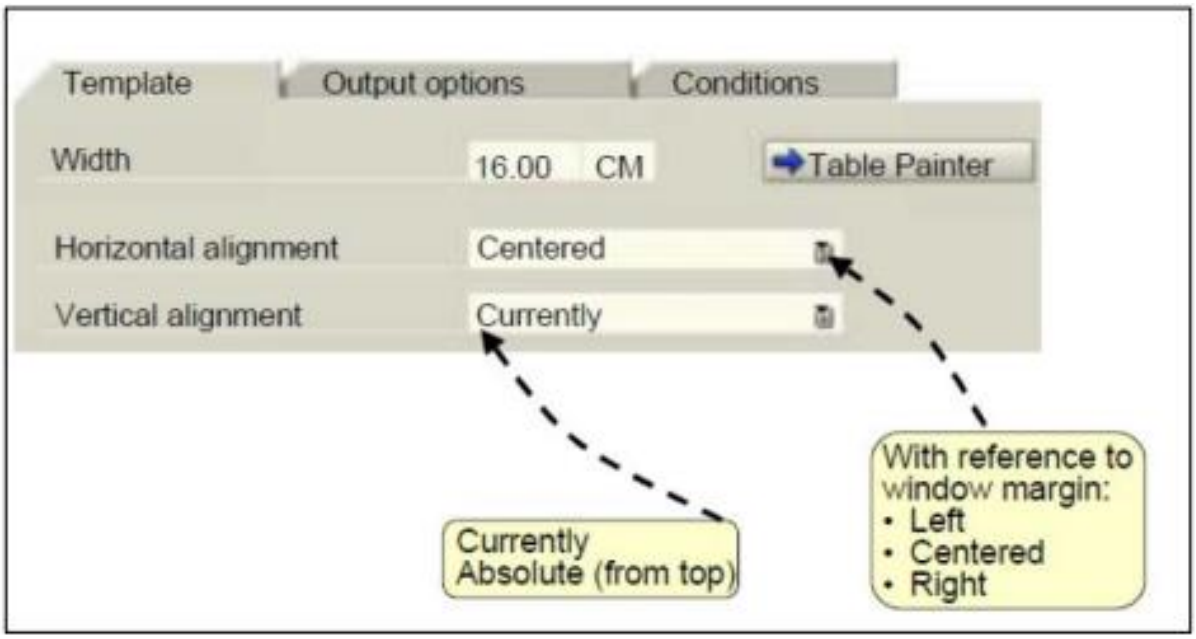


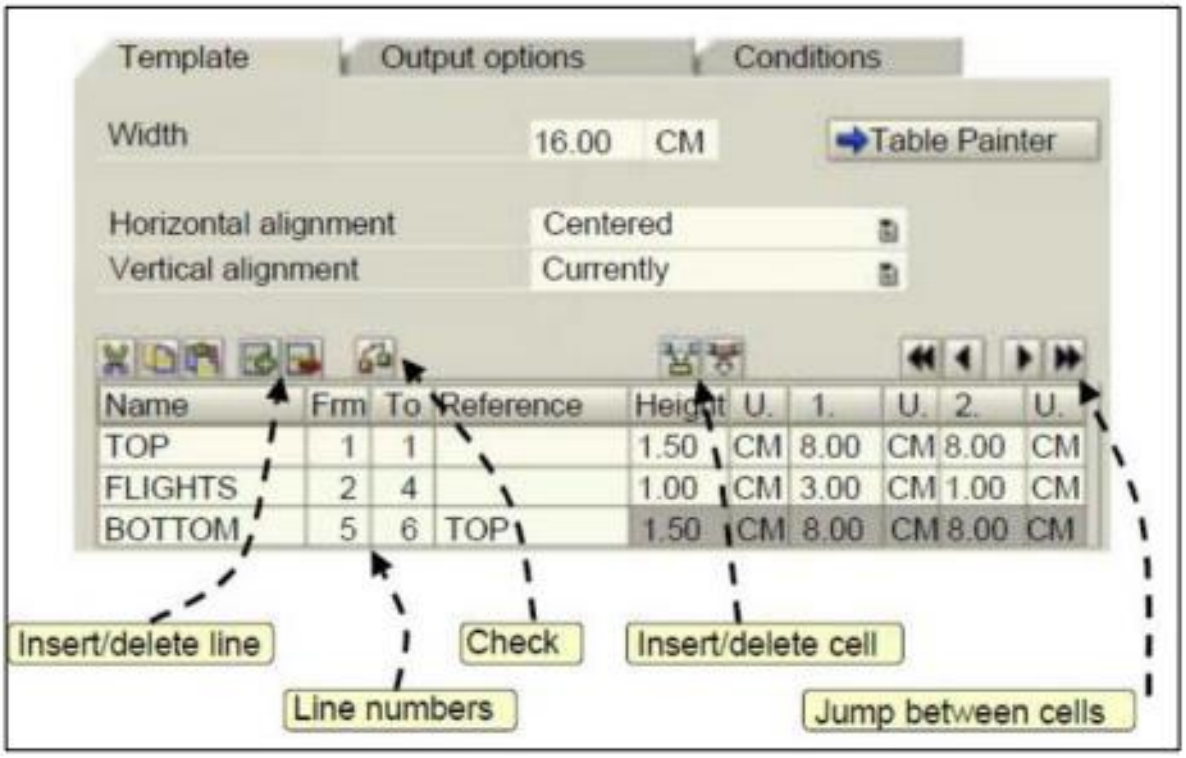

## *Template*

- ***You use the Template node type to output tables with a fixed layout and size.***
- ***Templates cannot be nested***

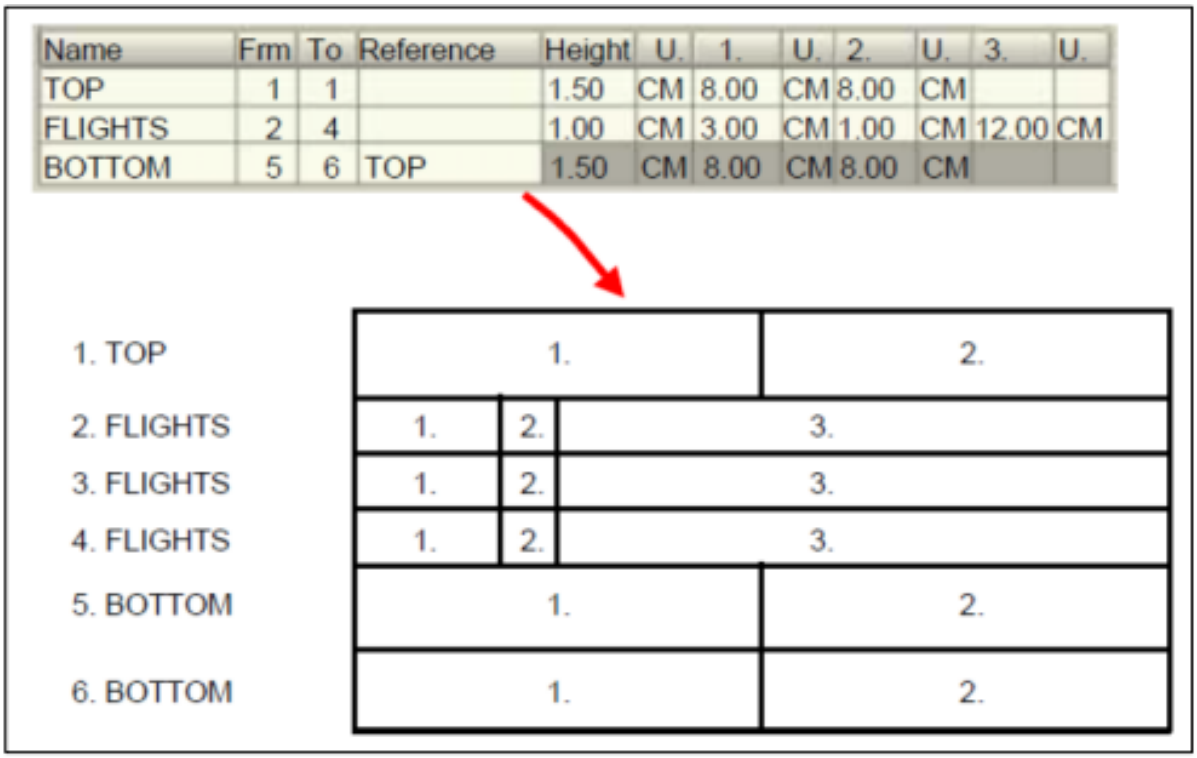


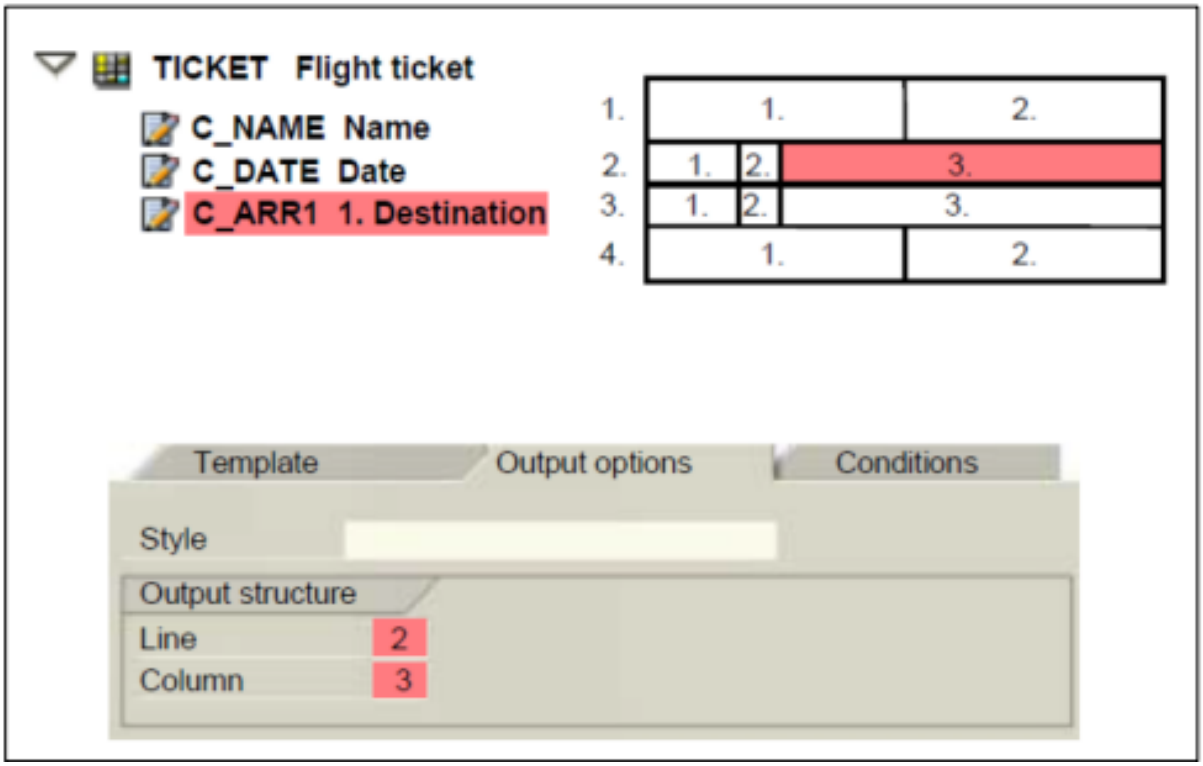
# *Template Layout*



| Name | Frm | To | Reference | Height | U. | 1. | U. | 2. | U. | 3. | U. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| TOP | 1 | 1 | | 1.50 | CM | 8.00 | CM | 8.00 | CM | | |
| FLIGHTS | 2 | 4 | | 1.00 | CM | 3.00 | CM | 1.00 | CM | 12.00 | CM |
| BOTTOM | 5 | 6 | TOP | 1.50 | CM | 8.00 | CM | 8.00 | CM | | |




## *Dynamic Page-Break*

- ***Page break triggered when the main window of a page is full***
- ***Only the contents of main window can spread over several pages***

## *Page-Numbering*

- ***&SFSY-PAGE&***

- *Specifies current page number*

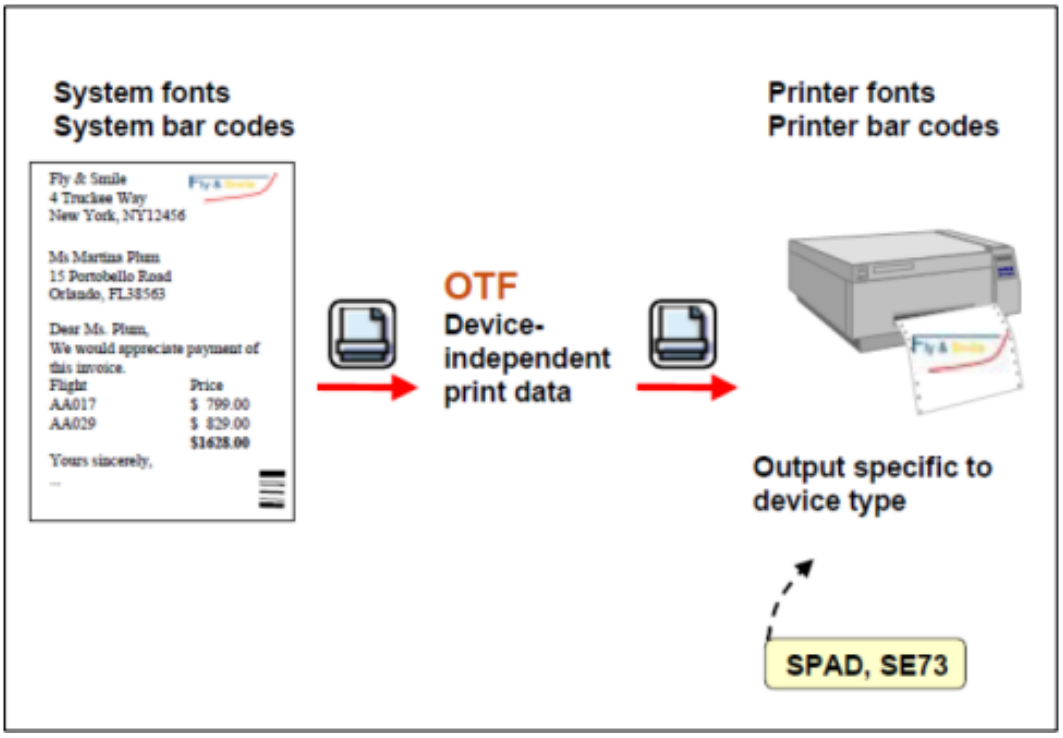
❖ ***&SFSY-FORMPAGES&***

- *Specifies total number of pages in the form*

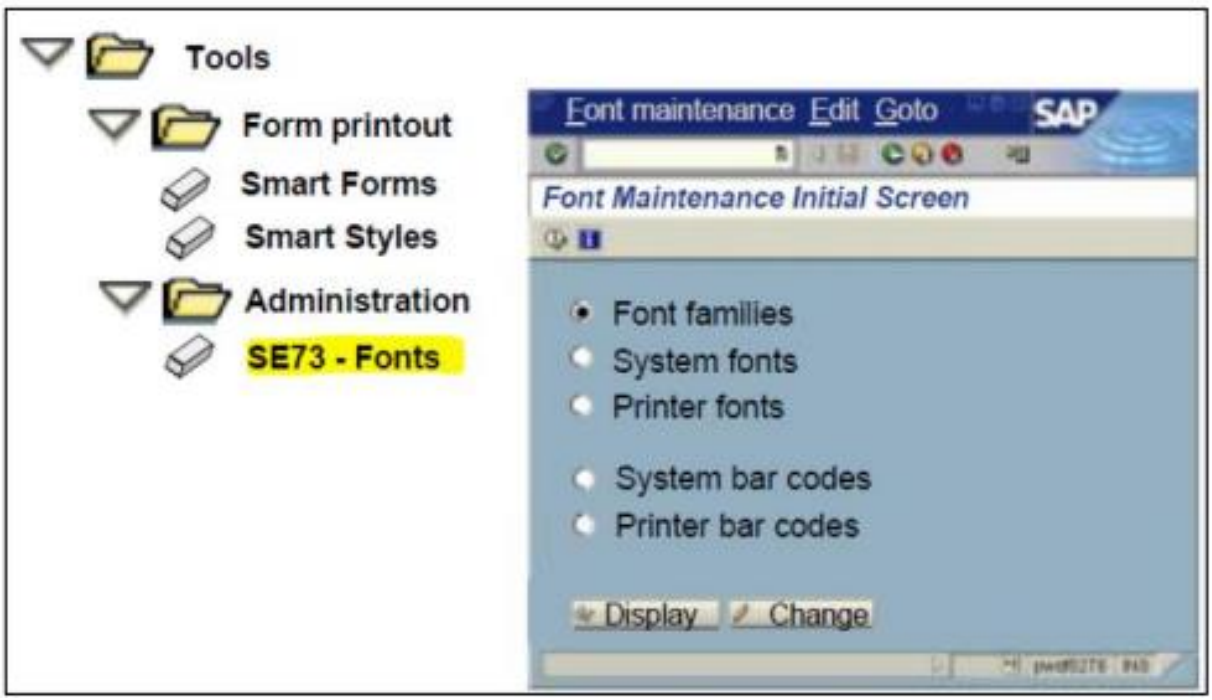
❖ ***&SFSY-JOBPAGE&***

- *Specifies total number of pages in all forms in the print job*

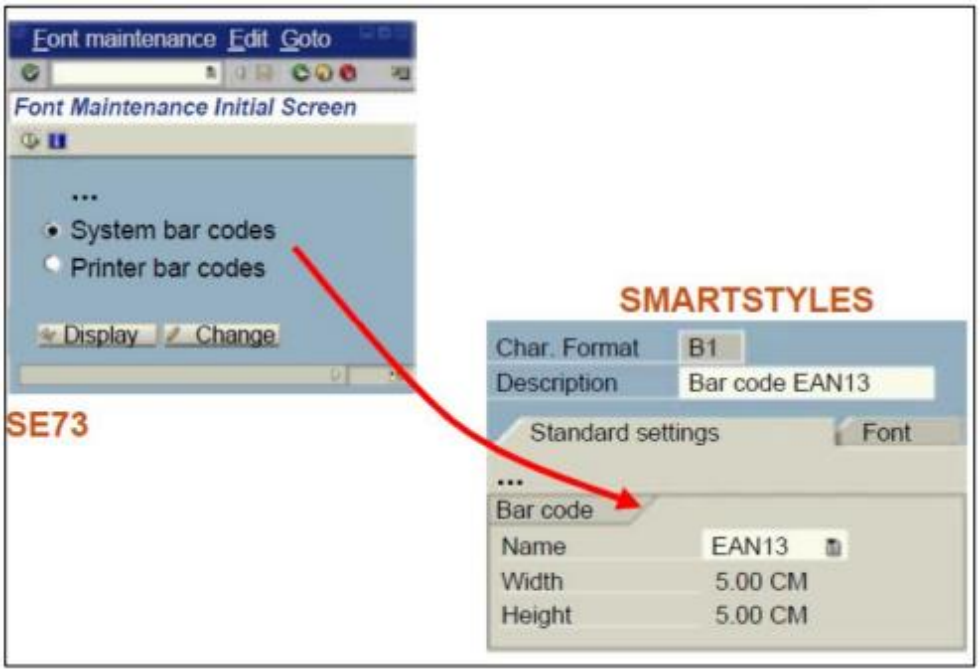
## *Font and Bar Code Maintenance*




## *Accessing Font Maintenance*



## Maintaining and Using Bar Codes




## OSS Notes for Fonts and Bar Codes

| | |
|---|---|
| 0008928 | List of supported printers/device types |
| 0005196 | Printing bar codes with SAPscript |
| 0017054 | How to copy or change a device type |
| 0012462 | How can I define a new printer font? |
| 0317851 | Printing PDF files in 4.6C/4.6B/4.5B/4.0B |
| 0201307 | TrueType fonts for Smart Forms/SAPscript |

## Review Question

- ***Question 1. Interface is used to transfer application data to the form definition.***
- ***Question 2: Text modules are included in forms using text nodes.***

---

# Lesson 19: Adobe Forms

---

## Overview

- ***Forms are used for mass printing in SAP systems. Besides using the printer for standard output you can also select the Internet (by using a generated HTML output), a fax, or e-mail as the output medium.***

- ***Tools Delivered by SAP for Form Designing***
  - ***SE71 – Sapscripts***
  - ***SmartForms – Smart Forms (introduced in SAP Basis Release 4.6C)***
  - ***SFP – Adobe Form (As of SAP NetWeaver '04 )***
- ***As of SAP NetWeaver '04 (in SAP Web Application Server), you can use a new solution to create interactive forms and print forms for the optimization of your form-based business processes. This solution uses Portable Document Format (PDF) and software from Adobe Systems Inc. that has been integrated into the SAP environment***

## ***Overview - Features***

- ***Create form templates for the layout that include logos or pictures***
- ***Edit forms online or offline***
- ***Forms can be filled in advance automatically with specific data from SAP applications and then sent to the correct recipients using secure methods***
- ***Automatic consistency checks for forms***
- ***Activate enhanced functions such as comments***
- ***Digital signatures and form certification***
- ***User-friendly tools reduce the time and costs associated with creating form layouts.***
- ***The usage of the PDF format means that forms retain their appearance regardless of the environment they are used in.***