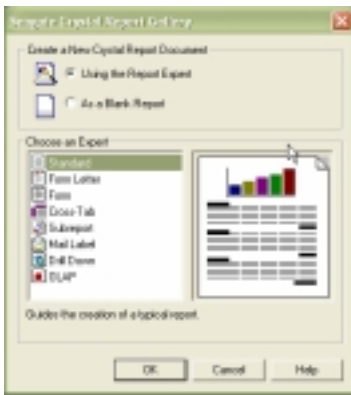


Passing Parameters from ACCPAC for Windows to Crystal Reports

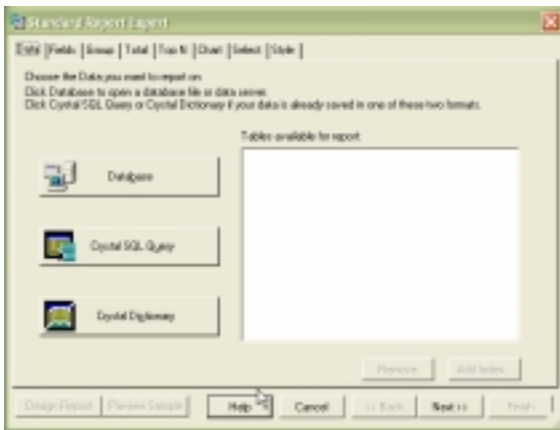
To successfully print reports from ACCPAC Advantage Series using Crystal Reports a number of issues should be considered. If you are going to create a report that will only attach to a single dataset then the report may be created directly within Crystal Reports and attached into ACCPAC. On the other hand, many reports may be generated in which you will not know the final data set location. For these you may want to create a report that needs few modifications to an .ini file and a Visual Basic for Applications (VBA) macro to accomplish.

Reports locked to a single data source

When creating the report you can just create a new report and select the dataset that you will use:



Open Crystal and create a new report

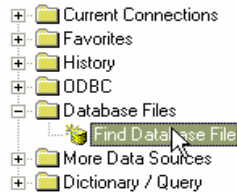


Select Database

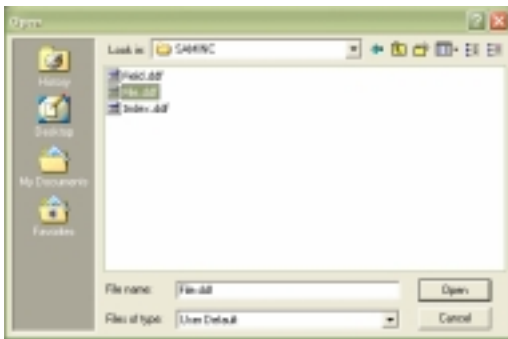
Passing Parameters from ACCPAC for Windows to Crystal Reports



Choose **Database files**.



Click on **Find Database File**.



Search for the **Drive, Path and Folder** that would contain your data. For a Pervasive.SQL database you will find **Field.DDF**, **File.DDF** and **Index.DDF** files. Select either the **Field.DDF** or **File.DDF** file.

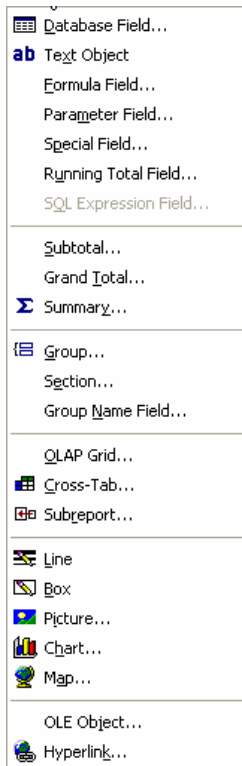


A list of all the data files will appear. Select the ones that you require for your report and then click the add button. This will allow you to access the data in these files. The report **will be locked to that location** unless you physically come back into the report and change this location. If you develop a report locally on C: then distribute the report to someone with their dataset on a network or a different drive/path then you will need to change the location in the report.

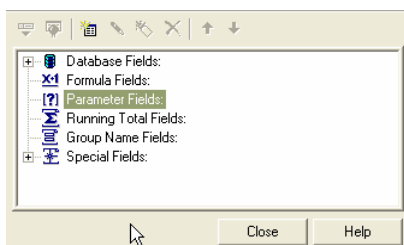
Reports without User Input Parameters and unknown data source location

Many times you will create a report that may be re-used for multiple clients, the client may wish to use the report on multiple datasets or you won't know the final directory location of the data. To resolve this you will need to add at least three parameters to your report **and** affect the .ini file for the module.

First let's look at the Crystal parameters:

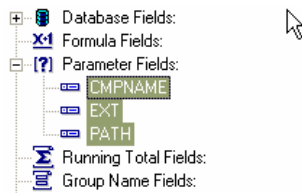
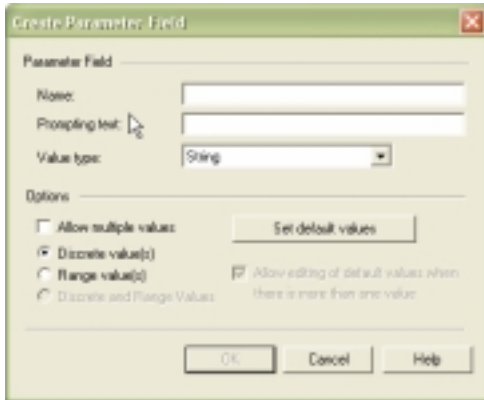


In Crystal Reports, once you have created your report you must create **Parameters** that will be passed from ACCPAC to Crystal to tell it what company name to use, what path the data has and the extension (.dat in Pervasive) the data uses. Click on **Insert** at the top of the Crystal toolbars and select **Parameter Field...**



Highlight the **Parameters Fields** and click on the **New** icon at the top.

Passing Parameters from ACCPAC for Windows to Crystal Reports



All parameters coming from ACCPAC need to be in **String** format.

Create three parameters called: **CMPNAME**, **EXT** and **PATH**. These parameters will automatically be passed from ACCPAC to Crystal so they do not need to be referenced anywhere else in Crystal. If you wish for the company name to appear in the report then you can use the parameter within the report. It will automatically know the name from whatever ACCPAC dataset one runs the report from.

Now let's look to see how we will change the **.ini** file. In each of the module directories, there is an **.ini** file that will be in the form of **XXrpt.ini** (where **XX** is the two letter designation of the module such as **GL**, **AR**, **AP** and such). The **.ini** file will be found in **d:\program files\accpac\oe42a** for Order Entry and the appropriate directory for any other module.

***** Object id = OE1200 *****

[OECONF01]

heading=28020

'caret=OECONF01

crystal=OECONF01

orientation=portrait

paper size=1

Report Name - must start with the modules' two letter abbreviation

The heading of the report window

Designates a Caret Report. The ' in front of the line comments out the line

Designates a Crystal Report

Report orientation (Portrait or Landscape)

Paper size of 1 is 8.5 x 11

Passing Parameters from ACCPAC for Windows to Crystal Reports

optparams=2

Specifies the number of parameters that are used in the report. In this case it will be 2. The parameter **CMPNAME** is always **assumed and doesn't need to be added!**

2=PATH COMPANY

Enter this parameter as shown. Use the space bar (not tab button) to space out - there is no defined number of spaces so make at least 5

3=EXT FILEEXT

Enter this parameter as shown. Use the space bar (not tab button) to space out - there is no defined number of spaces so make at least 5

Here is an example of a report for Order Entry that we will call OENUCOMM.RPT

Add an entry to the bottom of the *d:\program files\accpac\oe42a\oerpt.ini* file that would look like:

```
***** Object id = XXXX Order Entry Applet *****  
[OENUCOMM]  
heading="No Parameter report"  
crystal=OENUCOMM  
orientation=portrait  
paper size=1  
optparams=2  
2=PATH           COMPANY  
3=EXT            FILEEXT
```

And finally create a macro to print the report that would look like: (you could copy the text below into a macro and not even have to create one. Just change the OENUCOMM[OENUCOMM.RPT} to reflect you Crystal Report file name)

With the combination of the above three parameters in my report, the code in the OERPT.INI file and a very general macro (following page) you would be able to use this report in conjunction with any ACCPAC dataset that you would currently be in.

Passing Parameters from ACCPAC for Windows to Crystal Reports

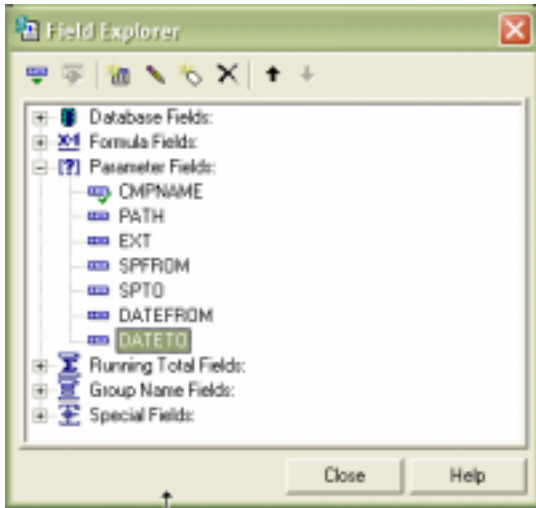
```
Sub MainSub()
'
' ACCPAC Macro file: E:\program files\ACCPAC\MACROS\OENUCOMMavb.avb
' Recorded at: Thu Dec 27 12:54:25 2001
'
On Error GoTo ACCPACErrorHandler

'Initialize report and set parameters and print
  Dim rptObj As ACCPACXAPILib.xapiReport
  Set rptObj = CreateObject("ACCPAC.xapiReport")
  rptObj.Select Session, "OENUCOMM[OENUCOMM.RPT]", ""
  rptObj.PrintDestination = PD_PREVIEW
  rptObj.PrintReport 1

Exit Sub
ACCPACErrorHandler:
  Dim Error As Variant
  If Errors.Count = 0 Then
    MsgBox Err.Description
  Else
    For Each Error In Errors
      MsgBox Error.Description
    Next
    Errors.Clear
  End If
  Resume Next
End Sub
```

Reports using User Input Parameters and from a VBA macro

Create your report as you normally would. Make sure that you have the **?CMPNAME**, **?PATH** and **?EXT** parameters as the first three parameters outlined above. In addition, any other parameters you pass from a VBA Macro to the report **must** be in string format. To show an example, we will modify the OENUCOMM.RPT to ask for a Salesperson range and Invoice date range. First add four additional parameters to the report:



?SPFROM - String format
?SPTO - String format
?DATEFROM - String (not date) format
?DATETO - String (not date) format

The selection criteria in the Crystal report must be changed to reflect the new parameters. **Note:** even if you are going to use parameters that will end up being numeric, date or Boolean you must create them as string and convert them in some way in the report.

//Selection Criteria:

```
{OEAUDH.INVDATE} in (tonumber(mid({?DATEFROM},7,4) &
mid({?DATEFROM},1,2) & mid({?DATEFROM},4,2))) to
(tonumber(mid({?DATETO},7,4) & mid({?DATETO},1,2) & mid({?DATETO},4,2))))
and {OEAUDH.SALESPER1} in {?SPFROM} to {?SPTO}
```

In the above selection criteria copied from the report, the **?DATEFROM** and **?DATETO** will be put in formatted as MM/DD/YYYY while the {OEAUDH.INVDATE} is a number kept in YYYYMMDD format.

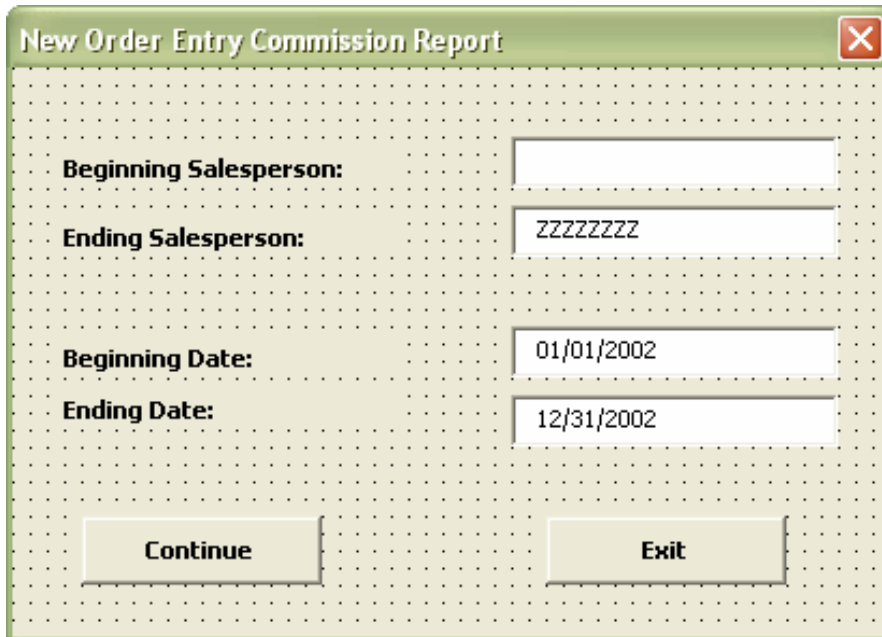
Passing Parameters from ACCPAC for Windows to Crystal Reports

Now, we must add the new parameters to the **OERPT.INI** file.

```
***** Object id = XXXX Order Entry Applet *****  
[OENUCOMM]  
heading="No Parameter report"  
crystal=OENUCOMM  
orientation=portrait  
paper size=1  
optparams=6  
2=PATH          COMPANY  
3=EXT           FILEEXT  
4=SPFROM       STRING  
5=SPTO         STRING  
6=DATEFROM     STRING  
7=DATETO       STRING
```

Finally, we will modify the OENUCOMM.AVB macro to ask for the proper parameters.

Create a new form:



The screenshot shows a Windows application window titled "New Order Entry Commission Report". The window has a standard title bar with a close button. The main area of the window is a form with a dotted grid background. The form contains four labeled input fields arranged vertically. The first field is labeled "Beginning Salesperson:" and is empty. The second field is labeled "Ending Salesperson:" and contains the text "ZZZZZZZZ". The third field is labeled "Beginning Date:" and contains the date "01/01/2002". The fourth field is labeled "Ending Date:" and contains the date "12/31/2002". At the bottom of the form, there are two buttons: "Continue" on the left and "Exit" on the right.

With the following code behind the form:

Passing Parameters from ACCPAC for Windows to Crystal Reports

```
Private Sub cmdContinue_Click()
    'If the Continue button is clicked
    'then save the input into the parameters
    SPFROM = Trim(frmlInput.txtSPFROM)
    SPTO = Trim(frmlInput.txtSPTO)
    DATEFROM = Trim(frmlInput.txtDateFrom)
    DATETO = Trim(frmlInput.txtDateTo)
    'Then unload the form
    Unload frmlInput
    'And go back to the main subroutine
End Sub

Private Sub cmdExit_Click()
    'If the Exit button is clicked - unload the macro
    Unload frmlInput
End Sub
```

and the following modifications to our original macro:

```
'Instantiate the parameter entries as string
Public SPFROM As String
Public SPTO As String
Public DATEFROM As String
Public DATETO As String
Sub MainSub()
    'Load and show the Input Form
    SPFROM = " "
    frmlInput.Show
    'When return from that, print the report
    PrintReport
End Sub

Sub PrintReport()
    '
    ' ACCPAC Macro file: E:\program files\ACCPAC\MACROS\OENUCOMMavb.avb
    ' Recorded at: Thu Dec 27 12:54:25 2001
    '
    On Error GoTo ACCPACErrorHandler

    'Initialize report and set parameters and print
    Dim rptObj As ACCPACXAPILib.xapiReport
    Set rptObj = CreateObject("ACCPAC.xapiReport")
    rptObj.Select Session, "OENUCOMM[OENUCOMM.RPT]", ""
    'Put the proper values into our report parameters
    rptObj.SetParam "SPFROM", SPFROM ' Report parameter: 4
    rptObj.SetParam "SPTO", SPTO ' Report parameter: 5
    rptObj.SetParam "DATEFROM", DATEFROM ' Report parameter: 6
    rptObj.SetParam "DATETO", DATETO ' Report parameter: 7

    rptObj.PrintDestination = PD_PREVIEW
    rptObj.PrintReport 1
Exit Sub
'NOTE: Error handler routine removed for brevity
```