

Quick Report Wizard – Part II

By Luis Pineiros, Technical Services Team Member, 4D Inc.

Technical Note 09-35

Table of Contents

Table of Contents	2
Abstract	3
Introduction.....	3
Types of Reports	3
Quick Report Plug-in	3
Quick Report Commands	4
List Reports in the Sample Database	5
Step 1: Select your fields and the sorting order	11
Step 2: Define the values inside your break levels.....	14
Step 3: Define an action in your break levels.....	15
Step 4: Define a format for your cells.....	17
Step 5: Define the style attributes for your cells	18
Step 6: Define a border for your cells	19
Step 7: Define the width of your columns.....	22
Step 8 - 9: Define the page header and footer for your report.....	23
Step 10: Generate your Report.....	24
Cross-Tab Reports in the Sample Database.....	25
Step 1-2-3: Define a format for your cells	26
Step 4: Select a computation for Data Source cell	28
Step 5: Select a computation for the Total Column:.....	29
Step 6: Select a computation for the Grand Total line	30
Step 7: Select the sorting order.....	30
Step 8: Formatting your data	32
Step 9: Define the style attributes for your cells	33
Step 10: Define a border for your cells	34
Step 11: Define the width of your columns.....	37
Step 12 - 13: Define the page header and footer for your report.....	38
Step 14: Generate your Report.....	39
Navigation.....	40
Conclusion.....	47

Abstract

Reports represent one of the most important functions in data management. A Quick Report is the ideal way to offer your users the possibility of creating their own reports to extend the custom reporting capabilities of your application. This Technical Note revisits creating a custom interface for Quick Reports using 4D v11 SQL commands.

Part II focuses in the implementation and programming of the sample database. Part I (Knowledge asset #75840) focused on the interface and functionality of the Quick Report Wizard.

Introduction

In Part I of this Technical Note we already established how useful it is to be able to reproduce a Quick Report interface and control the printing options available to your users.

In Part II we analyze the programming code necessary to establish a Quick Report interface. By understanding the QR Commands available in 4D v11 SQL and how to apply them, you can complete the implementation of a "Print Center" in your application. Printing options should include a range of reports from built-in reports the user can select from a menu item to user-generated custom reports.

Types of Reports

There are essentially two types of Quick Reports; List Reports and Cross-Tab Reports. This Technical Note addresses both.

List Reports display records in a list with break levels where additional mathematical operations can be performed.

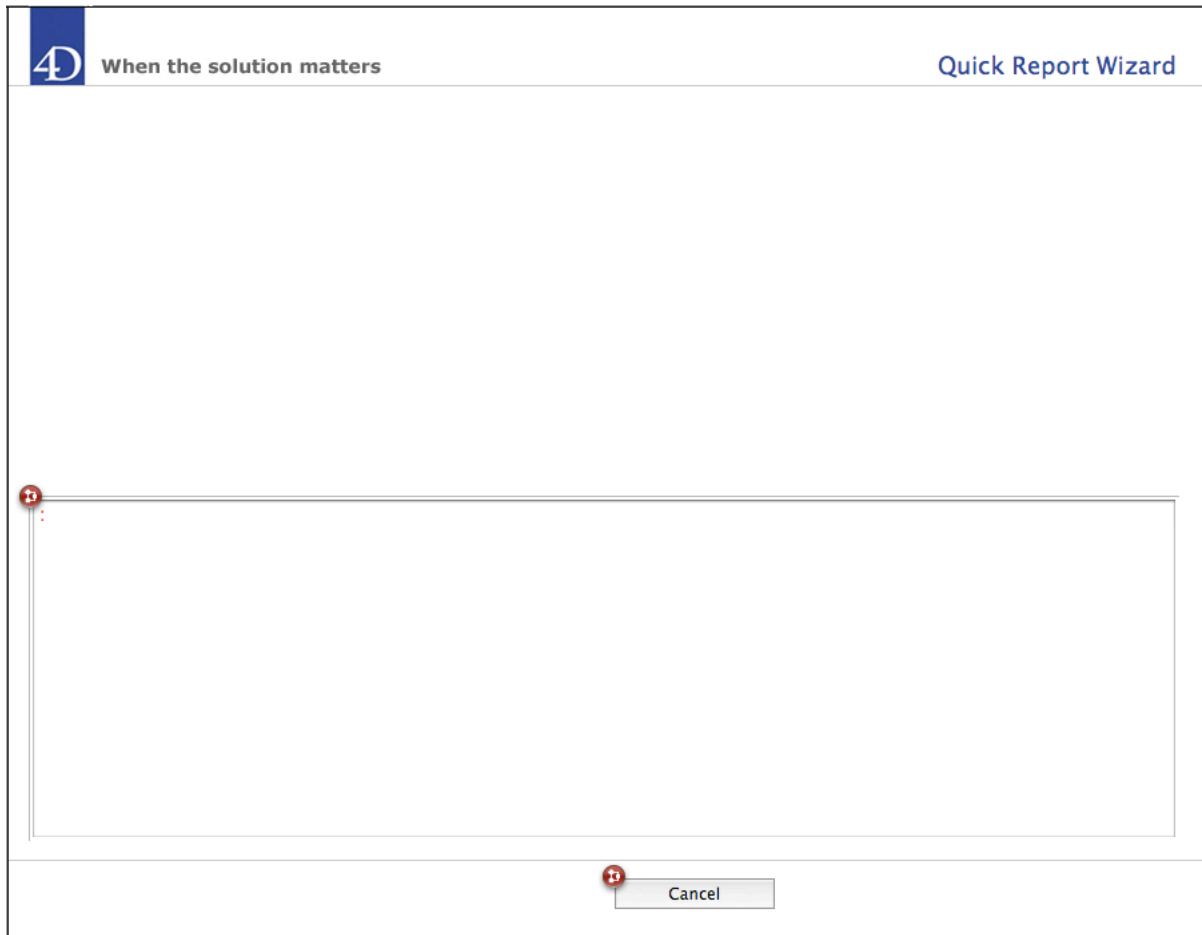
Cross-Tab Reports display information as a two-dimensional table. You can display data from a data source as categories that are functions of two other data sources.

Quick Report Plug-in

In order to create a Quick Report interface we need to use the Report Plug-in Area. The functionality of this plug-in is similar to any other 4D v11 SQL plug-in. The main difference is that it is not an external plug-in; it has been integrated into 4D v11 SQL. As a result, there is no need to install it in the Plugins folder and removing it is not an option.

The Report Plug-in is similar to other 4D v11 SQL Plug-ins that have been integrated into 4D, such as Web Area or 4D Chart. To use the Report Plug-in, all you have to do is create a form and add the Report Plug-in. In this Technical Note,

the Report Plug-in is displayed using the command DIALOG do build the custom interface.



Quick Report Commands

4D v11 SQL offers over 40 commands to programmatically create and control all the functions of a Quick Report. For more detailed information about these commands, you can consult the Quick Report section of the 4D v11 SQL Language Reference Manual or <http://www.4d.com/docs/V6U/V6U00039.HTM>.

This Technical Note analyzes the use of these commands to create the functionality of the Quick Report Generator in your own application.

Most of the Quick Report commands require the QR Plugin Area as the first parameter. The Object Name in the sample database for the Area is: QRBuilder_Obj and the Variable Name is: QRBuilder. The Quick Report commands require the Variable Name. For example, if we wanted to place our report in a field, we would use the following command:

```
QR REPORT TO BLOB (QRBuilder;[Table]BLOB_Field)
```

As we review each step in the process of creating List Reports and Cross-Tab report, all the Quick Reports commands used in the process are explained in detail.

List Reports in the Sample Database

The sample database included with this Technical Note illustrates the idea of creating List Reports using Quick Report Commands and a custom interface. In the steps required to create the report, the actual code and commands are described in detail. You can review the process and decide whether to add features to enhance the reports or eliminate options to simplify them.

The first screen simply allows the user to decide the type of report they want and where to obtain the information to create the report.

When the solution matters Quick Report Wizard

Select the type of report you want to print:

☒ **List Report**
To print a List Report only the fields that are going to be printed.

Company	Rank	Revenue (Millions)
Exxon Mobil	1	442,851
Wal-Mart Stores	2	405,607
Chevron	3	263,159
ConocoPhillips	4	230,764
General Electric	5	183,207

☐ **Cross-Tab Report**
To print a Crosstab Report, three sets of values need to be defined: the row, the column and one for computing the data.

	Jan	Feb	Total
Exxon	200	330	530
Chevron	150	100	250
Grand Total	350	430	780

For the selected table: [Clients] 50 Clients

Title
Title
Detail
Grand total

Cancel Next

The QR commands used in this page are initialization commands used in the On Load Form event.

```
QR EXECUTE COMMAND (QRBuilder;qr cmd new)
```

Initializes the QR Are for a new report using the **qr cmd new** constant.

```
QR ON COMMAND (QRBuilder;"Wizard_OnCommand")
```

This command simply redirects the user-selected Quick Report commands to a method called **Wizard_OnCommand**.

Essentially what this method accomplishes is to redirect the QR commands to a single method and in some cases add more functionality to the selected commands and in others we simply let 4D execute the command without any modifications.

```

.....
` Project Method: Wizard_OnCommand
` Description: Redirect User Commands
` Called by: On Load/Wiz Form
` Calls: Nothing
` Parameters: $1 - Longin - QR Area
`             $2 - Longin - Command
` Returns: Nothing
` Created by Jean-Yves Fock-Hoon
` Modified by Luis Piñeiros
` 2009
.....

C_LONGINT($1;$2)
C_LONGINT($SelectedCommand;$Area)
C_LONGINT($DropCol;$IsHere;$IsHere1;$IsHere2;$b;$iCol)

C_TEXT($vComStat)

$SelectedCommand:=$2
$Area:=$1
C_REAL($left;$top;$right;$bottom)

Case of
: (vPage=2)
` define columns
  Case of
  : ($SelectedCommand=2600) ` Insert column
  : ($SelectedCommand=2601) ` Delete column
    QR GET SELECTION(QRBuilder;$left;$top;$right;$bottom)
    $DropCol:=$left
    $IsHere:=Find in array(al_Sort_ColID;$DropCol)
    If ($IsHere>0)
      DELETE FROM ARRAY(at_Sort_ColText;$DropCol;1)
      DELETE FROM ARRAY(ap_Sort_ColPict;$DropCol;1)
      DELETE FROM ARRAY(al_Sort_ColOrder;$DropCol;1)
      DELETE FROM ARRAY(al_Sort_ColID;$DropCol;1)
      ` Columns ID are no longer the same Col 3 may become now 2
      ` Need to be recomputed
      For ($iCol;1;Size of array(al_Sort_ColID))
        If (al_Sort_ColID{$iCol}>$DropCol)
          al_Sort_ColID{$iCol}:=al_Sort_ColID{$iCol}-1
        End if
      End for
      QR EXECUTE COMMAND ($Area;$SelectedCommand)

      QR SET SORTS(QRBuilder;al_Sort_ColID;al_Sort_ColOrder)
    End if
  : ($SelectedCommand=2602) ` Hide column

```

```

: ($SelectedCommand=2603) ` Edit column
  QR EXECUTE COMMAND($Area;$SelectedCommand)
  If (OK=1)
    QR GET SELECTION(QRBuilder;$left;$top;$right;$bottom)

    $IsHere:=Find in array(al_Sort_ColID;$left)
    If ($IsHere>0)
      QR GET INFO COLUMN (QRBuilder;$left;vTitle;vObject;vHide;
vAutoSize;vRepeatedValue;vFormat)
      at_Sort_ColText{$IsHere}:=vObject
    End if
  End if

: ($SelectedCommand=2606) ` Format
  QR GET SELECTION(QRBuilder;$left;$top;$right;$bottom)
  $b:=QR Get command status(QRBuilder;2606;$vComStat)

  Case of
    : (($left=2) & ($top=2))
      vFormatDataSource:=$vComStat
    : (($left=1) & ($top=2))
      vFormatRows:=$vComStat
    : (($left=2) & ($top=1))
      vFormatColumns:=$vComStat
  End case

: ($SelectedCommand=3002) ` Move column Left
  QR GET SELECTION(QRBuilder;$left;$top;$right;$bottom)

  $IsHere2:=Find in array(al_Sort_ColID;$left)
  $IsHere1:=Find in array(al_Sort_ColID;$left-1)

  QR EXECUTE COMMAND($Area;$SelectedCommand)
  If ($IsHere2>0)
    al_Sort_ColID{$IsHere2}:=al_Sort_ColID{$IsHere2}-1

  End if
  If ($IsHere1>0)
    al_Sort_ColID{$IsHere1}:=al_Sort_ColID{$IsHere1}+1
  End if
  QR SET SORTS(al_Sort_ColID;al_Sort_ColOrder)

: ($SelectedCommand=3003) ` Move column Right
  QR GET SELECTION(QRBuilder;$left;$top;$right;$bottom)

  $IsHere2:=Find in array(al_Sort_ColID;$left)
  $IsHere1:=Find in array(al_Sort_ColID;$left+1)

  QR EXECUTE COMMAND($Area;$SelectedCommand)
  If ($IsHere2>0)
    al_Sort_ColID{$IsHere2}:=al_Sort_ColID{$IsHere2}+1

  End if
  If ($IsHere1>0)
    al_Sort_ColID{$IsHere1}:=al_Sort_ColID{$IsHere1}-1
  End if
  QR SET SORTS(al_Sort_ColID;al_Sort_ColOrder)

End case

If(($SelectedCommand#2605) & ($SelectedCommand#2604) & ($SelectedCommand#2601)
& ($SelectedCommand#2603) & ($SelectedCommand#3002) & ($SelectedCommand#3003))

```

```

    QR EXECUTE COMMAND ($Area;$SelectedCommand)
End if

: (vPage=3)
` Edit format and style attributes
If (($SelectedCommand>=506) & ($SelectedCommand<=510) |
($SelectedCommand=2604))
    QR EXECUTE COMMAND ($Area;$SelectedCommand)

Else
    BEEP
End if

: (vPage=4)
` Action in break levels
If ($SelectedCommand=2610)
    QR EXECUTE COMMAND ($Area;$SelectedCommand)
Else
    BEEP
End if

: (vPage=5)
` Edit format and style attributes
If ($SelectedCommand=2606)
    QR EXECUTE COMMAND ($Area;$SelectedCommand)
Else
    BEEP
End if

: (vPage=6)
` Edit format and style attributes
If (($SelectedCommand=1000) | (($SelectedCommand>=2308) &
($SelectedCommand<=2324)) | (($SelectedCommand>=500) &
($SelectedCommand<=505)) | (($SelectedCommand>=511) &
($SelectedCommand<=512)) | ($SelectedCommand=1002) |
($SelectedCommand=1003) | ($SelectedCommand=1005))
    QR EXECUTE COMMAND ($Area;$SelectedCommand)
Else
    BEEP
End if

: (vPage=7)
` Cells border
If ($SelectedCommand=2609)
    QR EXECUTE COMMAND ($Area;$SelectedCommand)
Else
    BEEP
End if

: (vPage=8)
` Width columns
If ($SelectedCommand=2605)
    QR EXECUTE COMMAND ($Area;$SelectedCommand)
Else
    BEEP
End if

: (vPage=9)
` Header
If ($SelectedCommand=2005)
    QR EXECUTE COMMAND ($Area;$SelectedCommand)
Else
    BEEP
End if

: (vPage=10)
` Footer

```



```

If ($SelectedCommand=2005)
    QR EXECUTE COMMAND($Area;$SelectedCommand)
Else
    BEEP
End if

: (vPage=11)
    \ Destination / printing
    \ 2500-2504;
Case of
    : (($SelectedCommand>=2500) & ($SelectedCommand<=2508))
        QR EXECUTE COMMAND($Area;$SelectedCommand)
    : (($SelectedCommand=2006) | ($SelectedCommand=2007) |
($SelectedCommand=2008))
        QR EXECUTE COMMAND($Area;$SelectedCommand)
Else
    BEEP
End case

    \
    \ Crosstab
    \

: (vPage=19)
    \ Computation Data source
    If (($SelectedCommand>=506) & ($SelectedCommand<=510))
        QR EXECUTE COMMAND($Area;$SelectedCommand)
    Else
        BEEP
    End if

: (vPage=20)
    \ Computation Total
    If (($SelectedCommand>=506) & ($SelectedCommand<=510))
        QR EXECUTE COMMAND($Area;$SelectedCommand)
    Else
        BEEP
    End if

: (vPage=21)
    \ Computation Grtand Total
    If (($SelectedCommand>=506) & ($SelectedCommand<=510))
        QR EXECUTE COMMAND($Area;$SelectedCommand)
    Else
        BEEP
    End if

: (vPage=24)
    If((($SelectedCommand=1000) | (($SelectedCommand>=2308) &
($SelectedCommand<=2324)) | (($SelectedCommand>=500) &
($SelectedCommand<=505)) | (($SelectedCommand>=511) &
($SelectedCommand<=512)) | ($SelectedCommand=1002) |
($SelectedCommand=1003) | ($SelectedCommand=1005))
        QR EXECUTE COMMAND($Area;$SelectedCommand)
    Else
        BEEP
    End if

: (vPage=25)
    \ Cells border
    If ($SelectedCommand=2609)
        QR EXECUTE COMMAND($Area;$SelectedCommand)
    Else
        BEEP
    End if

: (vPage=26)
    \ Width columns
    If ($SelectedCommand=2605)

```

```

        QR EXECUTE COMMAND ($Area;$SelectedCommand)
    Else
        BEEP
    End if

End case
REDRAW WINDOW

```

The selected table drop-down menu in the **at_CurrentTable** variable:

This method makes all the records in the table selected active for a Quick Report and puts the number of records and the table name in a variable.

```

QR SET REPORT TABLE (QRBuilder;at_CurrentTable)

If (((report1+report2)=1) & (at_CurrentTable#0))
    ENABLE BUTTON (bNextpage)
End if
QRWrbAllRecords:=1
ALL RECORDS (Table (at_CurrentTable)->)
QRWvNumRecs:=String(Records in selection (Table (at_CurrentTable)->))+
"+Substring (at_CurrentTable{at_CurrentTable};2;Length (at_CurrentTable{at_Curren
tTable})) -2)

```

The All Records radio button in the **QRWrbAllRecords** variable:

If you click on Query and select records in the current table, you can Click on All Records to default back to using all the records in the current table for your Quick Report.

```

ALL RECORDS (Table (at_CurrentTable)->)
QRWvNumRecs:=String(Records in selection (Table (at_CurrentTable)->))+
"+Substring (at_CurrentTable{at_CurrentTable};2;Length (at_CurrentTable{at_Curren
tTable})) -2)

```

The **Query** button:

It displays the Query Editor window to select records for the Quick Report via a Query.

```

QRWrbAllRecords:=0
QUERY (Table (at_CurrentTable)->)
QRWvNumRecs:=String(Records in selection (Table (at_CurrentTable)->))+
"+Substring (at_CurrentTable{at_CurrentTable};2;Length (at_CurrentTable{at_Curren
tTable})) -2)

```

Step 1: Select your fields and the sorting order

The Tables and Fields HList is managed by the **Wizard_Load_HLTables** method.

This method loads the table fields and related table fields in the HLTables HList.

```

.....
Project Method: Wizard_Load_HLTables
Description: Loads HL Tables
Called by: Wiz Form
Calls: Wizard_Load_related_fields
Parameters: None
Returns: Nothing
Created by Jean-Yves Fock-Hoon
Modified by Luis Piñeiros
2009
.....

C_LONGINT($Ref;$ItemPos)

Case of
: (at_showfields=1)
  Table only
  $Ref:=Find in array(∅at_CurrentTable;at_CurrentTable{at_CurrentTable})
  If ($Ref#-1)
    $Ref:=∅at_CurrentTableID{$Ref}

```

```

        SELECT LIST ITEMS BY REFERENCE (ØHL_StructureDef;$Ref)
        $ItemPos:=Selected list items (ØHL_StructureDef)
        GET LIST ITEM (ØHL_StructureDef;$ItemPos;$ItemRef;$ItemText;HLTables)
        SAVE LIST (hltables;"Tables")

    End if

: (at_showfields=2)
    \ Table and related fields
    Wizard_Load_related_fields

: (at_showfields=3)
    \ All Tables
    HLTables:=ØHL_StructureDef

End case
REDRAW LIST (HLTables)

```

The sort order is managed by the **Wiz_Sort** method.

This method handles the Command-click or Control-Click to delete a sort from the list. It also handles the On Drop Form event to add fields to the at_Sort_ColText Scrollable Area and the ap_Sort_ColPict Scrollable Area.

```

\ .....
\ Project Method:  Wizard_Sort
\ Description:  Manages Sort Order in List Report
\ Called by:  Wiz Page
\ Calls:  Nothing
\ Parameters:  None
\ Returns:  Nothing
\ Created by  Jean-Yves Fock-Hoon
\ Modified by Luis Piñeiros
\ 2009
\ .....

C_TEXT($vtitle;$object;$vformat;$format)
C_REAL($vHide;$vAutoSize;$vRepeatedValue)
C_BOOLEAN($expanded;$Found)
C_LONGINT($loopColumns;$ColID)
C_TEXT($vObject;$NewObject)

Case of
: (Form event=On Clicked )
    If (Windows Ctrl down)
        If (at_Sort_ColText>0)
            DELETE FROM ARRAY (at_Sort_ColText;at_Sort_ColText;1)
            DELETE FROM ARRAY (ap_Sort_ColPict;at_Sort_ColText;1)
            DELETE FROM ARRAY (al_Sort_ColOrder;at_Sort_ColText;1)
            DELETE FROM ARRAY (al_Sort_ColID;at_Sort_ColText;1)

            QR SET SORTS (QRBuilder;al_Sort_ColID;al_Sort_ColOrder)
        End if
    End if

: (Form event=On Drop )
    DropPos:=Drop position
    DRAG AND DROP PROPERTIES (srcObject;SourceElement;srcProcess)
    If ((srcObject)=(->HLTables))

```

```

        GET LIST ITEM(HLTables;SourceElement;$ItemRef;$NewObject)

        If (DropPos=-1)
            DropPos:=Size of array(at_Sort_ColText)+1
        End if

        $Found:=False
        For ($loopColumns;1;QR Count columns(QRBuilder))
            QR GET INFO COLUMN(QRBuilder;$loopColumns;$vTitle;$vObject;$vHide;
                $vAutoSize;$vRepeatedValue;$format)
                If ($vObject=$NewObject)
                    $Found:=True
                    $ColID:=$loopColumns
                    $loopColumns:=QR Count columns(QRBuilder)+1
                End if
            End for

        If (Not($Found))
            QR INSERT COLUMN(QRBuilder;QR Count columns(QRBuilder)+1;$NewObject)
            $ColID:=QR Count columns(QRBuilder)

        End if

        INSERT IN ARRAY(at_Sort_ColText;DropPos;1)
        INSERT IN ARRAY(ap_Sort_ColPict;DropPos;1)
        INSERT IN ARRAY(al_Sort_ColOrder;DropPos;1)
        INSERT IN ARRAY(al_Sort_ColID;DropPos;1)
        at_Sort_ColText{DropPos}:=$NewObject
        ap_Sort_ColPict{DropPos}:=$vPict_Up
        al_Sort_ColOrder{DropPos}:=1
        al_Sort_ColID{DropPos}:=$ColID

        End if
        QR SET SORTS(QRBuilder;al_Sort_ColID;al_Sort_ColOrder)
    End case

```

Step 2: Define the values inside your break levels

Break levels get defined in the Area itself. No methods are required at this stage.

When the solution matters Quick Report Wizard

Define the values inside your break levels: List Report

You can define a computation on each break level or grand total line by selecting the cell and clicking on the appropriate icon in the computation toolbar. You can also embed these values inside a cell. Embed values by writing a sentence that uses the following codes:

#	Value that generated the break level	##S	Sum	##X	Max
##z	Current value of the z th column	##A	Average	##C	Count
Ex: ##5	Current value for column 5	##N	Min		

You can also check the 'repeated value' button in order to repeat your values when printing the breaks.

	[Clients]Company_Nam	[Clients]Fortune500_Ran	[Clients]Revenue_inMillio
Title	Company_Name	Fortune500_Rank	Revenue_inMillions
Detail			
[Clients]Fortune500_Rank changed			
Grand total	##C		

2 Cancel Back Next

Step 3: Define an action in your break levels

When the solution matters Quick Report Wizard

Define an action in your break levels: List Report

You can define an action such as a form feed or re-print the titles after each break level.

Enter the following values to specify Break actions:

- No action: 0
- Break page: 32000
- Spacing of X points: X (from 1 up to 31999)

Break Action: Get Action Set Action

Title	[Clients]Company_Nam	[Clients]Fortune500_Ran	[Clients]Revenue_inMillio
Title	Company_Name	Fortune500_Rank	Revenue_inMillions
Detail			
[Clients]Fortune500_Rank changed			
Grand total			

3 Cancel Back Next

Two buttons are required: **Get Action**

```

C_REAL($left;$top;$right;$bottom)

QR GET SELECTION(qrBuilder;$left;$top;$right;$bottom)
ARRAY LONGINT($arr1;0)
ARRAY LONGINT($arr2;0)

QR GET SORTS(qrbuilder;$arr1;$arr2)

Case of
: ($top=1)
    $top:=-1
: ($top=2)
    $top:=-2
: ($top>(Size of array($arr1)+2))
    $top:=-3
: ($top=0)
    Else
        $top:=$top-2
End case
If ($top>0)
    QR GET TOTALS SPACING(QRBuilder;$top;vBreakValue)
End if

```

and **Set Action**.

```

C_REAL($left;$top;$right;$bottom)
QR GET SELECTION(qrBuilder;$left;$top;$right;$bottom)
ARRAY LONGINT($arr1;0)
ARRAY LONGINT($arr2;0)

QR GET SORTS(qrbuilder;$arr1;$arr2)

Case of
  : ($top=1)
    $top:=-1
  : ($top=2)
    $top:=-2
  : ($top>(Size of array($arr1)+2))
    $top:=-3
  : ($top=0)
    Else
      $top:=$top-2
End case
If ($top>0)
  QR SET TOTALS SPACING(QRBuilder;$top;vBreakValue)
End if

```

In addition a Break Action variable that get used in both methods to determine the spacing above a subtotal row using the **QR SET TOTAL SPACING** command.

Step 4: Define a format for your cells

Cell formats gets defined in the Area itself. No methods are required at this stage.

4 When the solution matters Quick Report Wizard

Define a format for your cells: List Report

The Quick Report needs to format your data.
If no format is chosen, it will use the default one in 4D. A format is defined for the whole column. You can use available formats that 4D provides or use your own for text and numerical values.

If you want to define a format for your texts or numerical values, you can use the #, 0, * or ^ characters to define them. The characters defined in your format will be replaced by one of the characters of the value to be printed. Be sure that you have enough characters in your format to print the whole value. The # character will print the character, if any, while the other characters will print the character, if any, or a 0 or a space or a * if there are no more characters available to be printed. Date, time and picture fields have pre-defined formats.

A format will be used for each value for this column, even if you do want to print it into another column when using the #z defined in the previous step.

Title	[Clients]Company_Nam	[Clients]Fortune500_Ran	[Clients]Revenue_inMillion:
Title	Company_Name	Fortune500_Rank	Revenue_inMillions
Detail			\$###,##0.00;-\$###,##0.00
[Clients]Fortune500_Rank changed			
Grand total	##C		

4 Cancel Back Next

Step 5: Define the style attributes for your cells

Cell styles get defined in the Area itself. No methods are required at this stage.

4D When the solution matters Quick Report Wizard

Define the style attributes for your cells: List Report

Styles attributes can be defined for each cell.

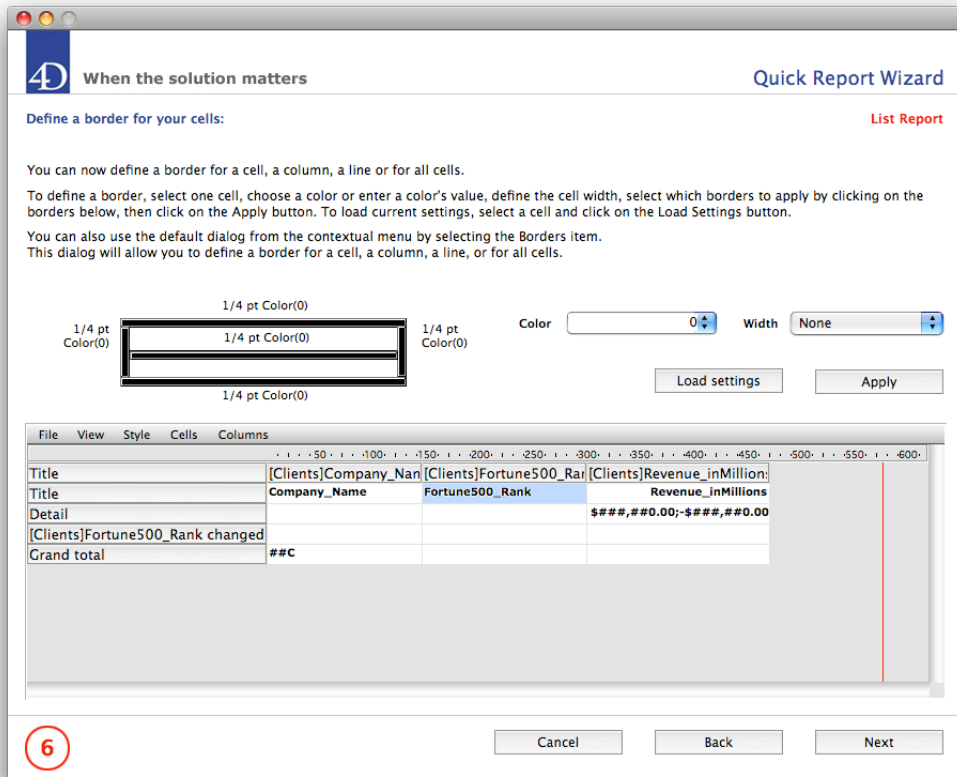
You can define your own styles such as font name, size, bold, italic and underline attributes, main color, justification or background color.

The background color can be alternated.
The first color will be used when printing even lines while the alternate background color will be used for odd lines.

Title	[Clients]Company_Nam	[Clients]Fortune500_Rar	[Clients]Revenue_inMillion:
Title	Company_Name	Fortune500_Rank	Revenue_inMillions
Detail			###,##0.00;-###,##0.00
[Clients]Fortune500_Rank changed			
Grand total	##C		

5 Cancel Back Next

Step 6: Define a border for your cells



To define the color the **at_Color3** drop-down list is required:

```
C_LONGINT ($value)
If (Self->>0)
    $value:=Self->
    vColorLine:=Num (at_ColorID {$value})
End if
```

To define the width the **at_WidthLine** drop-down list is required.

In addition, the **Load Settings** button:

This method gets the break levels, set default information about the borders and set the default RGB colors for the lines of the borders.

```
C_LONGINT ($left;$top;$right;$bottom;$Col;$Row)

QR GET SELECTION (qrBuilder;$left;$top;$right;$bottom)
ARRAY LONGINT ($arr1;0)
ARRAY LONGINT ($arr2;0)

QR GET SORTS (qrbuilder;$arr1;$arr2)
Case of
: ($top=1)
    $top:=-1
```

```

: ($stop=2)
  $stop:=-2
: ($stop>(Size of array($arr1)+2))
  $stop:=-3
: ($stop=0)
Else
  $stop:=$stop-2
End case

$Col:=$left
$Row:=$stop

C_TEXT(vTextBorder)
vTextBorder:=""
C_LONGINT($Color;$Size;$Color1;$Color2;$Color3;$Color4;$Color5)
vRight:=""
vLeft:=""
vTop:=""
vBottom:=""
ARRAY TEXT($arr_val;5)
$arr_val{1}:="none"
$arr_val{2}:="1/4 pt"
$arr_val{3}:="1/2 pt"
$arr_val{4}:="1 pt"
$arr_val{5}:="2 pts"

If (($left#0) & ($stop#0))
  QR GET BORDERS(qrBuilder;$Col;$Row;1;$Size;$Color1) ` Left border
  vLeft:=$arr_val{$Size+1}+" Color("+String($Color1)+") "
  QR GET BORDERS(qrBuilder;$Col;$Row;2;$Size;$Color2) ` Top border
  vTop:=$arr_val{$Size+1}+" Color("+String($Color2)+") "
  QR GET BORDERS(qrBuilder;$Col;$Row;4;$Size;$Color3) ` Right border
  vRight:=$arr_val{$Size+1}+" Color("+String($Color3)+") "
  QR GET BORDERS(qrBuilder;$Col;$Row;8;$Size;$Color4) ` Bottom border
  vBottom:=$arr_val{$Size+1}+" Color("+String($Color4)+") "
  QR GET BORDERS(qrBuilder;$Col;$Row;32;$Size;$Color5) ` Center border
  vCenter:=$arr_val{$Size+1}+" Color("+String($Color5)+") "

  SET RGB COLORS(*;"RectLeftLine";$color1;$color1)
  SET RGB COLORS(*;"RectTopLine";$Color2;$Color2)
  SET RGB COLORS(*;"RectRightLine";$Color3;$Color3)
  SET RGB COLORS(*;"RectBottomLine";$Color4;$Color4)
  SET RGB COLORS(*;"RectCenterLine";$Color5;$Color5)

Else
  BEEP
End if

```

And the **Apply** button:

This method applies the selected information to the borders.

```

C_LONGINT($left;$stop;$right;$bottom;$Color1;$Color2;$Color3;$Color4;$Color5;$Size)
C_LONGINT($Col;$Row;$NewSize;$vColorLine)

ARRAY TEXT($arr_val;5)
$arr_val{1}:="none"
$arr_val{2}:="1/4 pt"

```

```

$arr_val{3}:="1/2 pt"
$arr_val{4}:="1 pt"
$arr_val{5}:="2 pts"

If (report1=1)
    ` List
    ARRAY LONGINT($arr1;0)
    ARRAY LONGINT($arr2;0)

    QR GET SELECTION(qrBuilder;$left;$top;$right;$bottom)
    QR GET SORTS(qrbuilder;$arr1;$arr2)
    Case of
        : ($top=1)
            $top:=-1
        : ($top=2)
            $top:=-2
        : ($top>(Size of array($arr1)+2))
            $top:=-3
        : ($top=0)
            Else
                $top:=$top-2
    End case
    $Col:=$left
    $Row:=$top
Else
    ` Cross table
    $Col:=at_CTColumn
    $Row:=at_CTLine
End if
$vColorLine:=vColorLine
$NewSize:=at_WidthLine-1
If (($left>0) & ($top#0))
    ` A cell has been selected
    If (vLeftLine=1)
        SET RGB COLORS(*;"RectLeftLine";$vColorLine;0)
        QR SET BORDERS(qrBuilder;$Col;$Row;1;$NewSize;$vColorLine) ` Left border
        vLeft:=$arr_val{$NewSize+1}+" Color("+String($vColorLine)+") "
    Else
        QR GET BORDERS(qrBuilder;$Col;$Row;1;$Size;$Color1) ` Left border
        vLeft:=$arr_val{$Size+1}+" Color("+String($Color1)+") "
    End if
    If (vTopLine=1)
        SET RGB COLORS(*;"RectTopLine";$vColorLine;0)
        QR SET BORDERS(qrBuilder;$Col;$Row;2;$NewSize;$vColorLine) ` Top border
        vTop:=$arr_val{$NewSize+1}+" Color("+String($vColorLine)+") "
    Else
        QR GET BORDERS(qrBuilder;$Col;$Row;2;$Size;$Color2) ` Top border
        vTop:=$arr_val{$Size+1}+" Color("+String($Color2)+") "
    End if
    If (vRightLine=1)
        SET RGB COLORS(*;"RectRightLine";$vColorLine;0)
        QR SET BORDERS(qrBuilder;$Col;$Row;4;$NewSize;$vColorLine) ` Right border
        vRight:=$arr_val{$NewSize+1}+" Color("+String($vColorLine)+") "
    Else
        QR GET BORDERS(qrBuilder;$Col;$Row;4;$Size;$Color3) ` Right border
        vRight:=$arr_val{$Size+1}+" Color("+String($Color3)+") "
    End if
    If (vBottomLine=1)
        SET RGB COLORS(*;"RectBottomLine";$vColorLine;0)
        QR SET BORDERS(qrBuilder;$Col;$Row;8;$NewSize;$vColorLine) ` Bottom border
        vBottom:=$arr_val{$NewSize+1}+" Color("+String($vColorLine)+") "
    Else

```

```

QR GET BORDERS (qrBuilder;$Col;$Row;8;$Size;$Color4) ` Bottom border
vBottom:=$arr_val{$Size+1}+" Color("+String($Color4)+") "

End if

If (vCenterLine=1)
SET RGB COLORS (*;"RectCenterLine";$vColorLine;0)
QR SET BORDERS (qrBuilder;$Col;$Row;32;$NewSize;$vColorLine) ` Center border
vCenter:=$arr_val{$NewSize+1}+" Color("+String($vColorLine)+") "
Else
QR GET BORDERS (qrBuilder;$Col;$Row;32;$Size;$Color5) ` Center border
vCenter:=$arr_val{$Size+1}+" Color("+String($Color5)+") "

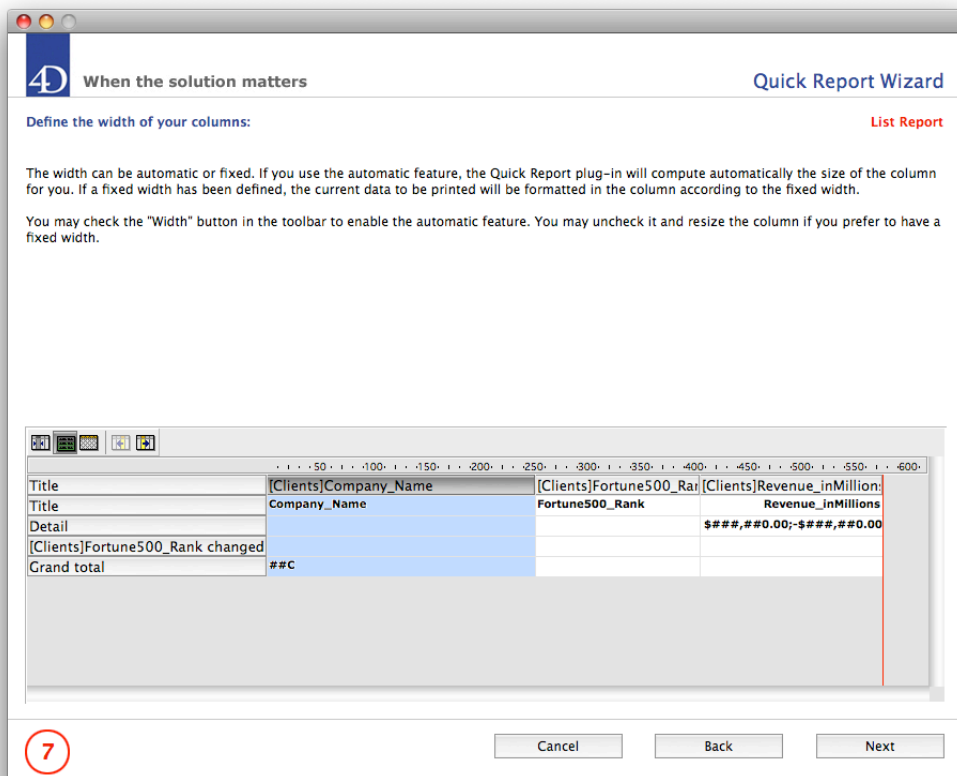
End if

Else
BEEP
End if

```

Step 7: Define the width of your columns

Defining the width of your columns is accomplished in the Area itself. No methods are required at this stage.



Step 8 - 9: Define the page header and footer for your report

Quick Report Wizard

Define the page header of your report: List Report

You can define the page header by requesting the appropriate dialog in the 'File' menu, item 'Header and Footer' from the menu bar in the Quick Report area, or use the following panel:

Header

Left text: Font: Picture:

Center text:

Right text:

Height: ☐ On the left ☒ On the center ☐ On the right

Size:

Style: ☒ Bold ☐ Italic ☐ Underline

File View Style Cells Columns

Title	(Clients)Company_Name	(Clients)Fortune500_Rai	(Clients)Revenue_inMillion:
Title	Company_Name	Fortune500_Rank	Revenue_inMillions
Detail			###,##0.00;-###,##0.00
(Clients)Fortune500_Rank changed			
Grand total	##C		

8

The **Set Header** button is required:

This method takes the information from the Header section and applies it to the Quick Report.

```
vPictAlignment:=Num(Alignment1=1)+(Num(Alignment2=1)*2)+(Num(Alignment3=1)*3)
QR SET HEADER AND FOOTER(QRBuilder;1;vTextLeft;vTextCenter;vTextRight;
vHeight;VPictHeader;vPictAlignment)
vFont:=Font number(at_FontName{at_FontName})
QR SET TEXT PROPERTY(QRBuilder;0;-4;1;vFont)
QR SET TEXT PROPERTY(QRBuilder;0;-4;2;vSize)
QR SET TEXT PROPERTY(QRBuilder;0;-4;3;vStyleBold)
QR SET TEXT PROPERTY(QRBuilder;0;-4;4;vStyleItalic)
QR SET TEXT PROPERTY(QRBuilder;0;-4;5;vStyleUnderline)
```

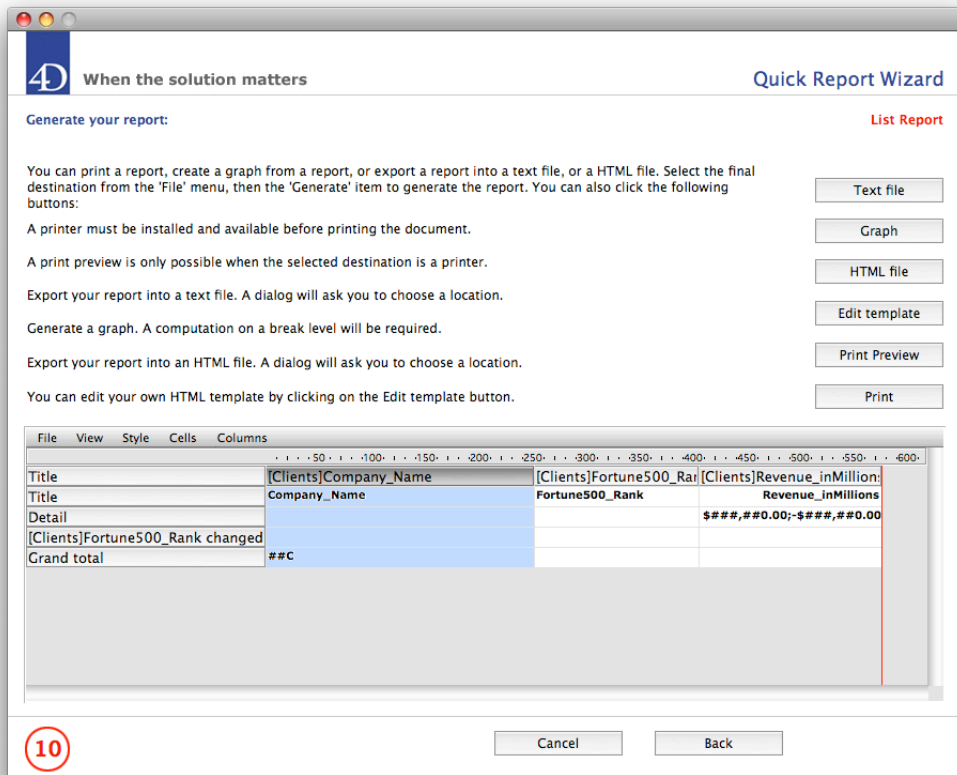
The **Set Footer** button is also required:

This method takes the information from the Footer section and applies it to the Quick Report.

```
vPictAlignment:=Num(Alignment1=1)+(Num(Alignment2=1)*2)+(Num(Alignment3=1)*3)
QR SET HEADER AND FOOTER(QRBuilder;2;vTextLeft;vTextCenter;vTextRight;
vHeight;VPictFooter;vPictAlignment)
```

```
vFont:=Font number (at_FontName{at_FontName})
QR SET TEXT PROPERTY (QRBuilder;0;-4;1;vFont)
QR SET TEXT PROPERTY (QRBuilder;0;-4;2;vSize)
QR SET TEXT PROPERTY (QRBuilder;0;-4;3;vStyleBold)
QR SET TEXT PROPERTY (QRBuilder;0;-4;4;vStyleItalic)
QR SET TEXT PROPERTY (QRBuilder;0;-4;5;vStyleUnderline)
```

Step 10: Generate your Report



The following options are available:

Text file:

```
QR SET DESTINATION (QRBuilder;2)
QR RUN (QRBuilder)
```

Graph:

```
QR SET DESTINATION (QRBuilder;4)
QR RUN (QRBuilder)
```

HTML file:

```
QR SET DESTINATION (QRBuilder;5)
QR RUN (QRBuilder)
```


Edit template:

```
QR SET DESTINATION(QRBuilder;5)

vHTMLTemplate:=QR Get HTML template(QRBuilder)
vPage:=vPage+1
Wizard_Changepage
```

Print Preview:

```
QR SET DESTINATION(QRBuilder;1)
QR EXECUTE COMMAND(QRBuilder;qr cmd print preview)
```

Print:

```
QR SET DESTINATION(QRBuilder;1)
QR RUN(QRBuilder)
```

Cross-Tab Reports in the Sample Database

The Cross-Tab Reports included in the sample database shares some of the code used in the List Reports. When that is the case, the appropriate step in the List Reports section is referenced for the method used.

When selecting Cross-Tab Reports, the first page of the sample database uses the same code to initialize the form, the table drop-down menu and the Query button as in the List Reports.

When the solution matters Quick Report Wizard

Select the type of report you want to print:

☐ List Report
To print a List Report only the fields that are going to be printed.

Company	Rank	Revenue (Millions)
Exxon Mobil	1	442,851
Wal-Mart Stores	2	405,607
Chevron	3	263,159
ConocoPhillips	4	230,764
General Electric	5	183,207

☒ Cross-Tab Report
To print a Crosstab Report, three sets of values need to be defined: the row, the column and one for computing the data.

	Jan	Feb	Total
Exxon	200	330	530
Chevron	150	100	250
Grand Total	350	430	780

For the selected table: [Sales]

☒ All Records 300 Sales Query

Title
Title
Detail
Grand total

Cancel Next

Step 1-2-3: Define a format for your cells

The first 3 steps share the **at_Showfields** drop-down menu. It uses the method **Wizard_Load_HLTables** already described in Step 1 of the List Reports.

When the solution matters

Quick Report Wizard

Select a field for columns: Crosstab Report

Fields

This will define the titles of columns. Each record will be parsed and each value for this field may create a new column if the value does not exist yet.

Select a field for columns

- [Sales]Tracking_No
- [Sales]Sales_ID
- [Sales]Client_ID
- [Sales]Sales_Amount
- [Sales]Sales_Date
- [Sales]Invoice_No

Current table and related fields

Drop your choice here: [Sales]Sales_Date

	Column 1	Column 2	Total
Line 1			
Line 2			
Grand Total			

	[Sales]Sales_Date	Total
[Clients]Company_Name	[Sales]Sales_Amount	Σ Sum
Grand Total	Σ Sum	Σ Sum

1

Cancel Back Next

Step 1 uses the **vCTColumn** variable to drop the user's choice for the columns.

The methods for Steps 1 to 3 use the QR GET INFO COLUMN to obtain information about the columns and the QR SET INFO COLUMN to set the appropriate values to columns 1, 2 and 3.

```
C_BOOLEAN($expanded)
C_TEXT($title;$object;$vformat)
C_REAL($vHide;$vAutoSize;$vRepeatedValue)

Case of
: (Form event=On Drop )
    DRAG AND DROP PROPERTIES(SourceObject;SourceElement;SourceProcess)

    Case of
        : (SourceObject->=HLTables)
            GET LIST ITEM(HLTables;SourceElement;$ItemRef;Self->)

        Else
            Self->:=""

    End case
End case
```

```

C_REAL($vHide;$vAutoSize;$vRepeatedValue)

QR GET INFO COLUMN(QRBuilder;1;$title;$object;$vHide;$vAutoSize;
$vRepeatedValue;$vformat)
QR SET INFO COLUMN(QRBuilder;1;Self->;Self>;$vHide;$vAutoSize;
$vRepeatedValue;$vformat)

```

Step 2 uses the **vCTLine** variable to drop the user's choice for the rows.

```

C_BOOLEAN($expanded)
C_TEXT($Title)
C_TEXT($title;$object;$vformat)
C_REAL($vHide;$vAutoSize;$vRepeatedValue)

Case of
: (Form event=On Drop )
    DRAG AND DROP PROPERTIES(SourceObject;SourceElement;SourceProcess)

    Case of
    : (SourceObject->=HLTables)
        GET LIST ITEM(HLTables;SourceElement;$ItemRef;Self->)

    Else
        Self->:=""

    End case
End case

C_REAL($vHide;$vAutoSize;$vRepeatedValue)

QR GET INFO COLUMN(QRBuilder;2;$title;$object;$vHide;$vAutoSize;
$vRepeatedValue;$vformat)
QR SET INFO COLUMN(QRBuilder;2;Self->;Self->;$vHide;$vAutoSize;$vRepeatedValue;
$vformat)

```

Step 3 uses the **vCTDataSource** variable to drop the user's choice for the data source.

```

C_BOOLEAN($expanded)
C_TEXT($Title)
C_TEXT($title;$object;$vformat)
C_REAL($vHide;$vAutoSize;$vRepeatedValue)
Case of
: (Form event=On Drop )
    DRAG AND DROP PROPERTIES(SourceObject;SourceElement;SourceProcess)

    Case of
    : (SourceObject->=HLTables)
        GET LIST ITEM(HLTables;SourceElement;$ItemRef;Self->)

    Else
        Self->:=""

    End case
End case

C_REAL($vHide;$vAutoSize;$vRepeatedValue)

QR GET INFO COLUMN(QRBuilder;3;$title;$object;$vHide;$vAutoSize;
$vRepeatedValue;$vformat)
QR SET INFO COLUMN(QRBuilder;3;Self->;Self->;$vHide;$vAutoSize;$vRepeatedValue;
$vformat)

```

Step 4: Select a computation for Data Source cell

A computation for the data source cell gets defined in the Area itself. No methods are required at this stage.

4 When the solution matters Quick Report Wizard

Select a computation for Data Source cell: Crosstab Report

Computation

The field previously defined in the Data area will be used to perform these computations.

The Data Source cell data is the intersection cell between a column and a line.
You need to define a data source and at least one computation for this area.
If you do not provide a computation or nothing can be computed, the cell will be empty.

	Column 1	Column 2	Total
Line 1			
Line 2			
Grand Total			

Select the defined cell and choose the computations you want to perform from the toolbar or from the contextual menu.

File View Style Cells Columns

[Sales]Sales_Date Total

[Clients]Company_Name [Sales]Sales_Amount Sum

Grand Total Sum

4 Cancel Back Next

Step 5: Select a computation for the Total Column:

The **vTotalTitle** variable is used to enter a new title for the Total:

```
QR SET TOTALS DATA (QRBuilder;3;1;vTotalTitle)
```

The screenshot shows the 'Quick Report Wizard' window at step 5, titled 'When the solution matters'. The main instruction is 'Select the a computation for the Total column:'. A red label 'Crosstab Report' is in the top right. The 'Computation' section explains that the data source field will be used for computations and that the total is the far right column, which will not be printed if no calculation is defined. A small table illustrates this with columns 'Column 1', 'Column 2', and 'Total'. The 'Total' column for 'Line 1' and 'Line 2' is shaded with diagonal lines. Below this, a text box 'Enter a new title for Total:' contains the word 'Total'. A note at the bottom of the section says 'Select the defined cell and choose the computations you want to perform from the toolbar or from the contextual menu.' Below the instruction box is a preview window with a menu bar (File, View, Style, Cells, Columns) and a toolbar. The preview shows a table with columns '[Sales]Sales_Date', '[Sales]Sales_Amount', and 'Total'. The 'Total' column has a 'Sum' computation. The 'Grand Total' row also has a 'Sum' computation. At the bottom left, a red circle with the number '5' indicates the current step. At the bottom right are 'Cancel', 'Back', and 'Next' buttons.

4 When the solution matters Quick Report Wizard

Select the a computation for the Total column: Crosstab Report

Computation

The field defined as data source will be used to perform these computations.

Total is the far right column.
This column will not be printed if no calulation will be defined

	Column 1	Column 2	Total
Line 1			
Line 2			
Grand Total			

Enter a new title for Total: Total

Select the defined cell and choose the computations you want to perform from the toolbar or from the contextual menu.

File View Style Cells Columns

	[Sales]Sales_Date	Total
[Clients]Company_Name	[Sales]Sales_Amount	Sum
Grand Total	Sum	Sum

5 Cancel Back Next

Step 6: Select a computation for the Grand Total line

The **vGrandTotalTitle** variable is used to enter a title for Grand Total:

```
QR SET TOTALS DATA (QRBuilder;1;3;vGrandTotalTitle)
```

When the solution matters Quick Report Wizard

Select a computation for the Grand Total line: Crosstab Report

Computation

The field defined as data source will be used to perform these computations.

Grand Total is the last line that will be printed. It will contain calculations per column.
This line will not be printed if no calculation will be defined.
Calculations between Total and Grand Total will not be printed if no calculation has been defined for Total and Grand Total

	Column 1	Column 2	Total
Line 1			
Line 2			
Grand Total			

Enter a title for Grand Total:

Select the defined cell and choose the computations you want to perform from the toolbar or from the contextual menu.

File View Style Cells Columns

	[Sales]Sales_Date	Total
[Clients]Company_Name	[Sales]Sales_Amount	Σ Sum
Grand Total	Σ Sum	Σ Sum

6 Cancel Back Next

Step 7: Select the sorting order

Grouped radio buttons are used for the Rows and Columns. Row_Sort 1-3 for the rows and Column_Sort 1-3 for the Columns. The method **wizard_ctsetsort** is used to manage all the sort buttons.

This method uses the choices from the Rows and Columns radio button banks to put their values in two LongInt arrays and pass them as parameters in the QR SET SORTS command.

```
.....  
Project Method: Wizard_ctsetsort  
Description: Manages Sorts  
Called by: Wiz Form  
Calls: Nothing  
Parameters: None  
Returns: Nothing  
Created by Jean-Yves Fock-Hoon  
Modified by Luis Piñeiros  
2009  
.....
```

```

ARRAY LONGINT($al_Setsort;0)
ARRAY LONGINT($al_SetOrder;0)

Case of
: (Row_Sort1=1)
    INSERT IN ARRAY($al_Setsort;1;1)
    INSERT IN ARRAY($al_SetOrder;1;1)
    $al_Setsort{1}:=2
    $al_SetOrder{1}:=1
: (Row_Sort2=1)
    INSERT IN ARRAY($al_Setsort;1;1)
    INSERT IN ARRAY($al_SetOrder;1;1)
    $al_Setsort{1}:=2
    $al_SetOrder{1}:=1
End case

Case of
: (Column_Sort1=1)
    INSERT IN ARRAY($al_Setsort;1;1)
    INSERT IN ARRAY($al_SetOrder;1;1)
    $al_Setsort{1}:=1
    $al_SetOrder{1}:=1
: (Column_Sort2=1)
    INSERT IN ARRAY($al_Setsort;1;1)
    INSERT IN ARRAY($al_SetOrder;1;1)
    $al_Setsort{1}:=1
    $al_SetOrder{1}:=1
End case

QR SET SORTS(QRBuilder;$al_Setsort;$al_SetOrder)

```

4 When the solution matters Quick Report Wizard

Select the sorting order: Crosstab Report

Sorting order

You can now define sorting orders for rows and columns. Click on the following buttons to choose the

Rows:

☐ Do not sort

☒ Sort in ascending order

☐ Sort in descending order

Columns:

☐ Do not sort

☒ Sort in ascending order

☐ Sort in descending order

	[Sales]Sales_Date	Total
[Clients]Company_Name	[Sales]Sales_Amount	Σ Sum
Grand Total	Σ Sum	Σ Sum

7

Cancel
Back
Next

Step 8: Formatting your data

All of these methods obtain information about the Columns using the QR GET INFO COLUMN command and it sends the formatting information to the Quick Report via the QR SET INFO COLUMN command.

4D When the solution matters Quick Report Wizard

Formating your data: Crosstab Report

Define the format of your cells

You can now define default fomats from the contextual menu or enter your custom format below

Rows: applied for row titles

Columns: applied for column titles

Data source: applied for values displayed in cells, grand total and total

The Quick Report needs to format your data. If no format is chosen, it will use the default one in 4D.
A format is defined for the whole column. You can use available formats that 4D provides or use your own for text and numerical values.

[Sales]Sales_Date	Total
[Clients]Company_Name	[Sales]Sales_Amount
	Σ Sum
Grand Total	Σ Sum

8 Cancel Back Next

To enter custom formats for rows, the **vFormatRows** variable is used:

```
C_TEXT($title;$object;$vformat)
C_LONGINT($vHide;$vAutoSize;$vRepeatedValue)

QR GET INFO COLUMN(QRBuilder;2;$title;$object;$vHide;$vAutoSize;
$vRepeatedValue;$vformat)
QR SET INFO COLUMN(QRBuilder;2;$title;$object;$vHide;$vAutoSize;
$vRepeatedValue;Self->)
```

To enter custom formats for columns, the **vFormatColumnsvariable** is used:

```
C_TEXT($title;$object;$vformat)
C_LONGINT($vHide;$vAutoSize;$vRepeatedValue)

QR GET INFO COLUMN(QRBuilder;1;$title;$object;$vHide;$vAutoSize;
$vRepeatedValue;$vformat)
QR SET INFO COLUMN(QRBuilder;1;$title;$object;$vHide;$vAutoSize;
$vRepeatedValue;Self->)
```


To enter custom formats for the data source, the **vFormatDataSourcevariable** is used:

```
C_TEXT($title;$object;$vformat)
C_LONGINT($vHide;$vAutoSize;$vRepeatedValue)
QR_GET_INFO_COLUMN(QRBuilder;3;$title;$object;$vHide;$vAutoSize;
$vRepeatedValue;$vformat)
QR_SET_INFO_COLUMN(QRBuilder;3;$title;$object;$vHide;$vAutoSize;
$vRepeatedValue;Self->)
```

Step 9: Define the style attributes for your cells

The style attributes for the cells get defined in the Area itself. No methods are required at this stage.

When the solution matters Quick Report Wizard

Define the style attributes for your cells: Crosstab Report

Styles attributes can be defined for each cell.

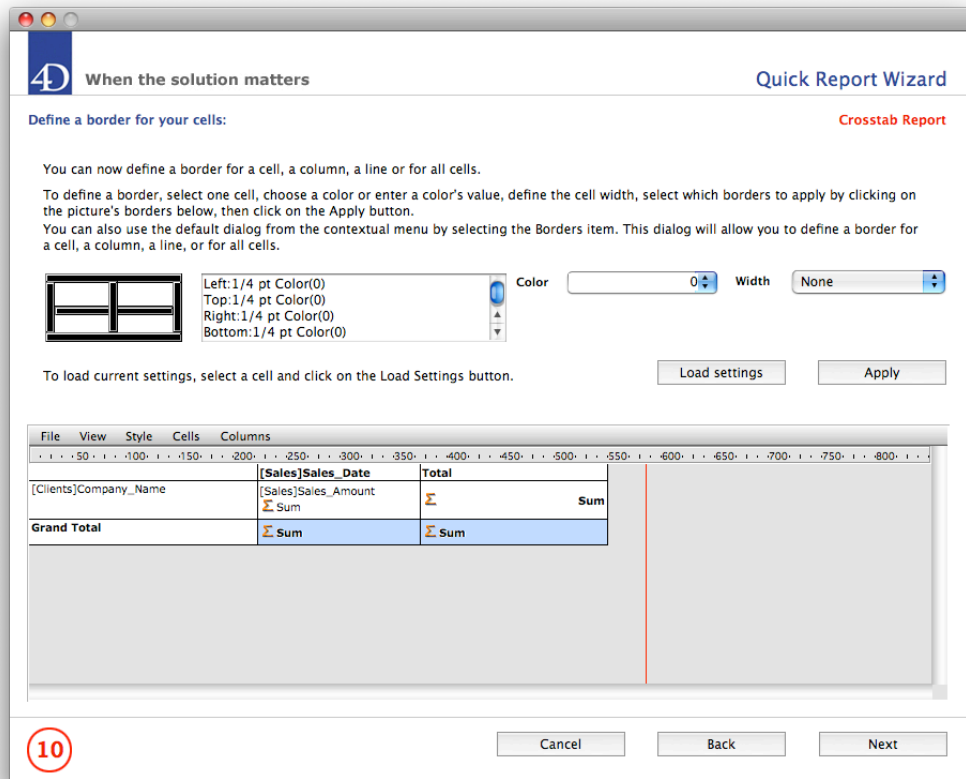
You can define your own styles such as font name, size, bold, italic and underline attributes, main color, justification, or background color.

The background color can be alternated. The first color will be used when printing even lines while the alternate background color will be used for odd lines.

[Clients]Company_Name	[Sales]Sales_Date	Total	Sum
	Σ Sum	Σ	Σ
Grand Total	Σ Sum	Σ Sum	Σ

9 Cancel Back Next

Step 10: Define a border for your cells



To define a border for the cells a color drop-down menu with the variable **at_Color3** is used:

```
C_LONGINT($value)
$value:=Self->
If ($value>0)
    vColorLine:=Num(at_ColorID{$value})
End if
```

A predefined line width drop-down menu with the variable **at_WidthLine** is also required.

In addition, two buttons. One for **Load settings**:

These methods are similar to the ones described in the List Report section.

```
C_LONGINT($left;$top;$right;$bottom)
C_LONGINT($Col;$Row)

QR GET SELECTION(qrBuilder;$left;$top;$right;$bottom)
ARRAY LONGINT($arr1;0)
ARRAY LONGINT($arr2;0)
QR GET SORTS(qrbuilder;$arr1;$arr2)
```

```

$Col:=$stop
$Row:=$left

C_TEXT(vTextBorder)
vTextBorder:=""
C_LONGINT($Color;$Size;$Color1;$Color2;$Color3;$Color4;$Color5;$Color6)
vRight:=""
vLeft:=""
vTop:=""
vBottom:=""
ARRAY TEXT($arr_val;5)
$arr_val{1}:="none"
$arr_val{2}:="1/4 pt"
$arr_val{3}:="1/2 pt"
$arr_val{4}:="1 pt"
$arr_val{5}:="2 pts"

If (($left#0) & ($stop#0))
    QR GET BORDERS(qrBuilder;$Col;$Row;1;$Size;$Color1) ` Left border
    vLeft:=$arr_val{$Size+1}+" Color("+String($Color1)+") "
    QR GET BORDERS(qrBuilder;$Col;$Row;2;$Size;$Color2) ` Top border
    vTop:=$arr_val{$Size+1}+" Color("+String($Color2)+") "
    QR GET BORDERS(qrBuilder;$Col;$Row;4;$Size;$Color3) ` Right border
    vRight:=$arr_val{$Size+1}+" Color("+String($Color3)+") "
    QR GET BORDERS(qrBuilder;$Col;$Row;8;$Size;$Color4) ` Bottom border
    vBottom:=$arr_val{$Size+1}+" Color("+String($Color4)+") "
    QR GET BORDERS(qrBuilder;$Col;$Row;32;$Size;$Color5) ` Center border
    vCenter:=$arr_val{$Size+1}+" Color("+String($Color5)+") "

    SET RGB COLORS(*;"RectLeftLine@";$color1;$color1)
    SET RGB COLORS(*;"RectTopLine@";$Color2;$Color2)
    SET RGB COLORS(*;"RectRightLine@";$Color3;$Color3)
    SET RGB COLORS(*;"RectBottomLine@";$Color4;$Color4)
    SET RGB COLORS(*;"RectCenterLine@";$Color5;$Color5)
    If (report2=1)
        QR GET BORDERS(qrBuilder;$Col;$Row;16;$Size;$Color6) ` Center border
        vVertCenter:=$arr_val{$Size+1}+" Color("+String($Color6)+") "
        SET RGB COLORS(*;"RectCenterVertLine@";$Color6;$Color6)

        vStatus:="Left:"+vLeft+Char(13)+"Top:"+vTop+Char(13)+"Right:"+vRight+Char(13)
    )+"Bottom:"+vBottom+Char(13)+"Hor. Center"+vLeft+Char(13)+vCenter+Char(13)+
    "Vert. center:"+vVertCenter
    End if
Else
    BEEP
End if

```

And one button for **Apply**:

```

C_LONGINT($left;$stop;$right;$bottom;$Color1;$Color2;$Color3;$Color4;$Color5;$Size)
C_LONGINT($col;$row;$vColorLine;$newsize)
ARRAY TEXT($arr_val;5)
$arr_val{1}:="none"
$arr_val{2}:="1/4 pt"
$arr_val{3}:="1/2 pt"
$arr_val{4}:="1 pt"
$arr_val{5}:="2 pts"
If (report1=1)
    ARRAY LONGINT($arr1;0)
    ARRAY LONGINT($arr2;0)
    QR GET SELECTION(qrBuilder;$left;$stop;$right;$bottom)

```

```

QR GET SORTS(qrbuilder,$arr1,$arr2)
Case of
    : ($top=1)
        $top:=-1
    : ($top=2)
        $top:=-2
    : ($top>(Size of array($arr1)+2))
        $top:=-3
    : ($top=0)
    Else
        $top:=$top-2
End case
$Col:=$left
$Row:=$top
Else
    ` Cross table
    QR GET SELECTION(qrBuilder;$left;$top;$right;$bottom)
    $Row:=$left
    $Col:=$Top
End if
$vColorLine:=vColorLine
$NewSize:=at_WidthLine-1
If (($left>0) & ($Top#0))
    ` A cell has been selected
    If (vLeftLine=1)
        SET RGB COLORS(*;"RectLeftLine@";$vColorLine;0)
        QR SET BORDERS(qrBuilder;$Col;$Row;1;$NewSize;$vColorLine) ` Left border
        vLeft:=$arr_val{$NewSize+1}+" Color("+String($vColorLine)+") "
    Else
        QR GET BORDERS(qrBuilder;$Col;$Row;1;$Size;$Color1) ` Left border
        vLeft:=$arr_val{$Size+1}+" Color("+String($Color1)+") "
    End if
    If (vTopLine=1)
        SET RGB COLORS(*;"RectTopLine@";$vColorLine;0)
        QR SET BORDERS(qrBuilder;$Col;$Row;2;$NewSize;$vColorLine) ` Top border
        vTop:=$arr_val{$NewSize+1}+" Color("+String($vColorLine)+") "
    Else
        QR GET BORDERS(qrBuilder;$Col;$Row;2;$Size;$Color2) ` Top border
        vTop:=$arr_val{$Size+1}+" Color("+String($Color2)+") "
    End if
    If (vRightLine=1)
        SET RGB COLORS(*;"RectRightLine@";$vColorLine;0)
        QR SET BORDERS(qrBuilder;$Col;$Row;4;$NewSize;$vColorLine) ` Right border
        vRight:=$arr_val{$NewSize+1}+" Color("+String($vColorLine)+") "
    Else
        QR GET BORDERS(qrBuilder;$Col;$Row;4;$Size;$Color3) ` Right border
        vRight:=$arr_val{$Size+1}+" Color("+String($Color3)+") "
    End if
    If (vBottomLine=1)
        SET RGB COLORS(*;"RectBottomLine@";$vColorLine;0)
        QR SET BORDERS(qrBuilder;$Col;$Row;8;$NewSize;$vColorLine) ` Bottom border
        vBottom:=$arr_val{$NewSize+1}+" Color("+String($vColorLine)+") "
    Else
        QR GET BORDERS(qrBuilder;$Col;$Row;8;$Size;$Color4) ` Bottom border
        vBottom:=$arr_val{$Size+1}+" Color("+String($Color4)+") "
    End if
    If (vCenterLine=1)
        SET RGB COLORS(*;"RectCenterLine@";$vColorLine;0)
        QR SET BORDERS(qrBuilder;$Col;$Row;32;$NewSize;$vColorLine)
        `Center Horz border
        vCenter:=$arr_val{$NewSize+1}+" Color("+String($vColorLine)+") "
    Else
        QR GET BORDERS(qrBuilder;$Col;$Row;32;$Size;$Color5) ` Center Horz border

```

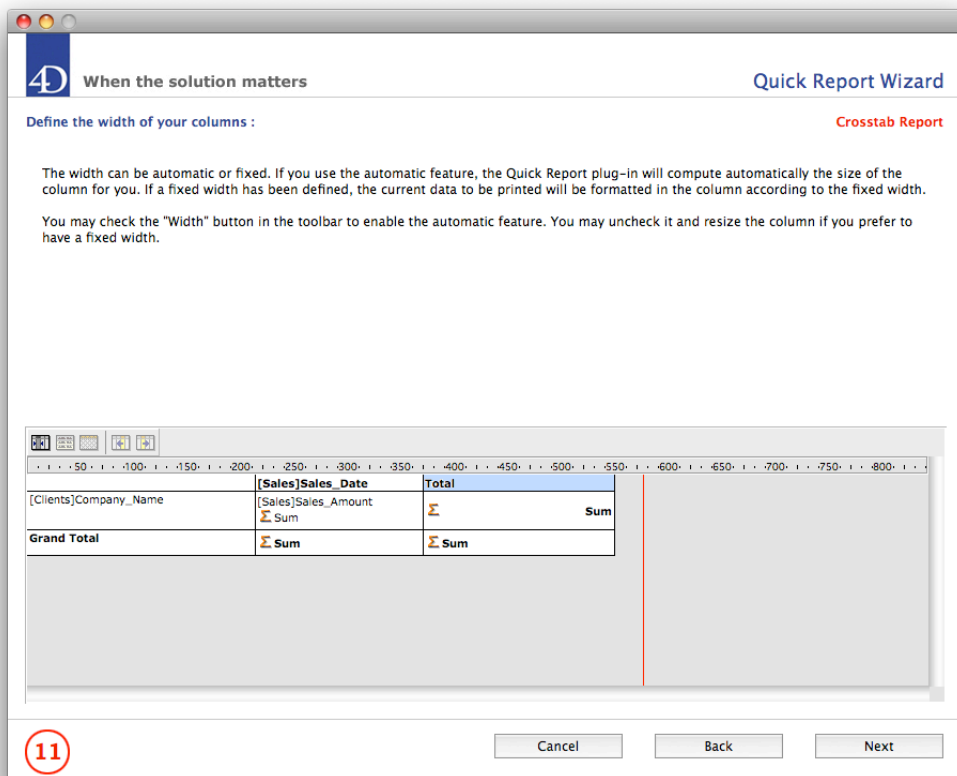
```

vCenter:=$arr_val{$Size+1}+" Color("+String($Color5)+") "
End if
If (vCenterVertLine=1)
    SET RGB COLORS(*;"RectCenterVertLine@";$vColorLine;0)
    QR SET BORDERS(qrBuilder;$Col;$Row;16;$NewSize;$vColorLine)
        Center vert border
    vVertCenter:=$arr_val{$NewSize+1}+" Color("+String($vColorLine)+") "
Else
    QR GET BORDERS(qrBuilder;$Col;$Row;16;$Size;$Color5) ` Center vert border
    vVertCenter:=$arr_val{$Size+1}+" Color("+String($Color5)+") "
End if
vStatus:="Left:"+vLeft+Char(13)+"Top:"+vTop+Char(13)+"Right:"+
vRight+Char(13)+"Bottom:"+vBottom+Char(13)+"Hor.Center"+vLeft+Char(13)+vCenter+
Char(13)+"Vert. center:"+vVertCenter
Else
    vStatus:=""
    BEEP
End if

```

Step 11: Define the width of your columns

Defining the widths of your columns can be accomplished in the Area itself. No methods are required at this stage.



Step 12 - 13: Define the page header and footer for your report

To define the page header, we use the **Set Header** button:

This method takes the information from the Header section and applies it to the Quick Report.

```
vPictAlignment:=Num(Alignment1=1)+(Num(Alignment2=1)*2)+(Num(Alignment3=1)*3)
QR SET HEADER AND FOOTER(QRBuilder;1;vTextLeft;vTextCenter;vTextRight;vHeight;
VPictHeader;vPictAlignment)
vFont:=Font number(at_FontName{at_FontName})
QR SET TEXT PROPERTY(QRBuilder;0;-4;1;vFont)
QR SET TEXT PROPERTY(QRBuilder;0;-4;2;vSize)
QR SET TEXT PROPERTY(QRBuilder;0;-4;3;vStyleBold)
QR SET TEXT PROPERTY(QRBuilder;0;-4;4;vStyleItalic)
QR SET TEXT PROPERTY(QRBuilder;0;-4;5;vStyleUnderline)
```

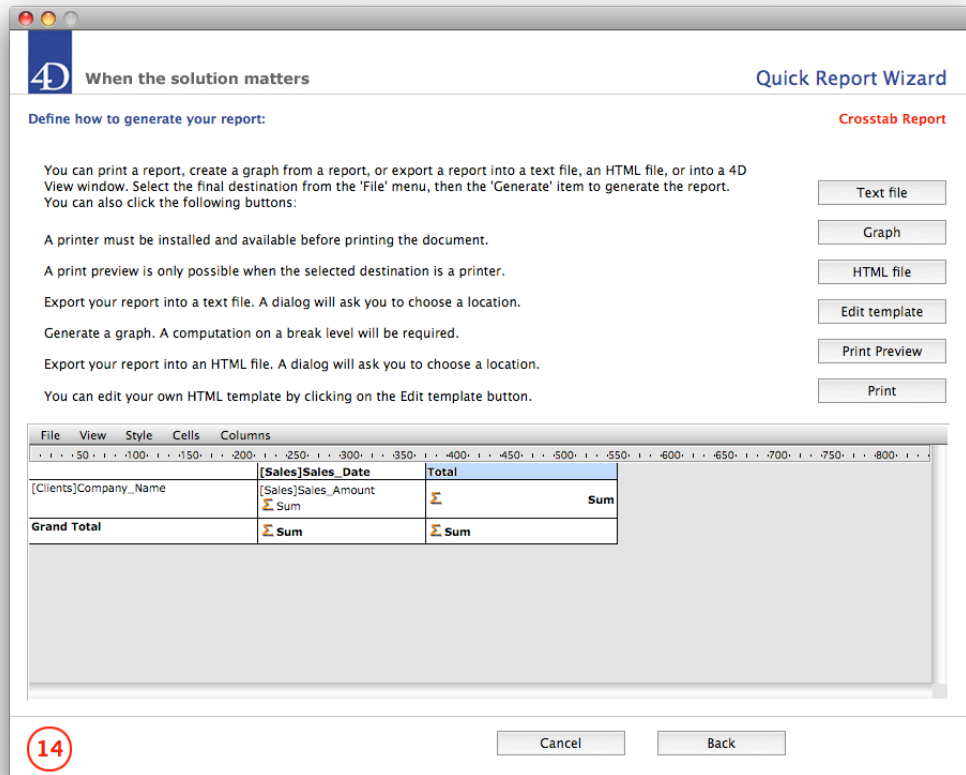
To define the page footer, we use the **Set Footer** button:

This method takes the information from the Footer section and applies it to the Quick Report.

```
vPictAlignment:=Num(Alignment1=1)+(Num(Alignment2=1)*2)+(Num(Alignment3=1)*3)
QR SET HEADER AND FOOTER(QRBuilder;2;vTextLeft;vTextCenter;vTextRight;vHeight;
VPictFooter;vPictAlignment)
vFont:=Font number(at_FontName{at_FontName})
QR SET TEXT PROPERTY(QRBuilder;0;-4;1;vFont)
```

```
QR SET TEXT PROPERTY (QRBuilder;0;-4;2;vSize)
QR SET TEXT PROPERTY (QRBuilder;0;-4;3;vStyleBold)
QR SET TEXT PROPERTY (QRBuilder;0;-4;4;vStyleItalic)
QR SET TEXT PROPERTY (QRBuilder;0;-4;5;vStyleUnderline)
```

Step 14: Generate your Report



The following options are available:

Text file:

```
QR SET DESTINATION (QRBuilder;2)
QR RUN (QRBuilder)
```

Graph:

```
QR SET DESTINATION (QRBuilder;4)
QR RUN (QRBuilder)
```

HTML file:

```
QR SET DESTINATION (QRBuilder;5)
QR RUN (QRBuilder)
```

Edit template:

```
QR SET DESTINATION(QRBuilder;5)

vHTMLTemplate:=QR Get HTML template(QRBuilder)
vPage:=vPage+1
Wizard_Changepage
```

Print Preview:

```
QR SET DESTINATION(QRBuilder;1)
QR EXECUTE COMMAND(QRBuilder;qr cmd print preview)
```

Print:

```
QR SET DESTINATION(QRBuilder;1)
QR RUN(QRBuilder)
```

Navigation

The navigation options in the List Reports and Cross-Tab Reports depend on 3 buttons:

A **Cancel** button with a Cancel standard action assigned to it.

A **Next** button in the first page:

This method initializes the Quick Report Area with default values.

```
QR EXECUTE COMMAND(QRBuilder;2000)

Case of
: (report1=1)
    vPage:=2
    vTitle:="List Report"
    QR SET REPORT KIND(QRBuilder;1)
    vStepTitle:="Step Two:"

    ` Reinit if Back button
    ARRAY TEXT(at_Sort_ColText;0)
    ARRAY LONGINT(al_Sort_ColID;0)
    ARRAY PICTURE(ap_Sort_ColPict;0)
    ARRAY LONGINT(al_Sort_ColOrder;0)

: (report2=1)
    vTitle:="Crosstab Report"
    QR SET REPORT KIND(QRBuilder;2)
    vCTColumn:=""
    vCTLine:=""
    vCTDataSource:=""
    vGrandTotalTitle:="Grand Total"
    vTotalTitle:="Total"
    QR SET TOTALS DATA(QRBuilder;1;3;vGrandTotalTitle)
    QR SET TOTALS DATA(QRBuilder;3;1;vTotalTitle)
    vStepTitle:="Step Three:"
    tc_CT_DataDef:=1
```



```

tc_CT_Computation:=1
QR SET AREA PROPERTY (QRBuilder;6;0)
QR SET AREA PROPERTY (QRBuilder;4;0)

cbCount1:=0
cbMin1:=0
cbMax1:=0
cbAverage1:=0
cbSum1:=1
cbCount2:=0
cbMin2:=0
cbMax2:=0
cbAverage2:=0
cbSum2:=1
cbCount3:=0
cbMin3:=0
cbMax3:=0
cbAverage3:=0
cbSum3:=1

Row_Sort1:=1
Row_Sort2:=0
Row_Sort3:=0
Column_Sort1:=1
Column_Sort2:=0
Column_Sort3:=0

Wizard_UpdateComputation (1)
Wizard_UpdateComputation (2)
Wizard_UpdateComputation (3)
vPage:=16
End case

Wizard_Changepage

at_showfields:=1
Wizard_Load_HLTables

```

This method updates the Count, Total and Grand Total computation values.

```

\ .....
\ Project Method: Wizard_UpdateComputation
\ Description: Updates Totals
\ Called by: Wiz Page
\ Calls: Nothing
\ Parameters: None
\ Returns: Nothing
\ Created by Jean-Yves Fock-Hoon
\ Modified by Luis Piñeiros
\ 2009
\ .....

C_LONGINT($Page;$Col;$Line)
C_REAL($Value)

$Page:=$1
Case of
: ($Page=1)
    $Value:=(cbCount1*16)+(cbMin3*4)+(cbMax3*8)+(cbAverage1*2)+cbSum1
    $Col:=2
    $Line:=2
: ($Page=2) `Total

```

```

        $Value:=(cbCount2*16)+(cbMin3*4)+(cbMax3*8)+(cbAverage2*2)+cbSum2
        $Col:=3
        $Line:=2

: ($Page=3)  ` Grand Total
        $Value:=(cbCount3*16)+(cbMin3*4)+(cbMax3*8)+(cbAverage3*2)+cbSum3
        $Col:=2
        $Line:=3

End case

QR SET TOTALS DATA(QRBuilder;$Col;$Line;$Value)

```

This method sets default values for the Quick Report Area for Menu Bar, File Tool Bar, Font, Computation, Background, Move/Hide and Contextual Menu.

```

` .....
` Project Method: Wizard_SetFloatingwindows
` Description: Sets Area Properties
` Called by: WizardChangePage
` Calls: Nothing
` Parameters:
` $1 - Longint - Menu bar
` $2 - Longint - File toolbar
` $3 - Longint - Font
` $4 - Longint - Computation
` $5 - Longint - Background
` $6 - Longint - Move/Hide
` $7 - Longint - Contextual menu
` Returns: Nothing
` Created by Jean-Yves Fock-Hoon
` Modified by Luis Piñeiros
` 2009
` .....
QR SET AREA PROPERTY(QRBuilder;1;$1)
QR SET AREA PROPERTY(QRBuilder;2;$2)
QR SET AREA PROPERTY(QRBuilder;3;$3)
QR SET AREA PROPERTY(QRBuilder;4;$4)
QR SET AREA PROPERTY(QRBuilder;5;$5)
QR SET AREA PROPERTY(QRBuilder;6;$6)
QR SET AREA PROPERTY(QRBuilder;7;$7)

```

This method loads the Tables and fields into a Hierarchical Array.

```

` .....
` Project Method: Wizard_Load_HLTables
` Description: Loads HL Tables
` Called by: Wiz Form
` Calls: Wizard_Load_related_fields
` Parameters: None
` Returns: Nothing
` Created by Jean-Yves Fock-Hoon
` Modified by Luis Piñeiros
` 2009
` .....

C_LONGINT($Ref;$ItemPos)

Case of
: (at_showfields=1)

```

```

        \ Table only
$Ref:=Find in array(Øat_CurrentTable;at_CurrentTable{at_CurrentTable})
If ($Ref#-1)
    $Ref:=Øat_CurrentTableID{$Ref}

    SELECT LIST ITEMS BY REFERENCE(ØHL_StructureDef;$Ref)
    $ItemPos:=Selected list items(ØHL_StructureDef)
    GET LIST ITEM(ØHL_StructureDef;$ItemPos;$ItemRef;$ItemText;HLTables)
    SAVE LIST(hltables;"Tables")

End if

: (at_showfields=2)
    \ Table and related fields
    Wizard_Load_related_fields

: (at_showfields=3)
    \ All Tables
    HLTables:=ØHL_StructureDef

End case
REDRAW LIST(HLTables)

```

This method loads the related fields into a Hierarchical Array.

```

.....
\ Project Method: Wizard_Load_related_fields
\ Description: Loads HL Related Fields
\ Called by: Wizard_Load_HLTables
\ Calls: Nothing
\ Parameters: None
\ Returns: Nothing
\ Created by Jean-Yves Fock-Hoon
\ Modified by Luis Piñeiros
\ 2009
.....

C_LONGINT($ItemPos;$Ref;$nbElem;$ind;$i;$CurrentList;$Sublist2)
C_BOOLEAN($Aig)

SELECT LIST ITEMS BY REFERENCE(ØHL_StructureDef;at_CurrentTable)
$ItemPos:=Selected list items(ØHL_StructureDef)
GET LIST ITEM(ØHL_StructureDef;$ItemPos;$ItemRef;$ItemText;sublist)

\ We have selected the first table in HLTables
\ $ItemRef is our table number and is the Many table
\ let's check into our arrays if we do have relations
HLTables:=Copy list(sublist)

ARRAY LONGINT(al_Tables2Display;1)
$Ref:=Find in array(Øat_CurrentTable;at_CurrentTable{at_CurrentTable})
If ($Ref#-1)
    $Ref:=Øat_CurrentTableID{$Ref}
    al_Tables2Display{1}:=$Ref
End if

$nbElem:=1
\ Find all One Tables
$Aig:=False
Repeat
    $Aig:=True

```

```

For ($i;1;Size of array(al_Tables2Display))
    $ind:=Find in array(Øal_Rel_ManyTable;al_Tables2Display{$i})
    While ($ind>0)
        If (Find in array(al_Tables2Display;Øal_Rel_OneTable{$ind})=-1)
            $Aig:=False
            $nbElem:=$nbElem+1
            INSERT IN ARRAY(al_Tables2Display;$nbElem;1)

            al_Tables2Display{$nbElem}:=Øal_Rel_OneTable{$ind}

        End if
        $ind:=Find in array(Øal_Rel_ManyTable;al_Tables2Display{$i};$ind+1)
    End while

End for

Until ($Aig)

ARRAY LONGINT(al_Tables2Display2;1)
$Ref:=Find in array(Øat_CurrentTable;at_CurrentTable{at_CurrentTable})
If ($Ref#-1)
    $Ref:=Øat_CurrentTableID{$Ref}
    al_Tables2Display2{1}:=$Ref
End if

$nbElem:=1
` Find all Many Tables
$Aig:=False
Repeat
    $Aig:=True
    For ($i;1;Size of array(al_Tables2Display2))
        $ind:=Find in array(Øal_Rel_OneTable;al_Tables2Display2{$i})
        If ($ind>0)
            If (Find in array(al_Tables2Display2;Øal_Rel_ManyTable{$ind})=-1)
                $Aig:=False
                $nbElem:=$nbElem+1
                INSERT IN ARRAY(al_Tables2Display2;$nbElem;1)

                al_Tables2Display2{$nbElem}:=Øal_Rel_ManyTable{$ind}
            End if
        End if
    End for

Until ($Aig)

DELETE FROM ARRAY(al_Tables2Display;1)

DELETE FROM ARRAY(al_Tables2Display2;1)

If (Size of array(al_Tables2Display)>0)
    $CurrentList:=New list

    For ($i;1;Size of array(al_Tables2Display))
        SELECT LIST ITEMS BY REFERENCE(ØHL_StructureDef;al_Tables2Display{$i})
        $ItemPos:=Selected list items(ØHL_StructureDef)
        GET LIST ITEM(ØHL_StructureDef;$ItemPos;$ItemRef;$ItemText;$sublist)

        ` We have selected the first table in HLTables
        ` $ItemRef is our table number and is the Many table
        ` let's check into our arrays if we do have relations

```

```

    $sublist2:=Copy list($sublist)

    APPEND TO LIST($CurrentList;$ItemText;$ItemRef;$Sublist2;False)
End for
$ItemRef:=500
APPEND TO LIST(HLTables;"One tables";$ItemRef;$CurrentList;False)
End if

If (Size of array(al_Tables2Display2)>0)
    $CurrentList:=New list

    For ($i;1;Size of array(al_Tables2Display2))
        SELECT LIST ITEMS BY REFERENCE(ØHL_StructureDef;al_Tables2Display2{$i})
        $ItemPos:=Selected list items(ØHL_StructureDef)
        GET LIST ITEM(ØHL_StructureDef;$ItemPos;$ItemRef;$ItemText;$sublist)

        ` We have selected the first table in HLTables
        ` $ItemRef is our table number and is the Many table
        ` let's check into our arrays if we do have relations
        $sublist2:=Copy list($sublist)

        APPEND TO LIST($CurrentList;$ItemText;$ItemRef;$Sublist2;False)
    End for
    $ItemRef:=501
    APPEND TO LIST(HLTables;"Many tables";$ItemRef;$CurrentList;False)
End if

```

A **Next** button in the rest of the pages:

```

vPage:=vPage+1
Wizard_Changepage

```

A **Back** button:

```

vPage:=vPage-1
Wizard_Changepage

```

This method uses the Wizard_SetFloattingwindows method to set the values of the Quick Report Area in each page of the Wiz Form.

```

` .....
` Project Method: WizardChangePage
` Description: Redirect User Commands
` Called by: Wiz Form
` Calls: Wizard_SetFloattingwindows
` Parameters: None
` Returns: Nothing
` Created by Jean-Yves Fock-Hoon
` Modified by Luis Piñeiros
` 2009
` .....

` Parameters:
` $1 - Longint - Menu bar
` $2 - Longint - File toolbar
` $3 - Longint - Font
` $4 - Longint - Computation
` $5 - Longint - Background
` $6 - Longint - Move/Hide

```

```

` $7 - Longint - Contextual menu

Case of
: (vPage=1) `Main dialog
    Wizard_SetFloattingwindows (0;0;0;0;0;0;0)
: (vPage=2) `Select fields
    Wizard_SetFloattingwindows (0;0;0;0;0;1;1)
: (vPage=3) `Define the values inside your break levels
    Wizard_SetFloattingwindows (0;0;0;1;0;1;1)
: (vPage=4) `Define an action in your break levels
    Wizard_SetFloattingwindows (0;0;0;0;0;0;1)
: (vPage=5) `Define a format for your cells
    Wizard_SetFloattingwindows (0;0;0;0;0;0;1)
: (vPage=6) `Define a Style attributes for your cells
    Wizard_SetFloattingwindows (0;0;1;0;1;0;1)
: (vPage=7) `Define a border for your cells
    Wizard_SetFloattingwindows (1;0;0;0;0;0;1)
: (vPage=8) `Define a width for your columns
    Wizard_SetFloattingwindows (0;0;0;0;0;1;0)
: (vPage=9) `Define a header
    Wizard_SetFloattingwindows (1;0;0;0;0;0;0)
: (vPage=10) `Define a footer
    Wizard_SetFloattingwindows (1;0;0;0;0;0;0)
: (vPage=11) `Destination/printing
    Wizard_SetFloattingwindows (1;1;0;0;0;0;0)

    ` Crosstab
: (vPage=16) ` Select a field for columns
    Wizard_SetFloattingwindows (0;0;0;0;0;0;0)
: (vPage=17) ` Select a field for lines
    Wizard_SetFloattingwindows (0;0;0;0;0;0;0)
: (vPage=18) ` Select a field for data source
    Wizard_SetFloattingwindows (0;0;0;0;0;0;0)
: (vPage=19) ` Computation for cell data
    Wizard_SetFloattingwindows (1;0;0;1;0;0;1)
: (vPage=20) ` Computation for Total
    Wizard_SetFloattingwindows (1;0;0;1;0;0;1)
: (vPage=21) `Computation for Grand Total
    Wizard_SetFloattingwindows (1;0;0;1;0;0;1)
: (vPage=22) `Soring order
    Wizard_SetFloattingwindows (0;0;0;0;0;0;0)
: (vPage=23) ` Formting data
    Wizard_SetFloattingwindows (0;0;0;0;0;0;0)
: (vPage=24) `Define a Style attributes for your cells
    Wizard_SetFloattingwindows (0;0;1;0;1;0;1)
: (vPage=25) `Define a border
    Wizard_SetFloattingwindows (1;0;0;0;0;0;1)
: (vPage=25) `Define a border
    Wizard_SetFloattingwindows (1;0;0;0;0;0;1)
: (vPage=26) `Define width
    Wizard_SetFloattingwindows (0;0;0;0;0;1;0)
: (vPage=27) `Define a header
    Wizard_SetFloattingwindows (1;0;0;0;0;0;0)
: (vPage=28) `Define a footer
    Wizard_SetFloattingwindows (1;0;0;0;0;0;0)
: (vPage=29) `Define a border
    Wizard_SetFloattingwindows (1;0;0;0;0;0;1)

End case
vCurrentPage:=Current form page
GOTO PAGE(vPage)

```

Conclusion

This Technical Note showed how to create a Quick Report Wizard using the built-in 4D v11 SQL Quick Report commands. Quick Reports can be used to provide your users with a powerful reporting tool that can complement the custom reports in your application. By replacing the Quick Report capability in 4D v11SQL Release 4 with a custom Quick Report Wizard, you can accomplish a higher level of control as well as minimize the complexity of presenting information in a number of different formats and styles.