Modulus Technical Note #23 - *SuperReport Forms*

(Amended 17/2/99)

Printing Forms

Forms printing in Modulus can use standard forms (known as ‘Modulus Internal Forms’) or custom (‘SuperReport Pro’) forms, which you can design to your own requirements.

If you wish to use SuperReport Pro forms by default, switch on the ‘SuperReport Editable Forms’ option on the Forms Settings 2 page of Preferences. If no SuperReport Pro form is present, the equivalent Modulus Internal Form will be used. In a particular situation, if you wish to use the Modulus Internal Form, hold down the <Option> (Mac) or <Alt> (Windows) key as you select the printing function.

SuperReport Pro Forms

Depending on which modules are present, SuperReport Pro versions of the following forms can be used:

File	**File Number**	**Report Code**
Sales Invoices	39	SR Invoice
		SR Invoice DT (ie an Invoice with a separate delivery address)
Sales Orders	24	SR Order Quote
		SR Order Auth
		SR Order Ack
		SR Delivery
Purchase Invoices	37	SR Remittance
		SR Cheque
Stock Movements	40	SR Movement
Purchase Orders	57	SR Purch Order
Service Contracts	17	SR Contract
Service Calls	20	SR Call Sheet

Modifying SuperReport Pro Templates

Samples of each of these are supplied with Modulus. If you wish to modify these samples to your own requirements, click the [Edit Templates] button in the Forms Settings 2 section of Preference and double-click on the required form in the resulting list. Ensure, however, that they retain the same Report Code so that Modulus can find them when it needs to. If a form is under development, it is recommended that the Report Code be changed temporarily, forcing Modulus to use

MODULUS SOFTWARE LTD MONKS MEADOW OAST APPLEDORE ROAD TENTERDEN KENT TN30 7DB UNITED KINGDOM

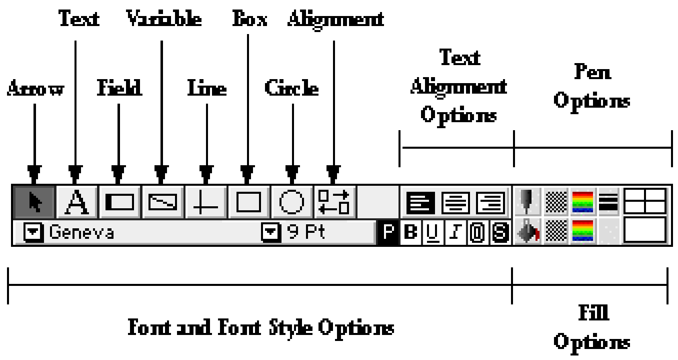
Switchboard: 01580 766666 E-mail: info@modulus.co.uk Fax: 01580 766667

its Internal Form until work on the SuperReport version has been completed.

Using SuperReport Pro

The basic skills needed to design forms in SuperReport are the same as those required for a simple drawing package. Items can be moved by dragging or resized by dragging handles.

The tool bar contains the following tools:



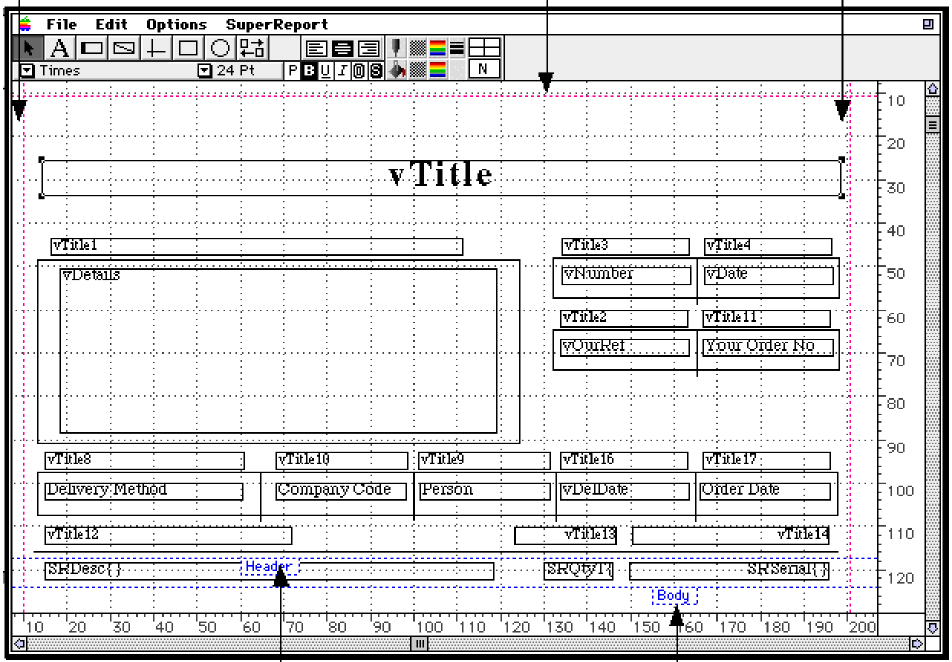
Arrow tool	for selecting, resizing and dragging objects
Text tool	for adding new static text items
Field tool	for adding new fields from the database
Variable tool	for adding new variable from the database
Line tool	for drawing either horizontal or vertical lines
Box tool	for drawing rectangles (and/or grids)
Circle tool	for drawing circles
Alignment tool	for aligning selected objects (a pop-up menu appears)

When working with forms, the Field tool will not be used very often: use of the Variable tool will be much more common. Some useful Field and Variable names are given in the Technical Notes discussing each form. Full details regarding the operation of each tool can be found in the Report Generator chapter, which is included in the Gold Core Functions manual and the Silver Reference manual.

Below is shown the top of the sample Delivery Note included with Modulus.

This area of the form is divided into two sections: the header and the body. The header contains infomration which will be printed at the top of each page, such as the delivery address and date. All this infomation appears above the blue Header line. If you need to change the size of the header area, the Header line can be moved by dragging the word 'Header'. The body area appears between the Header and Body lines and lists the Items that are being delivered. The body area is printed as many times as there are Items.

All objects should be within the pink dotted lines, which mark the printable area



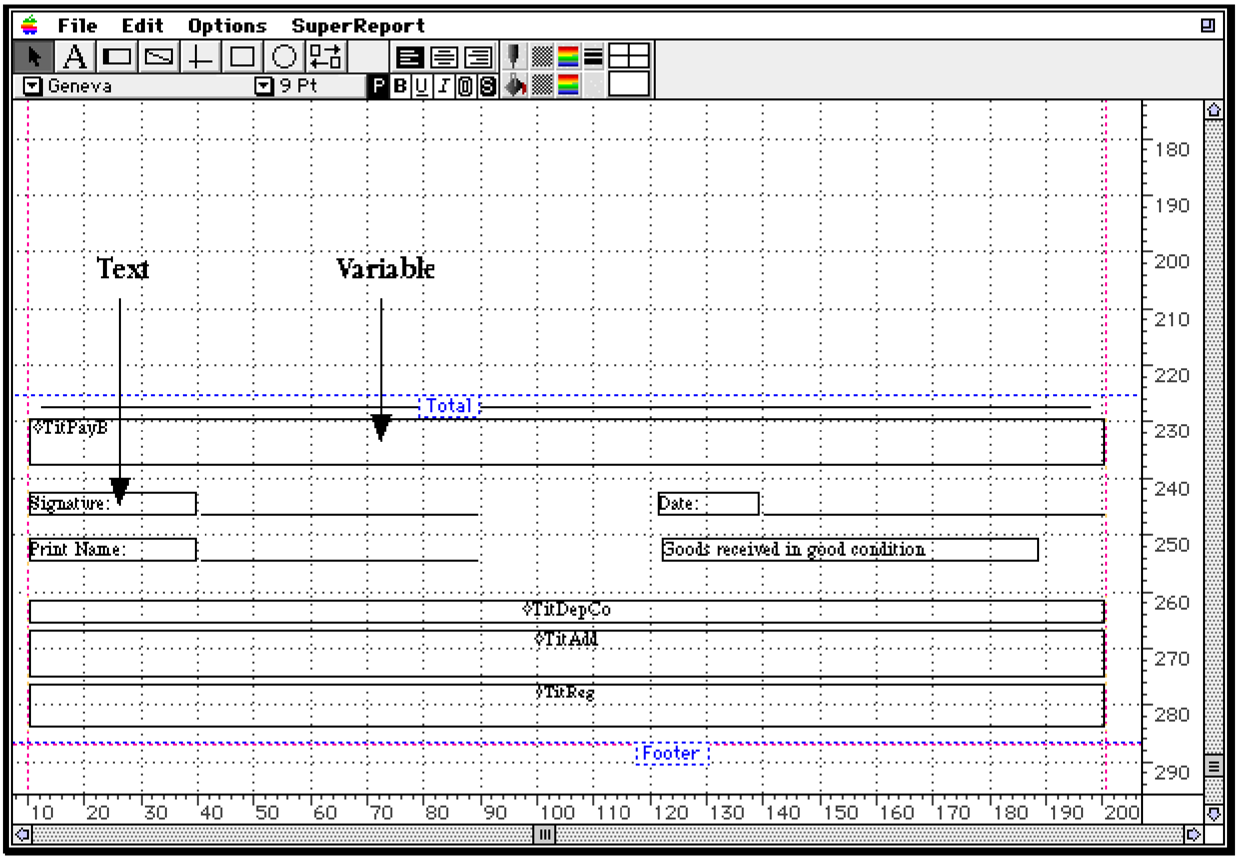
The Header line should be below the header information and above the Item variables, clear of and not touching any boxes

The Body line should be just below the Item variables

Below is shown the lower half of the Delivery Note.

Modulus Technical Note #23 - *SuperReport Forms*

(Amended 17/2/99)



The footer area (between the Total and Footer lines) is printed in a fixed position at the bottom of every page, irrespective of the number of Items. In many cases, such as the Invoice or the Purchase Order, the footer area will contain totals.

SuperReport is inconsistent in the manner in which it handles vertical lines. If you need a vertical line to cross area boundaries (for example, one that stretches from the header area into the body area), it is recommended that you draw two lines, one in each area each ending one pixel from the Header line.

Variables and Arrays in SuperReport Pro Forms.
When printing a form, Modulus' method of working is to load the information into variables before printing. These variables can then be included in the form if desired. In the first illustration in this Note, for example, the variable vDetails contains the Delivery Address (including a Contact Name), while the variable vDelDate contains the Delivery Date. The Technical Notes covering specific forms

MODULUS SOFTWARE LTD MONKS MEADOW OAST APPLEDORE ROAD TENTERDEN KENT TN30 7DB UNITED KINGDOM
Switchboard: 01580 766666 E-mail: info@modulus.co.uk Fax: 01580 766667

include lists of variables which can be used in those forms. Note that field identifiers are also contained in variables, usually named vTitle followed by a number. For example, vTitle1 appears above the Delivery Address and contains the text ‘Deliver to...’, while vTitle16 appears above the Delivery Date and contains the text ‘Delivery Date’. Again, these are listed in the Technical Notes discussing individual forms. If the text in a particular vTitle variable does not match your requirements, it is recommended that you remove the variable and replace it with text using the Text tool.

The general convention in Modulus is to precede variable names with the letter v. The exception to this rule is the ‘interprocess variable’, some of which are used in the footer area. These are preceded by ◊, which is obtained by using the key combination <Shift-Option-V> on the Macintosh and by pressing < and > in succession on Windows machines. Alternatively, hold down the <Alt> key while typing “0190”.

To place a variable into a form, use the Variable tool, draw the variable and select “Variable” in the Type pop-up menu in the resulting dialogue box.

In the body section of the form, Modulus loads the information into arrays, again before printing. An array is a set of variables, containing as many values as there are items to be printed. For example, in the illustration above, the Product Name for each item is contained in the SRDesc{} array. Most such arrays are preceded with the letters SR. When printing the second line of the Delivery Note for example, Modulus will use the second item in the SRDesc{} array to display the correct Product Name. To place an array into a form, use the Variable tool, draw the variable and select “Array (Automatic)” in the Type pop-up menu in the resulting dialogue box. Then, click [Repeat...] and ensure that the ‘Repeat this Object’ radio button is switched on. Also, switch on ‘Use Vertical Repeat Offset’ and set it to 2 points. There is no need to type the {} suffix as part of the array name: this will be done for you.

All other information is displayed using fields. Existing fields are distinguised by the lack of v, SR or ◊ leading characters. The use of fields should be limited to the Header area of the form and to those fields in the file to which the form belongs. A list of forms and their parent files appears on page one of this Note. Fields cannot be used succrssfully in the Body area of a form: arrays should be used instead. Information about loading data into arrays appears later in this Note.

Formatting Variables

Numbers and dates in fields, variables and arrays can be formatted. Select the Field or Variable tool as appropriate and draw the field or variable. Alternatively, double-click on one that already exists. In the resulting dialogue box, select an option from the Format pop-up menu. The number formats are self-explanatory. Below is a key for the date format options.

Short	23/6/94
Abbreviated	Thu, Jun 23, 1994
Long	Thursday, 23 June, 1994
Short2	23/06/94

Occasionally, numbers and dates are displayed using string variables. This is to ensure that, when the variable holds a zero value, they appear as blank when printed, rather than displaying 0 or 00/00/00. A side effect of this is that the variables cannot be formatted as described above.

If the formatting of such variables is more important than the hiding of zero information, you can place the value into another variable. For example, SRQtyT{} is an array of string variables used to display quantities on Invoices. If you would like to format the quantity in a particular way, rename SRQtyT{}, for example, SRQtyN{} by double-clicking and format it as required. Then, add the following line to the Start Script of the form:

MACRO(“Convert to no”)

Then, create a Macro with a Macro Code of “Convert to no”. Make sure it is a Same Process macro and enter the following as the Macro Text:

mSize:=**Size of array**(SRProdCode) `Learns how many times to run this For loop
ARRAY REAL(SRQtyN;mSize)
For(mCount;1;mSize)

```
    SRQtyN{mCount}:=num(SRQtyT{mCount})
  End for
```

Errors will result if the Macro is not 'Same Process' or if Tabs are inserted in the Macro Text.

Including your own information

You can add your own information to the Body area of a SuperReport Pro form. This example uses a Macro to display in the SR Remittance form whether each item is a pre-payment or a normal Invoice.

First, use the Variable tool to create an array in the Body area of the Remittance form named mSROther{}. Select "Array (Automatic)" in the Type pop-up menu in the resulting dialogue box. Then, click [Repeat...] and ensure that the 'Repeat this Object' radio button is switched on. Also, switch on 'Use Vertical Repeat Offset' and set it to 2 points. Then, call the Macro by adding the following line to the Start Script of the SR Remittance:

```
  MACRO("Show Payment")
```

Create a Macro with a Macro Code of "Show Payment". Make sure it is a Same Process macro and enter the following as the Macro Text:

```
  mSize:=Size of array(SRProdCode)    `Learns how many times to run this
For loop
  ARRAY STRING(11;mSROther;mSize)
  For(mCount;1;mSize)
    If(SRProdCode{mCount}="*@")
      mSROther{mCount}:="Pre-Payment"
    Else
      mSROther{mCount}:="Invoice"
    End if
  End for
```

In a similar manner, the following Macro can be used to display on a Delivery Note the Invoice Number to which each item belongs:

```
mSize:=Size of array(SRProdCode)    `Learns how many times to run this For loop
ARRAY STRING(11;mSRInvNo;mSize)
For(mCount;1;mSize)
  mSRInvNo:=[ORDER ITEMS]Invoice No
End for
```

The model used in this Macro will be useful for all forms displaying related file information in their Body areas. These include the Order Acknowledgement, Order Authorisation, Delivery Note, Estimate/Quotation and Service Contract (all of which use information from the [ORDER ITEMS] file as in the above example), the Stock Movement form (which uses information from the [STOCK ITEMS] file) and the Service Call sheet (which uses information from the [ORDER ITEMS] file in its Charges and Parts Used section).

Other forms display subfile information in their Body area. These include the Invoice, Purchase Order, Remittance and Service Call (in its Products Involved section). The following Macro will serve as a model for displaying subfile information in the Body area in one of these forms. It displays the Analysis Code of each Item on an Invoice.

```
mSize:=Size of array(SRProdCode)    `Learns how many times to run this For loop
ARRAY STRING(5;mSRAnCode;mSize)
For(mCount;1;mSize)
  mSRAnCode:=[INVOICES]ITEMS'Analysis Code
End for
```


MODULUS SOFTWARE LTD MONKS MEADOW OAST APPLEDORE ROAD TENTERDEN KENT TN30 7DB UNITED KINGDOM

Switchboard: 01580 766666 E-mail: info@modulus.co.uk Fax: 01580 766667