

# Crystal Reports Automation Server

## Data Access Objects (DAO) Sample Application

---

### Overview

This module outlines how to report off Data Access Objects (DAO) using the Seagate Crystal Report Automation Server. This section covers how to return the DAO recordset at runtime, display the data you are going to report off in a list box, create the report at runtime and preview the report.

### Contents

<b>MICROSOFT'S DATA ACCESS OBJECT.....</b>	<b>1</b>
<i>Completing the Application: .....</i>	<i>5</i>
<b>REFERENCES .....</b>	<b>6</b>
<b>CONTACTING CRYSTAL DECISIONS FOR TECHNICAL SUPPORT .....</b>	<b>6</b>

### Microsoft's Data Access Object

DAO is the COM interface to the Jet database engine. It is primarily used to access ISAM (Indexed Sequential Access Method) databases and supports most common ISAM database techniques. This would be used for the Native Jet (MDB), or ISAM databases such as Btrieve, dBase, FoxPro, or Paradox. DAO 3.5 can be used as an interface with RDO to access ODBC data sources; though it is limited in its abilities and functionality with ODBC. The first two things to do are reference the Microsoft DAO 3.51 Object Library and the Crystal Report Engine 6.0 Object Library. If it is not listed, go under Project, References, and browse and find the DAO350.dll. Add a module to the Project, in the General declarations add this code:

```
Declare Function vbEncodeIPtr Lib "p2smon.dll" (X As Object) As String
```

```
Declare Function CreateReportOnRuntimeDS Lib "p2smon.dll"  
(X As Object, ByVal reportPath$, ByVal fieldDefFilePath$,  
ByVal
```

```
bOverWriteExistingFiles%, ByVal bLaunchDesigner%) As  
Integer
```

```
Declare Function CreateFieldDefFile Lib "p2smon.dll" (X As
```

```
Object, ByVal fieldDefFilePath$, ByVal  
bOverWriteExistingFiles%) As Integer
```

In the General Declarations of the form add this code:

```
Dim CrApp As CRPEAuto.Application  
Dim CrRep As CRPEAuto.Report  
Dim DAORs As DAO.Recordset  
Dim DAODB As DAO.Database  
Dim wrkODBC As Workspace  
Dim wrkJet As Workspace
```

Step 1: Setting the database and creating the recordset.

```
Private Sub Form_Load()  
  
'this section binds the application object to the CrApp  
'variable. Also the DAORs object is created and bound.  
There are 2 methods shown. One uses ODBC  
  
the other Jet (Native). You can try both, just comment out  
the one you are not using.  
  
Set CrApp = CreateObject("crystal.crpe.application")  
  
  
'*****NOTE***** The following can be used to open the  
Database Craze.mdb via ODBC  
  
'Create Microsoft ODBC Workspace. If you have Seagate  
Crystal Reports 7, use Xtreme.mdb or  
  
"Xtreme Sample Data"  
  
Set wrkODBC = CreateWorkspace("", "admin", "", dbUseODBC)  
  
'set the location of your craze.mdb here  
  
Set DAODB = wrkODBC.OpenDatabase("Xtreme sample data")  
  
'code for creating the recordset. Uses the ODBC datasource  
name  
  
'Craze Sample Data. Make sure that this DSN is properly  
set up.  
  
Set DAORs = DAODB.OpenRecordset("Customer", dbOpenDynamic)  
  
  
'*****NOTE***** The following can be used to open the  
Database Craze.mdb Natively  
  
'Create Microsoft Jet Workspace  
  
Set wrkJet = CreateWorkspace("", "admin", "", dbUseJet)  
  
'set the location of the database here  
  
Set DAODB = wrkJet.OpenDatabase("C:\crw6032\Xtreme.mdb")  
  
'code for creating recordset off Customer table.
```

```
Set DAORs = DAOdb.OpenRecordset("Customer", dbOpenTable)
```

```
End Sub
```

Step 2: Put five Command buttons, a list box and a label on the form.

Rename Command button 2 to "Populate Recordset"

```
Private Sub Command2_Click()  
'this fills a list box with the third column, which for  
'Craze Sample Data happens to be Customer Name from  
Craze.mdb
```

```
DAORs.MoveFirst  
Label1.Caption = DAORs(2).Name  
Do Until DAORs.EOF  
    List1.AddItem DAORs(2)  
    DAORs.MoveNext  
Loop  
MsgBox "RecordSet Populated"  
Command3.Enabled = True  
Command1.Enabled = True  
Command4.Enabled = True
```

```
End Sub
```

Rename Command button 3 to "Create Field Definition"

```
Private Sub Command3_Click()  
'call to the function that creates the field definition  
file(ttx)  
'only. If successful it will return a 1, if it fails it  
returns  
'a 0  
  
If CreateFieldDefFile(DAORs, App.Path & "\DAO.ttx", True)  
<> 0 Then  
    MsgBox "Customer Table Field definition successfully  
created"  
Else  
    MsgBox "Failed to create Customer Table field definition  
file"  
End If
```

End Sub

Rename Command button 1 to "Create Report"

```
Private Sub Command1_Click()
'call to the function for creating a report on the DAO
RecordSet
If CreateReportOnRuntimeDS(DAORs, App.Path & "\test.rpt",
App.Path & "\test.ttx", True, True) <> 0 Then
    MsgBox "Field definition created, continue creating
report. Save report before previewing"
Else
    MsgBox "Failed to create Field Definition"
End If
'if the call returns 0 it failed to create the field
definition file (.ttx)
Command4.Enabled = True
End Sub
```

Rename Command button 4 to "Preview Report"

```
Private Sub Command4_Click()
Dim CrDB As crpeAuto.Database
Dim CrTables As crpeauto.DatabaseTables
Dim CrTable As crpeauto.DatabaseTable
On Error GoTo ErrHand:
'Now that the report has been created, we can set the
report object to it (open the report)
Set CrRep = CrApp.OpenReport(App.Path & "\test.rpt")
'Here are 2 methods to Set the data source to the report.
Each uses the SetPrivateData method. The
difference is how they get to the method. Comment out the
one you do not wish to test.
'Begin Method 1:
'This method steps through the object model by setting each
object until the SetPrivateData method is
'reached. We need to set the database, database tables and
database table to get to the method that we
'need: `SetPrivateData
    Set CrDB = CrRep.Database
    Set CrTables = CrDB.Tables
    Set CrTable = CrTables.Item(1)
```

```
'SetPrivateData sets the main reports data to the
daoCustomer recordset in memory
Call CrTable.SetPrivateData(3, DAORs)
`End Method 1:
`Begin Method 2:
`Quick and Dirty method for setting private data. This
method steps through the Object model in one
`line instead of creating and setting separate objects to
reach the desired method.
'SetPrivateData sets the main reports data to the
daoCustomer recordset in memory
Call crRep.Database.Tables.Item(1).SetPrivateData(3, DAORs)

`End Method 2:
`Preview the Report.
    CrRep.Preview
    Exit Sub
ErrHand:
    If Err.Number = 20507 Or Err.Number = 20525 Then
        MsgBox "Error opening report, please create
report then preview"
        Call Command1_Click
    Else
        MsgBox Err.Description & Err.Number
    End If

End Sub

Rename Command button 5 to "EXIT"

Private Sub Command5_Click()
    Unload Me
End Sub
```

## Completing the Application:

Now that the basic application has been created, we must follow a series of steps to successfully generate reports.

We begin first by running the project.

1. Push the button labeled "Populate Recordset". By doing so, we are simply populating the listbox and enabling the remaining buttons.

2. Next, we have the option of, either, pressing the “Create Field Def” or “Create Report” button. In this instance, however, we will press the “Create Report” button. By pushing the “Create Field Def”:
3. We are creating the field definition file (TTX file) only. By pushing the “Create Report”:
4. We are not only creating the TTX file but, also, bringing up the Seagate Crystal Reports designer so that the report can be created at that time.
5. Proceed to create a report and Save it as test.rpt.
6. Finally, we push preview. It generates the report that we just created based off the DAO recordset.
7. Congratulations! You just created an application that reports off a Data Access Object.

## References

The following reference materials were used during the creation of this Tech Support document, or are recommended for further reading on subjects covered by the Topics of this Module Section.

For more information on using the Active Data Driver and DAO see the Seagate Crystal Reports 6.0 Technical Reference, Ch. 6, p. 92 (techref.pdf) “Working with Active Data” and Seagate Crystal Reports Developer’s Manual.

See also “How to Create a Crystal Report using Data Definition Files”

## Contacting Crystal Decisions for Technical Support

We recommend that you refer to the product documentation and that you visit our Technical Support web site for more resources.

**Self-serve Support:**

<http://support.crystaldecisions.com/>

**Email Support:**

<http://support.crystaldecisions.com/support/answers.asp>

**Telephone Support:**

<http://www.crystaldecisions.com/contact/support.asp>