



English User Manual

**Copyright © (1990-2022) SW-Tools ApS**  
**Duevej 23**  
**DK-2680 Solrød Strand**  
**Denmark**  
**Phone: +45) 33 33 05 56**  
**Mail: swtools@swtools.com**  
**www: www.swtools.com**

**Functions for release**

**22/11/01 / 2022-09-01 008.384**

## Contents

Contents .....	2
1. Pre-Release functions.....	4
1.1. DIALOG - Function for additional input.....	5
1.2. FIRST - Calculate first (RAP).....	6
1.3. <u>FTP</u> - File Transfer Processor .....	7
1.4. PARAMS - Function for additional report start parameters (RAP) .....	8
1.5. PRINT(L= - Function for additional report start parameters (RAP) .....	9
1.6. PRINT(LAB= - Label function .....	10
1.7. SCRPR - Recall screen print (IQ) .....	11
1.8. <u>USERINFO</u> - Get information about user.....	12
2. Error corrections.....	13
2.1. Username protection by start of report was not properly implemented .....	14
2.2. Calculations in IQ/DM incorrectly written when source file size > 12 KB .....	15
2.3. G3000 files not supported on 99 files per unit systems .....	16
2.4. Focus lost when using chain(+999) from one DATAMASTER program to another .....	17
2.5. Free fields for Total records and key activated .....	18
2.6. GPF in IQ/DM when working with CHAIN() and EXIT.....	19
3. Navision Financials.....	20
3.1. Install Navision Financials and C/FRONT .....	21
3.2. Install and configure interface to Navision Financials .....	22
3.3. Import of table definitions.....	23
4. Concorde C4/C5/XAL.....	25
4.1. Install Concorde.....	26
4.2. Install and configure interface to Concorde.....	27
4.3. Import of table definitions.....	28
5. Btrieve version 5.10/6.15 .....	29
5.1. Install and configure interface to Btrieve.....	30
5.2. Advanced options.....	31
5.2.1. Open mode .....	32
5.2.2. Name suffix.....	33
5.2.3. Name options .....	34
5.2.4. Parameters.....	35
6. Printing Labels.....	36
6.1. How to use in RAPGEN.....	37
7. Application type for controlling interfaces.....	38
7.1. Type 1 - Support of Informix Database table locks .....	39
7.2. Type 2 - Support of SAMSON system.....	40
7.3. Type 4 - Always perform ORDER BY .....	41
7.4. Type 4096 - Alpha fields may NOT be NULL.....	42
7.5. Type 8192 - ODBC Drivers with support of one connection per table only.....	43
8. New pack types .....	44
8.1. P1053 - Right justified fields in C-ISAM files .....	45
8.2. New printers in TRIO .....	46
8.3. Screen printer .....	47
9. MOF printers .....	48
9.0.1. Printer type .....	49
9.0.2. Type 1 - Text printer .....	50
9.0.3. Type 2 - HTML printer.....	51
9.0.3.1. HTML output of standard reports .....	52
9.0.3.2. HTML output of non-standard reports.....	53
9.0.4. Type 3 - RTF printer .....	54
9.0.5. Type 4 - TXT printer .....	55

Functions for release

9.0.6. Type 5 - SSV printer.....	56
10. Functions.....	57
10.1. New functions.....	58
10.2. <u>LINE</u> - Retrieve or set the current IQ/DM line counter .....	59
10.3. <u>OBJECTGETSTRING</u> - Get index of an objects selected item.....	60
10.4. <u>GETINFO</u> - Get additional program information (IQ/DM).....	61
Figure list.....	62
Index.....	63

## **1. Pre-Release functions**

These functions may be released.

## 1.1. DIALOG - Function for additional input

Number DIALOG(Fields par1)

Parameters: Par1: Fields to show in dialog

Description: The DIALOG function enables the user to pop up dialogboxes with a selected set of fields at any point of a report execution or in an IQ program for example by click on a field.

DIALOG("#1,7-8,le#3") defines a dialog with the given fields. The fields documentation is used as floating online help when the mouse cursor is moved over the leading text.

Returnvalue: Zero.

See also: PARAMS

Example: DIALOG("1-3,11") /\* Make a dialog with the given fields

## **1.2. FIRST - Calculate first (RAP)**

FIRST

Parameters: None.

Description: All following lines are calculated first.

Returnvalue: None.

See also: NORMAL, LAST, AFTER, TOTAL, SORT, INIT, ATEXT

Example: FIRST                   /\* Switch to first calculations

## 1.3. **FTP** - File Transfer Processor

FTP(Text *par1*)

**Description:** The FTP function has been build in to enable advanced users to transfer files for example in a report based on a SSV file containing filenames. FTP("command") executes the ftp command as:

Note that freefields may be used for the command and that the 32 bit version supports long filenames.

**Returnvalue:** None.

**See also:** [sa funny.c\(099\) FTP Standard](#)

**Example:**

```
FTP("open 200.0.0.9")
FTP("user cms, mypas")
FTP("binary")
FTP("get /X.BASIC/0/AFIL c:/mydir/myfil")
FTP("quit")
```

## **1.4. PARAMS - Function for additional report start parameters (RAP)**

PARAMS(Fields par1)

Parameters: Par1: Fields to show in start parameter dialog

Description: PARAMS("#1,7-8,le#3") is a variant of the dialog function where the input is done by start of the report not during report execution.

Use of PARAMS in a report will add a button <Extra parameters> to the startup screen which then activates the dialog.

Returnvalue: None.

See also: DIALOG

Example: PARAMS("1-3,11") /\* Make a dialog with the given fields



## **1.5. PRINT(L= - Function for additional report start parameters (RAP)**

PRINT(L=Text par1)

Parameters: Par1: Define lines for next print

Description: Print sets the printlines for next print.

Returnvalue: None.

See also: PRINT

Example: PRINT("L=1-3,11")           /\* Set next lines to print

## **1.6. PRINT(LAB= - Label function**

PRINT(LAB=Text par1,Number par2,Number par3)

Parameters: Par1: some Par2: good Par3: param

Description: See mo.

Returnvalue: None.

See also: PRINT

Example: PRINT("LAB=1")           /\* Define label print

## **1.7. SCRVRT - Recall screen print (IQ)**

SCRVRT(Filename par1)

Parameters: Par1: Filename to show using the screen printer

Description: SCRVRT("filename") calls up the screen printer with the saved print from filename. This may for example be used in IQ by click on a field.

Returnvalue: None.

See also: PRINT

Example: SCRVRT("c:/w/ab.cde") /\* Show this file using the screen printer

## **1.8. USERINFO** - Get information about user

text USERINFO(number *par1*)

17=User defined

**Description:** This function gets the requested user information.

The number in *Par1* refers to the fieldnumber in the system file US where you may define field 11 to 17 individually for each installation, just be carefull if later upgrading the version of TRIO.

**Returnvalue:** String containing the user information.

**See also:**[sv\\_use.c login](#)

**Example:**

```
#11=USERINFO(6) /* Get the user first remark
```

## **2. Error corrections**

## **2.1. Username protection by start of report was not properly implemented**

When protecting the report with the subfunction PAS() it worked fine when working within the report generator. However, when starting the report from the windows system using command "RAPWIN.EXE <report number>" the system failed to check on the password. This has now been implemented.

## **2.2. Calculations in IQ/DM incorrectly written when source file size > 12 KB**

In IQ/DM the calculations was damaged if the filesize exceeded 12 KB. The damage would normally appear as missing parts or incorrect characters.

## **2.3. G3000 files not supported on 99 files per unit systems**

The search function for G3000 databases could not find any on a 99 files per unit based system. This was because the offset of the data area in a volume with 99 files per unit differed from the unit with 250 files.

The problem therefore not only applies to the search function, but also to the access of the actual data.



## **2.4. Focus lost when using chain(+999) from one DATAMASTER program to another**

When using chain(+999) in DATAMASTER you might not get the focus in the newly activated program as expected. This is caused by that DATAMASTER actually activates the chain directly after the calculations has finished, and hereby getting the focus in the new program. But the side effect is, because DATAMASTER is event driven, that the calling program will hereafter set the focus to the next/current input field again and therefore taking the focus away from the called program.

The problem is solved by using the rule, that DATAMASTER should only set focus on the field if the program is active and has got the focus.

## **2.5. Free fields for Total records and key activated**

All reports defined in RAPGEN has as free field WW#51 define to gain access for the total name. Furthermore, two free fields exists, WW#50 Total number of records for total and WW#52 the total keyvalue. Both values are used internally by RAPGEN to generate the requested totalname as 99 Name xxx, but was not activated for use in calculations.

## **2.6. GPF in IQ/DM when working with CHAIN() and EXIT**

There was an internal error in IQ/DATAMASTER which caused a GPF at some point when working with CHAIN and EXIT to shift between programs.

### **3. Navision Financials**

This section describes how to install and use TRIO & ODBC in connection with Navision Financials. C/FRONT version 1.2 or higher is supported.

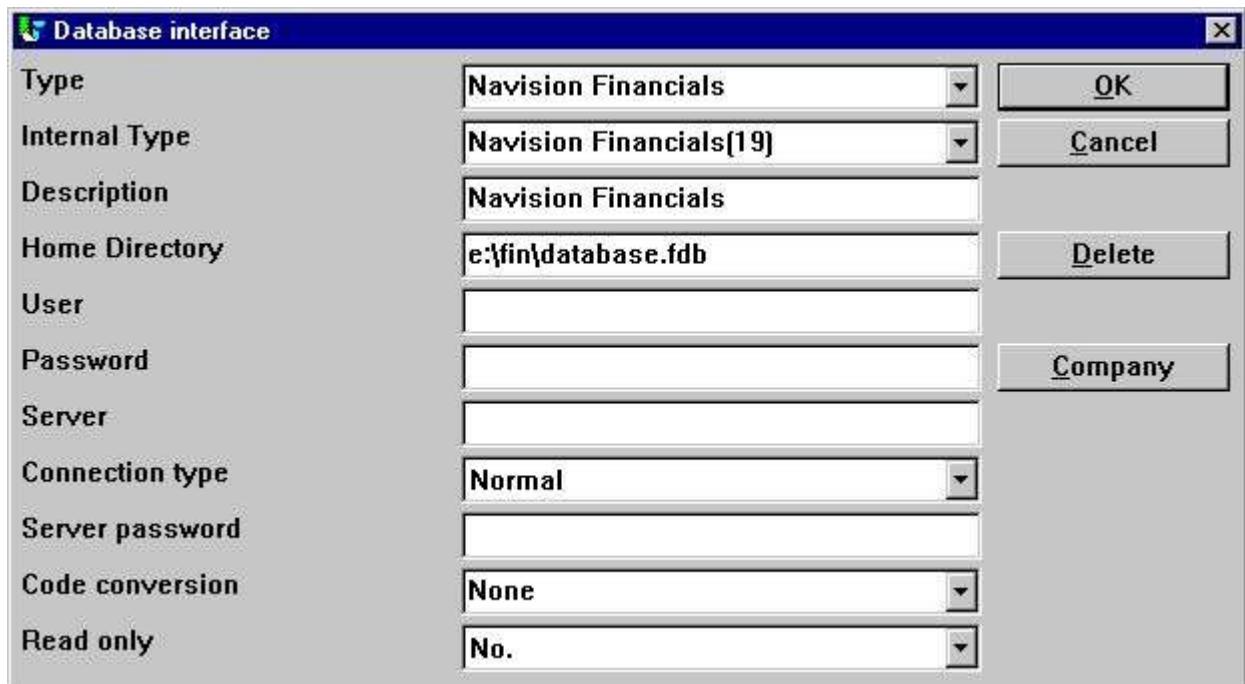
### **3.1. Install Navision Financials and C/Front**

Install Navision financials and C/Front for example in directory e:\fin and make sure that the PATH environment variable includes the directory e:\fin. If you change the PATH variable restart the Windows system.

## 3.2. Install and configure interface to Navision Financials

Follow the description for installing a ODBC driver interface in TRIO. The interface for Navision is named 'Navision Financials'.

By the function 'Database-interface' you may select the installed 'Navision Financials' where the following setup will correspond to the described sample installation.



The screenshot shows a dialog box titled "Database interface" with the following fields and values:

Field	Value	Action
Type	Navision Financials	OK
Internal Type	Navision Financials(19)	Cancel
Description	Navision Financials	
Home Directory	e:\fin\database.fdb	Delete
User		
Password		Company
Server		
Connection type	Normal	
Server password		
Code conversion	None	
Read only	No.	

### ***1. Sample setup of Navision Financials interface***

If you use C/Front with TCP/NETB for server connection the field Server may be used to enter **server,type** where server is the server name and type is tcp or netb.

### 3.3. Import of table definitions

Before you import the Navision table definitions it is best to create a new subsystem dedicated to use with Navision. The following figure show a sample of a subsystem created from menu 'File' - 'Subsystem' - 'New...':

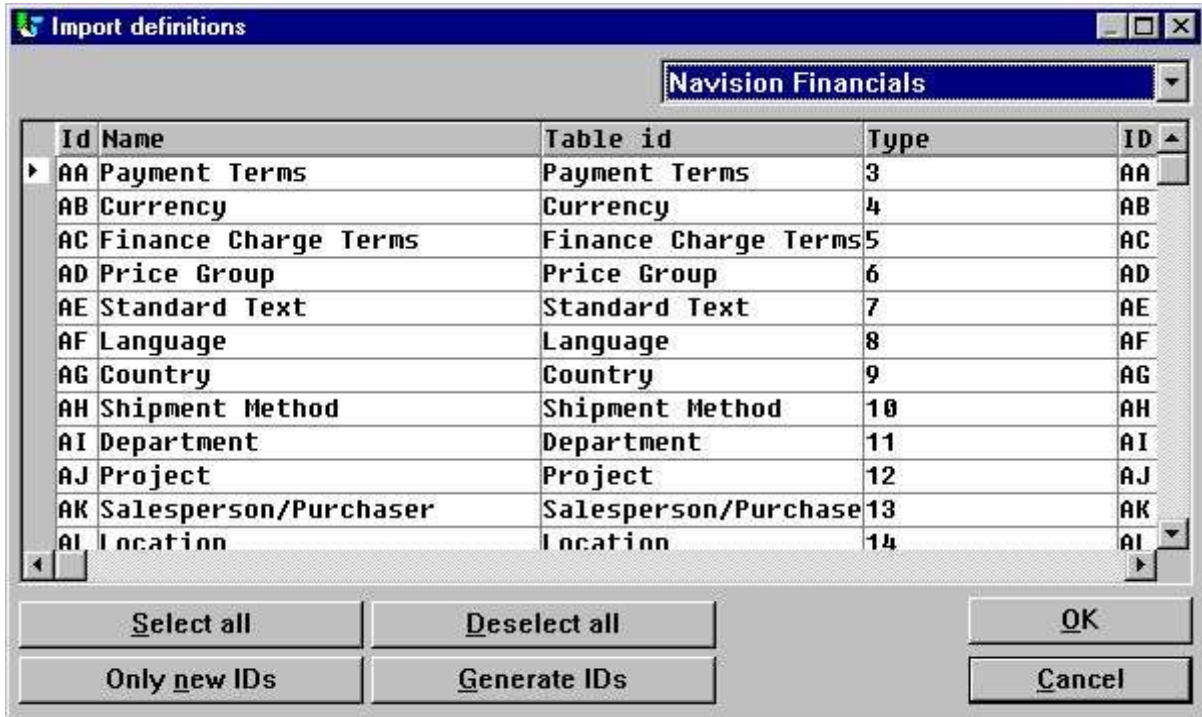
The screenshot shows a dialog box titled "Subsystem" with the following fields and values:

Name:	navision	0042
Password:		Type: Normal
Company:		Module: RAPGEN & IQ
File definitions:	e:\fin\navision	
Database:		
Reports:	e:\fin\navision	
Databaselock:		
Description:		

Buttons: OK, Cancel

#### ***2. New subsystem for Navision table definitions***

From the menu 'File' - 'Import ODBC definitions...' you may select interface 'Navision Financials' in order to import all or some of the table definitions.



**3. Import of all or some of the Navision table definitions**



## **4. Concorde C4/C5/XAL**

This section describes how to install and use TRIO & ODBC in connection with Concorde C4/C5/XAL.

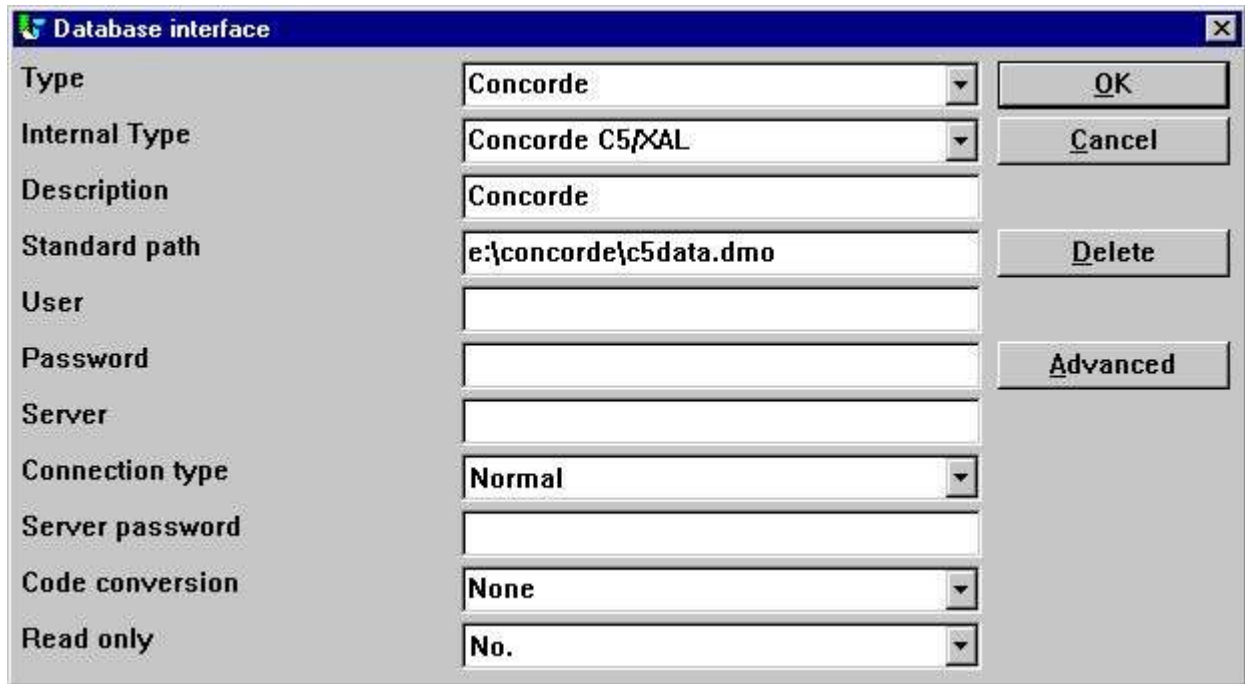
## **4.1. Install Concorde**

Install Concorde for example in directory e:\concorde and export all file definitions into a .DBD file. The export may be done by login as 'supervisor' and select the menu 'General' - 'Admendments' - 'Development menu'. From here you may select 'Export' - 'DBD' - 'All + Enum'. The exported file is then saved in the same directory as the Concorde installation, e.g. e:\concorde.

## 4.2. Install and configure interface to Concorde

Follow the description for installing a ODBC driver interface in TRIO. The interface for Navision is named 'Concorde C4' or 'Concorde C5/XAL'.

By the function 'Database-interface' you may select the installed 'Concorde' where the following setup will correspond to the described sample installation.



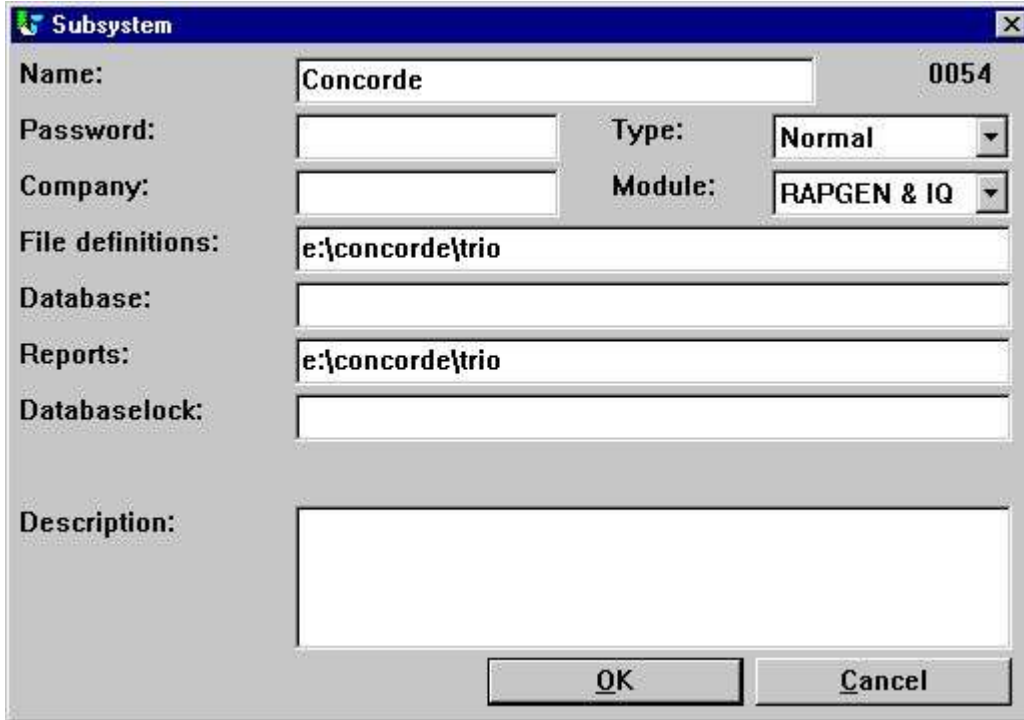
The screenshot shows a dialog box titled "Database interface" with a close button (X) in the top right corner. The dialog contains several fields and buttons:

Type	Concorde	OK
Internal Type	Concorde C5/XAL	Cancel
Description	Concorde	
Standard path	e:\concorde\c5data.dmo	Delete
User		
Password		Advanced
Server		
Connection type	Normal	
Server password		
Code conversion	None	
Read only	No.	

### 4. Sample setup of Concorde interface

### 4.3. Import of table definitions

Before you import the Concorde table definitions it is best to create a new subsystem dedicated to use with Concorde. The following figure show a sample of a subsystem created from menu 'File' - 'Subsystem' - 'New...':



#### ***5. New subsystem for Concorde table definitions***

From the menu 'File' - 'Import ODBC definitions...' you may select interface 'Concorde' in order to import all of the table definitions.

Please NOTE that a user name is required in order to import definitions. The user name is 'BASIC'.

When the user name has been entered the import will be of all tables, e.g no selection can be made. If required you may delete one or more of the imported definitions afterwards.

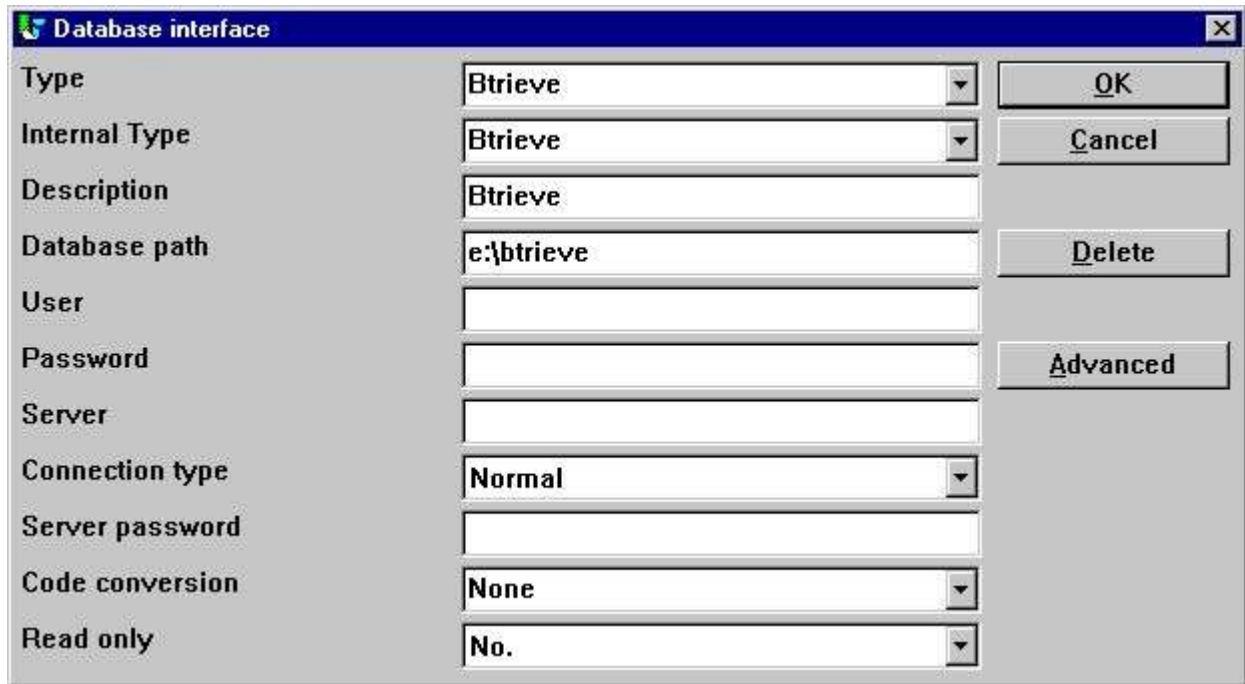
## **5. Btrieve version 5.10/6.15**

Btrieve files can be accessed directly without use of ODBC. This will give a much better performance when working with TRIO.

## 5.1. Install and configure interface to Btrieve

Follow the description for installing a ODBC driver interface in TRIO. The interface for Btrieve is named 'Btrieve'.

By the function 'Database-interface' you may select the installed 'Btrieve' where the following setup will correspond to access of Btrieve files located in the directory 'e:\btrieve'.



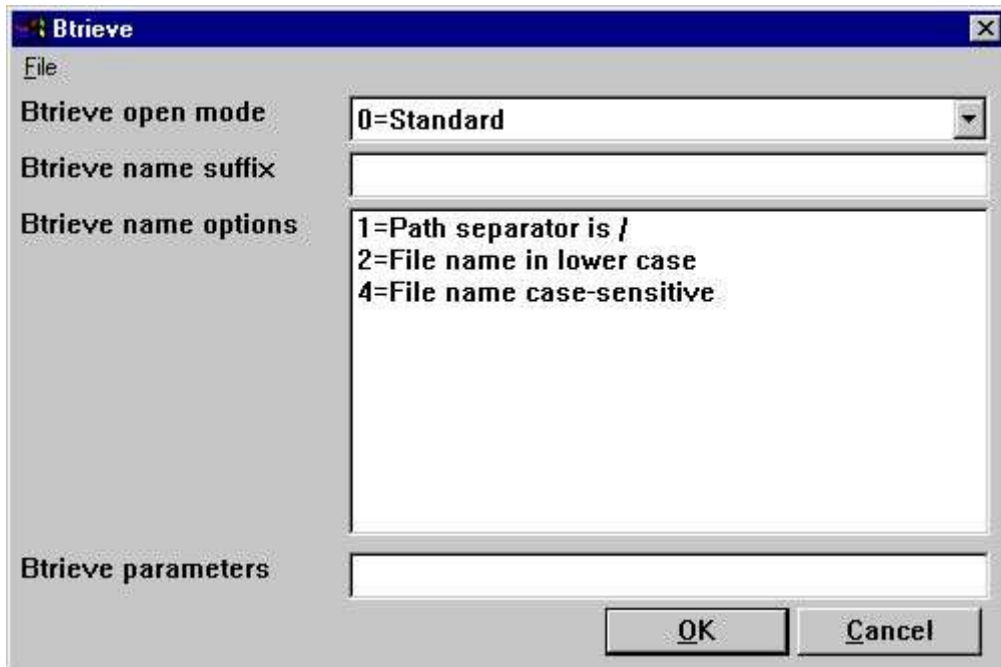
The screenshot shows a dialog box titled "Database interface" with a close button (X) in the top right corner. The dialog contains several fields and buttons:

Type	Btrieve	OK
Internal Type	Btrieve	Cancel
Description	Btrieve	
Database path	e:\btrieve	Delete
User		
Password		Advanced
Server		
Connection type	Normal	
Server password		
Code conversion	None	
Read only	No.	

**6. Sample setup of Btrieve interface**

## 5.2. Advanced options

The Btrieve interface allows you to control how files are to be opened and how the file name should be generated.



***7. Advanced settings on Btrieve interface***

## 5.2.1. Open mode

The open mode may be one of the following:

- **Standard**
- **Exclusive**
- **Single Engine File Sharing (Version 6.15 only)**
- **Multi Engine File Sharing (Version 6.15 only)**

By standard TRIO will open a Btrieve file in read-only mode unless it is a file which is to be updated on a report or in a DATAMASTER program.

If the open mode is Exclusive, no other program can gain access to the file.

The Single and Multi Engine File Sharing modes are for Btrieve version 6.15 only. Please refer to the Btrieve documentation for a detailed description of the openmodes.



## **5.2.2. Name suffix**

The name suffix of a Btrieve file was always set to .DAT in TRIO version  $\leq$  006.008. However, Btrieve files may be without any suffix or a different one than .DAT. Therefore, it is a field you may fill yourself when installing the interface for use in TRIO.

### 5.2.3. Name options

These options controls how the file name of a Btrieve file is generated by TRIO. The naming options for a Btrieve may override the normal generation of the actual file location. As an example, assuming that the interface has been setup with database path

**c:/btrieve/database**

and a file is defined with name

**customer**

and the suffix is set as

**dat**

the actual location of the file is

**c:/btrieve/database/customer.dat**

Because Btrieve does NOT allow the use of / in the path the default behaviour is to replace all occurrences of / with \ giving a location as

**c:\btrieve\database\customer.dat**

Should this however change in newer Btrieve versions or only be dependent on the operating system used, it can be changed here.

## 5.2.4. Parameters

When Btrieve 5.10 is activated is is required to pass some initial parameters to the Btrieve system. The parameters are by default:

**/m:48 /b:16 /f:20 /l:20 /p:4096**

Please refer to the Btrieve 5.10 manual for a detailed description of the possible parameters and the meaning hereof. The parameter string has no effect when using Btrieve 6.15 or newer.

## 6. Printing Labels

The subfunction PRINT() can be used to produce print of labels in a very simple way. The function syntax is:

**PRINT(LAB=direction columns rows width height copies)**

The direction control how to generate the labels on the label sheet. The valid values are:

**0 - No labels**

**1 - From left to right**

**2 - From top to bottom**

The multiply of number of columns and rows is the actual number of labels on one sheet.

The width and height of any label on the sheet can be given in centimeters or inches by using the following syntax:

**7cm equals 7 centimeters**

**2in equals 2 inches**

Finally when printing labels you may want to print multiple copies of the same label, which can be controlled by the copies parameter.

A sample use the the PRINT(LAB=...) function could be:

**PRINT(LAB=1 3 7 7cm 7cm 2)**

which will produce labels printed from left to right on a label sheet with 21 labels, 3 on each row, 7 rows, where each label has the width/height of 7 centimeters. Each label is printed in 2 copies.

## 6.1. How to use in RAPGEN

To produce the simple print of labels a report must be defined as a letter. The main file of the report should as an example be the customer file if you want to print customer labels.

On the layout of the letter you only require to define one label in the top left part of the layout.

When the label is defined you must remember to change the Page control, which is normally set to :1,1-99 for a letter, to 1-x where x is the number of lines used to define the label.

Finally you have to add the following calculations to the report:

**FIRST**

**PRINT(LAB=1 3 7 7cm 7cm 2)**

Please remember to have the PRINT(LAB=...) in the FIRST section. Otherwise, it will reset the label control for each record read from the main file.

## **7. Application type for controlling interfaces**

The application type can be used to control how TRIO should work with an interface. It is normally used in connection with ODBC interfaces, but may in some cases also be present for others.

When using the type for an interface, please note that one or more types may be used at the same time by adding the types together.

## **7.1. Type 1 - Support of Informix Database table locks**

The normal use of an ODBC data source for Informix will lock any used table when connected to it. This will for example cause a problem if a user is working with IQ or RAPGEN to query or print information from the Informix database at the same time when a user wants to work with the table from another application.

It is described in the Informix database documentation that you have to execute a SQL statement

**set isolation to dirty read**

to avoid the problem.

## **7.2. Type 2 - Support of SAMSON system**

The SYSTEM named SAMSON, is a financial system for DOS/Windows and UNIX, using C-ISAM as Database. In this system, all files are created with a company number as the first key part always. TRIO supports the use of company numbers on interfaces and sub systems, but normally only in connection with the file name.

This type support company number as the first key part, only for the interface C-ISAM (MASI),



### 7.3. Type 4 - Always perform ORDER BY

Because not all ODBC drivers support the standard given by Microsoft, you may experience SQL errors when trying to print reports with access to a table using the ODBC driver. The error will state a missing ORDER BY ... for the table.

A report that selects the fields from a customer table, where customer is the primary key will generate

```
SELECT customer,name,address FROM customer_table
```

If type 4 is set it will generate

```
SELECT customer,name,address FROM customer_table ORDER BY customer
```

## **7.4. Type 4096 - Alpha fields may NOT be NULL**

Because some ODBC drivers / SQL Databases does NOT support alpha numeric fields with no content, e.g "" it is necessary to inform TRIO that all alpha fields must contain some value. This type will automatically use the field pack type 1048 on all fields, which will set the content of an empty alpha field to one space.

## **7.5. Type 8192 - ODBC Drivers with support of one connection per table only**

Because some ODBC drivers / SQL Databases requires a unique connection for each table, where as TRIO tries to optimize table access by doing the connection to the database only once it can be necessary to set this type.

If this type is set, TRIO will create a connection to the database for each table used. Please note, if the ODBC data source requires the user to enter a user name / password to connect to the database, it will be required to do so for each table used. To prevent this, the user name and password must be setup one for the database interface in TRIO.

## **8. New pack types**

## **8.1. P1053 - Right justified fields in C-ISAM files**

Because the field pack type 1043, which is used for the interface Concorde is working with an additional byte, the use hereof will fail on C-ISAM files.

By using the P1053 right justified fields will work properly.

## **8.2. New printers in TRIO**

### **8.3. Screen printer**

The screen setup in TRIO may now be given multiple time by setting the check mark for screen printer on an empty prtinter line in the setup dialog.

Unknown picture (g:/rapdoc/mod-eng001.jpg)

#### ***8. Additional screen printer insertion***

## 9. MOF printers

In general TRIO now provides a number of new printers which may be setup initially as a screen printer. The button 'Setup' can now be used to set the type of printer output for the defined screen printer.

Unknown picture (g:/rapdoc/mod-eng002.jpg)

### ***9. Setup of screen printer to MOF printer***



## **9.0.1. Printer type**

The printer type may be one of the following:

- 1 - Text**
- 2 - HTML**
- 3 - RTF**
- 4 - TXT**
- 5 - SSV**

## **9.0.2. Type 1 - Text printer**

Because the Window Generic text printer has its limitations and does not produce a very good and useable content, TRIO now has its own text printer. The output will be into one file, where each page is separated by a form feed (the hexadecimal character 0x0c). The printer will automatically calculate the width of any field used in the layout in order to produce an output with no overlapping fields.

### **9.0.3. Type 2 - HTML printer**

The HTML (Hyper Text Markup Language) printer can be used to produce output directly to internet/intranet systems. This means, an output from this printer can directly be viewed in your internet/intranet browser, e.g. Netscape or Explorer.

The output may also be send as attachment to an e-mail, whereas the receiver of the mail may then view the report directly in a browser.

Because HTML does NOT support any pixel positions the printer can be used in two ways.

### **9.0.3.1. HTML output of standard reports**

TRIO will have no problems with standard reports. The report will be converted to a HTML table, where each field corresponds to a column in the table.

Any total level or group total defined will be inserted in the same table.

All used pictures, OLE objects and charts will automatically be converted to the standard picture format GIF (Graphical Interchange Format) known to HTML.

### **9.0.3.2. HTML output of non-standard reports**

If you have reports defined, which cannot directly be converted into a table you might experience a wrong output from the HTML printer. However, you may then setup a second HTML printer and selection the option

**'Use textprinter to make fixed output'**

When this printer is used instead, the report output will be correct, because it is a simple text output using a fixed font in the HTML language.

For this fixed HTML printer no pictures, OLE objects or charts will be included.

### **9.0.4. Type 3 - RTF printer**

The RTF (Rich Text Format) printer produces output in the same way as the HTML printer. The output from here can as an example be used to include directly in a Microsoft Word document. The only difference between the RTF and HTML printer lies in the definition file used, so please refer to the description of the HTML printer.

## **9.0.5. Type 4 - TXT printer**

## **9.0.6. Type 5 - SSV printer**



## **10. Functions**

## **10.1. New functions**

**10.2. LINE** - Retrieve or set the current IQ/DM line counter

**10.3. OBJECTGETSTRING**- Get index of an objects selected item

## **10.4. GETINFO**- Get additional program information (IQ/DM)

number GETINFO(number *par1*, text *par2*)

**Description:** This function allows you to get some special information from an IQ/DM program. The type 0 and 1 will return the unique id of the window, which may be used by other functions to manipulate the window. A sample of this is present in the OLE manual. When the type is 2 to 5 the function requires a field reference in *par2*. For example, to get the start column for article field number 7 *par2* should equal "va#7". The coordinates of a field is here given in the actual size of the field defined in IQ/DM.

If you require the actual coordinates of a field according to the scale factor currently used, e.g. zoom in/out, use the type 6 to 9 instead.

**Returnvalue:** Type 0-1 returns a unique window id. The value can be held in a 9,T2 field format. Type 2-9 returns a field coordinate. The value can be held in a 9,T2 field format.

**See also:**

**Example:** GETINFO(0) /\* Get the IQ program window id GETINFO(2,"va#7"); /\* Get the start x coordinate of va field 7

## Figure list

1. Sample setup of Navision Financials interface.....	22
2. New subsystem for Navision table definitions .....	23
3. Import of all or some of the Navision table definitions .....	24
4. Sample setup of Concorde interface .....	27
5. New subsystem for Concorde table definitions .....	28
6. Sample setup of Btrieve interface .....	30
7. Advanced settings on Btrieve interface .....	31
8. Additional screen printer insertion.....	47
9. Setup of screen printer to MOF printer.....	48

## Index

### A

ATEXIT ..... 5

### B

BASIC ..... 6;27

Btrieve ..... 28;29;30;31;32;33;34;61

### C

Calculations .....14

CHAIN.....18

C-ISAM .....39;44

Concorde.....24;25;26;27;44;61

### D

Database..... 21;26;29;38;39

Database-interface ..... 21;26;29

DATAMASTER..... 16;18;31

DBD.....25

DIALOG..... 4;7

### F

Filename .....10

Financials ..... 19;20;21;22;61

FTP ..... 6

### G

GETINFO .....60

### H

HTML ..... 48;50;51;52;53

### I

Import ..... 22;23;27;61

Informix .....38

IQ .....4;10;14;18;38;58;60

### L

Label ..... 9

Language ..... 50

### M

MASI..... 39

Microsoft .....40;53

### N

Navision ..... 19;20;21;22;23;26;61

Netscape ..... 50

### O

ODBC

19;21;22;24;26;27;28;29;37;38;40;41;  
42

OLE ..... 51;52;60

### P

P1053 ..... 44

PARAMS ..... 4;7

Printer..... 48

### R

RAPGEN ..... 17;36;38

RTF.....48;53

### S

SAMSON..... 39

SCRPT ..... 10

Server..... 21

### T

TRIO

11;19;21;24;26;28;29;31;32;33;37;39;  
41;42;45;46;47;49;51

### U

USERINFO ..... 11