



# Custom Reports

---

*Three Rivers Systems, Inc*

*Published: 20 February 2007*

---

## **Abstract**

This document is designed to provide information necessary to use customized reports in CAMS Enterprise by modifying existing reports. It will discuss the following:

- Setup Considerations
- Customize existing reports
- Create custom reports
- Create custom reports for use with BYORs
- Create advanced custom reports
- How to handle subreports
- How to use .tx files for data
- Add customized reports to report selection dropdowns lists
- SSI files for dropdowns

This document will not discuss creating a Crystal report, except when necessary to add detail to the above items or to prevent errors. The use of Crystal Reports is beyond the scope of this document. It is assumed that you know how to use Crystal Reports.



## Disclaimer

*Three Rivers Systems, Inc. makes no representation or warranties with respect to the contents or use of this guide. Further, Three Rivers Systems, Inc. reserves the right to revise this guide and make changes to its contents at any time without obligation to notify any person or entity of such revisions or changes.*

*In no event will Three Rivers Systems, Inc. be liable to buyer or any other party for any damages, including any lost profits, lost savings, or other special incidental or consequential damages arising out of the use of or inability to use such product, even if Three Rivers Systems, Inc. has been advised of the possibility of such damages, or for any claim by any other party.*

*© 2006 Three Rivers Systems, Inc. All rights reserved. All company and product names included in this site may be trademarks or registered trademarks of their respective companies. The information contained in these pages is subject to change at any time without prior notice.*

# Table of Contents

Table of Contents .....	4
Setup Considerations .....	6
Overview.....	6
Customization Rule .....	6
Elements.....	7
Adding a Report.....	9
Modifying a Pre-Existing CAMS Enterprise Report.....	9
Step-By-Step: Modify a Report with TTX .....	9
Create a Custom Report (Intermediate).....	11
Security.....	11
Step-By-Step: Modify a Report with Views .....	13
Create a Custom Report for BYORs (Intermediate).....	19
Step-By-Step: Open the Starter Report .....	19
Step-By-Step: Link Other Tables or Views.....	20
Step-By-Step: Adding Data Sources .....	21
Step-By-Step: Link Data Sources.....	22
Step-By-Step: Add Report to CAMS Report List.....	23
Create a Custom Report (Advanced).....	25
Displaying The Custom Report .....	29
User Define Routines .....	31
Advanced Variation With Dropdowns .....	33
Final Code .....	38
Adding A Customized Report .....	43
Step-By-Step: Add a Customized Report to a Drop Down .....	43
Report Items .....	46
Subreports .....	46
TTX Files .....	47

Verify Database ..... 48

Type 1000 Reports ..... 49

SSI Files..... 51

    Admissions ..... 51

    Registration ..... 51

    Billing ..... 52

    Financial Aid ..... 52

    Housing ..... 53

    Health ..... 53

    Parking ..... 53

    Development ..... 53

    Faculty ..... 54

    Meeting..... 54

Index ..... 55

# Setup Considerations

## Overview

Using customized reports is easy in CAMS Enterprise. Some items to consider are:

- Knowing the name of the report that you want to customize.
- Knowing where (in what report dropdown list) the customized report will be accessed.
- Knowing what SSI (Server Side Include) file needs to be modified.

You can get the name of a report that you want to modify by opening the report and looking in the status bar of the report to see the name. Or you may use the SSI Files section at the end of this document to find out what SSI file you can look in for the name of the report.

The SSI Files section at the end of this document will assist you in determining which SSI file is used for a specific report dropdown.

Once you have customized your report you will need to modify the appropriate SSI file and then you will be able to access the report immediately from CAMS Enterprise.

## Customization Rule

When customizing a report provided by Three Rivers Systems, we recommend that you first save the report in a custom report subdirectory within the Reports directory. We advise giving the customized report a different name, for example, starting each custom report with the initials of your institution. This will avoid situations where your custom report could be overwritten by Three Rivers Systems during the patch process.



**Note:** *If your custom report uses subreports, those subreports may also need to be copied to the Custom folder for the report to work correctly.*

On occasion, Three Rivers Systems may make changes to CAMS Enterprise that may adversely affect your customized report. To troubleshoot the issue, please check the What's New documentation and CAMS Manager emails for the latest information. If further assistance is needed, please contact your CAMS manager.

Although a fairly straightforward process, issues can arise, so it is suggested that you make a backup copy of any SSI files you need to modify before you begin.

## Elements

There are four (4) elements to a CAMS Enterprise report:

- The report criteria choice screen(s)
- The SSI file that contains the names and options for each of the reports available in the dropdown list.
- The ASP page that will call the query and then the report.
- The Crystal Report

### The report criteria choice screen(s)

Each of the tabs/screens on the report screens allows you to choose the criteria for your report. Most of these fields are exported to record sets which are then included in your Crystal Reports.

Figure 1: Sample Report Criteria page

### SSI Files

SSI (Server Side Include) files are how CAMS Enterprise populates the dropdown lists from which you select reports.

SSI files are usually located in the “SSI” subfolder (for your college database) of your CAMS Enterprise folder. (SSI2 is for the training database.) They are simple text files which contain HTML code and can be edited with any text editor in Windows.

Items appear in the dropdown list in the same order as they appear in the file.

Figure 2: Sample SSI file

The HTML code in the SSI file contains all the information that CAMS Enterprise needs to run a report; an internal system code, the location of the Crystal report file, the asp file that pulls the data and passes it to the report, and the display text you will see for the report in the drop down on CAMS Enterprise.



**Note:** There are two folders under the CAMS Enterprise installation, **SSI** and **SSI2**. The **SSI** folder has all the SSI files associated with your institution's database and **SSI2** has the files associated with the training database. When CAMS Enterprise patches require a CAMS Manager to modify an SSI file for new or modified reports Three Rivers Systems will modify the corresponding SSI file in the SSI2 folder allowing the CAMS Manager to verify that they've made the changes correctly.

### ASP Pages

The ASP pages that do the query and call the Crystal Reports are all stored in the root folder of the CAMS Enterprise installation. Where possible, use the CAMS Enterprise ASP pages that are already created.

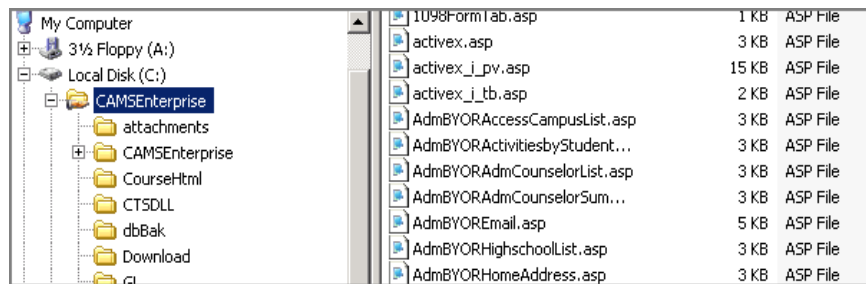


Figure 3: Root Folder

### Crystal Report

After selecting the criteria, selecting the report from the dropdown, and clicking the **Print** button, all the above items work together to create the actual report.

It is very important that all four of these elements be correctly set up and associated with each other.

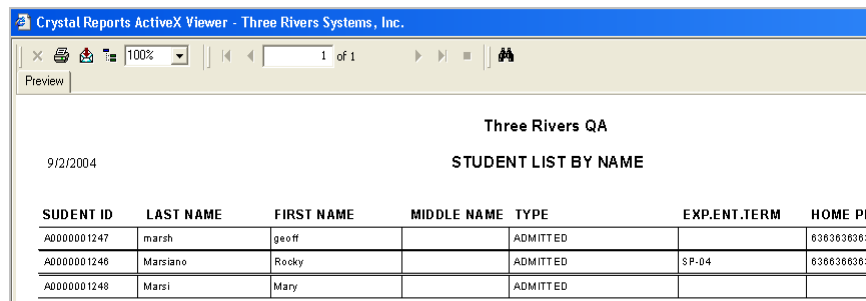


Figure 4: Sample Crystal Report

## Adding a Report

CAMS Enterprise reports are usually created using data sets that are populated from the CAMS Enterprise middleware. Modifying certain reports from CAMS Enterprise allows the developer to **only** use the fields that are available for the original report. If additional fields are required for a report but are not in a data set (TTX) then a custom report must be developed.

Three options to adding additional reports to CAMS Enterprise:

- Modifying a Pre-Existing CAMS Enterprise Report
- Create a Custom Report (Basic)
- Create a Custom Report (Advanced)

## Modifying a Pre-Existing CAMS Enterprise Report

When modifying a pre-existing report care must be taken to not overwrite the original report file.



### Step-By-Step: Modify a Report with TTX

1. Open the report in Crystal Reports. Immediately use Save As to rename the file and save it to the Custom directory under the CAMSEnterprise >Reports directory.
2. Reports that contain a TTX data source under the Field Explorer cannot have additional fields added to the report. This is because the data provided to the TTX is static. Changes to the internal code would have to be made to allow additional fields.

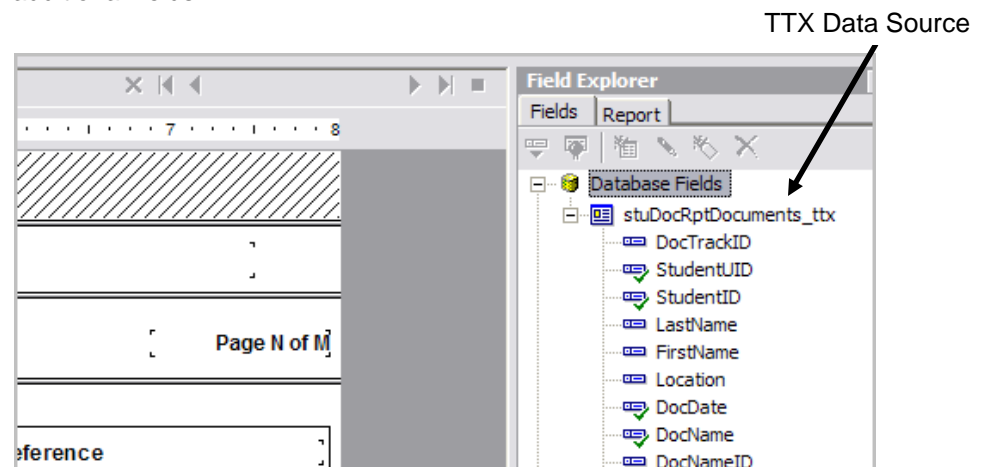


Figure 5: TTX Data Source

3. Modify the report as desired. You can only add fields to the report that are available in the Database Fields section.
4. Save the file.
5. See **Adding a Customize Report** on page 43 to make the report available to users.



**Notes:** Additional fields may not be available to search from in a BYOR screen.

## Create a Custom Report (Intermediate)

The following information will provide the necessary steps to add a custom report that uses a view(s) for the data source.

### Security

Designing the following reports types requires that you have a valid SQL user name and login (integrated security should never be used as the report will fail to run at runtime out of CAMS Enterprise). Three Rivers Systems cannot supply this for you, you need to contact your CAMS Manager, Network Administrator, and/or Database Administrator for this information.



#### Step-By-Step: Modify a Report with Views

Some reports in CAMS use SQL views for their data source. Unlike TTX files the view provides the data to the report. This means that technically you could modify a view with the additional fields needed and they would be available in the report.



**Warning: Never delete fields from the view. You can only add fields.**

Before you can customize the view you need to determine which view it is.

1. Determine that the report you want to customize is using a view.

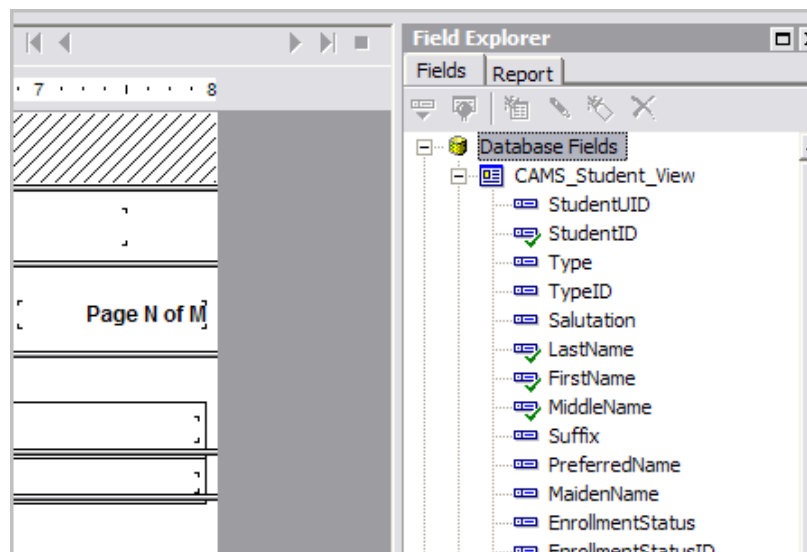


Figure 6: View Data Source

2. You can now open that view in SQL and add your additional fields.

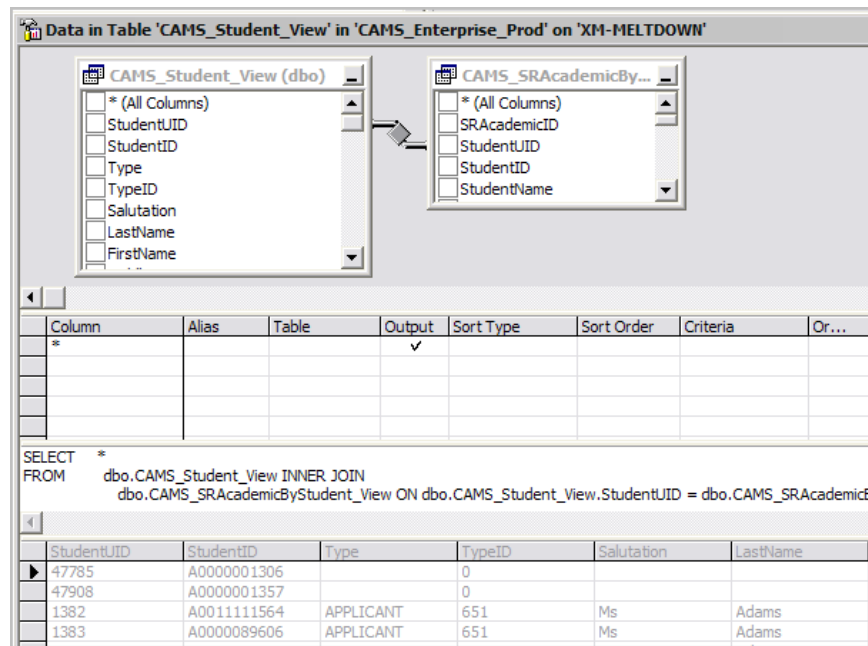


Figure 7: Add Fields to a View

3. Save the view.
4. Open the report in Crystal Reports. Immediately use Save As to rename the file and save it to the Custom directory under the CAMSEnterprise >Reports directory. Click **Database >Verify Database**. Once it is updated it will now display your additional fields for you to add to the report.
5. Customize the report as desired.
6. See **Adding a Customize Report** on page 43 to make the report available to users.
7. **Notes:** Additional fields may not be available to search from in a BYOR screen.





## Step-By-Step: Modify a Report with Views

Before creating the custom report you will need to determine what the report is used for, what data should be provided, the names of all the fields and their data types, and the layout of the report.

The custom report used for this documentation displays a list of all advisors and their advisees.

1. Open Crystal Reports and access the **Database Expert**. Use **Create New Connection** and select **OLE DB (ADO)** and then **Microsoft OLE DB Provider for SQL Server**.

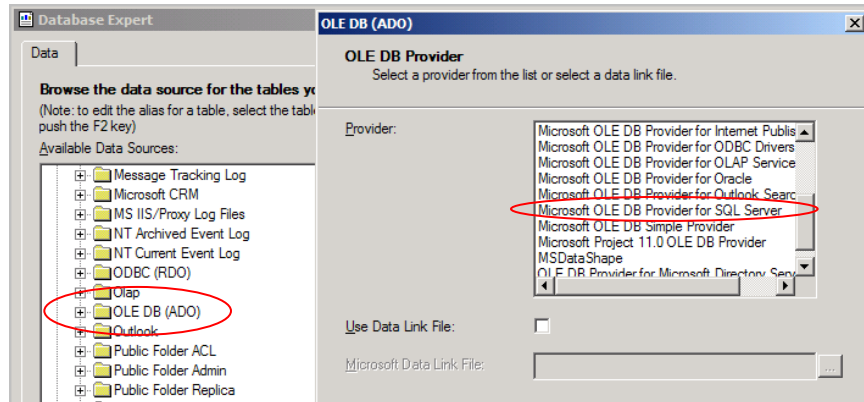


Figure 8: Set New Connection

2. Click **Next**.

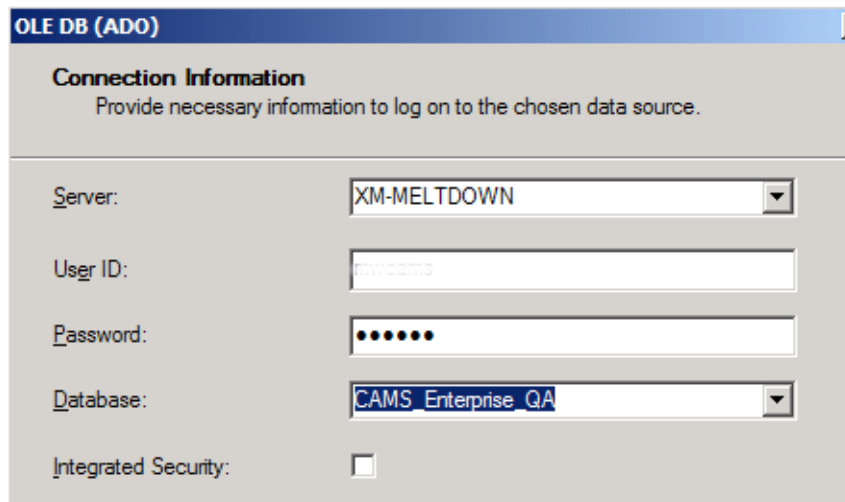


Figure 9: Connection Information

3. A valid User ID and Password must be used that has access to the SQL server tables or views for the CAMS Enterprise database. Do not use Integrated Security.
4. Click **Finish**.

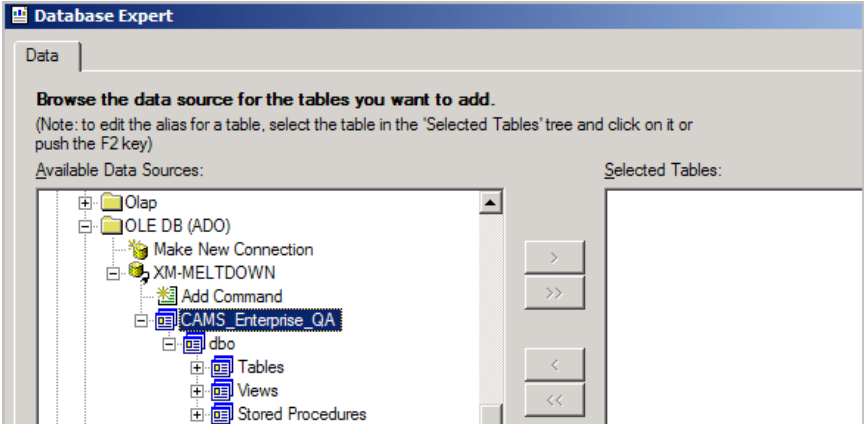


Figure 10: Access to Tables and Views

You can now select the tables and/or views that you would like to use in the report. If possible, create views to provide all the data and use that as the only data source. Crystal Reports does not handle multiple data sources very well.

- 5. Ensure that if more than one table or view is used that they are linked correctly.

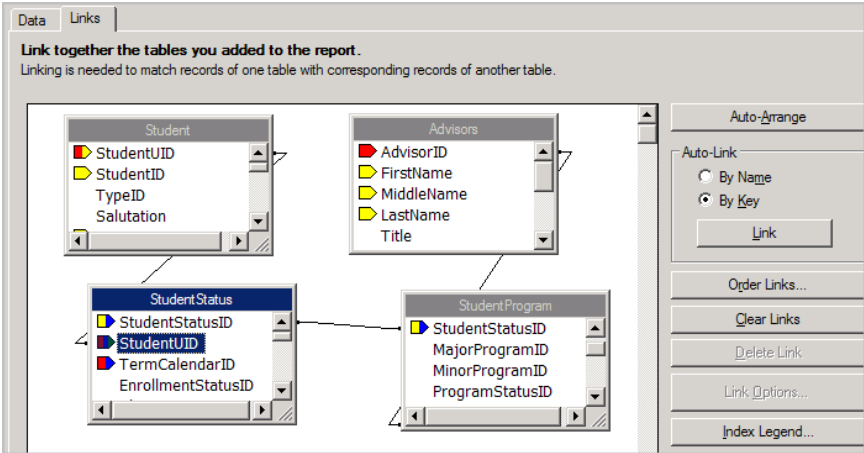


Figure 11: Linked Tables

- 6. Create the report. Notice that there are no TTX files in use. You are working directly from the tables and views.

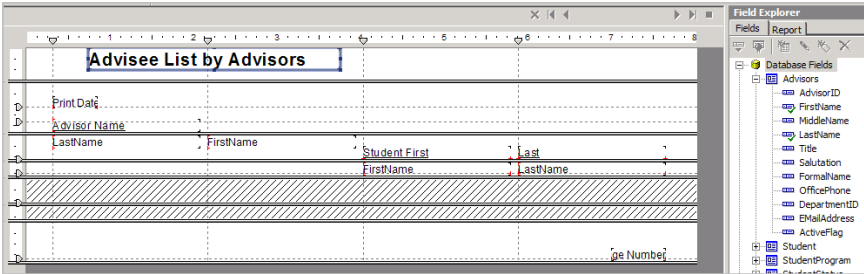


Figure 12: Create the Report

7. Save the report in CAMS Enterprise >Reports >Custom folder. For this example the report is called **TRSAadviseeList.rpt**.
8. Create the ASP page

A special ASP page must be used to run this and any other custom reports created above.

Below is the ASP code necessary to run the custom report. No modification of this code is required. Save the ASP in the CAMS Enterprise folder. For this example it is called **TRSCustomReports.asp**.



**Warning: It must be saved to this location.**

```
<%@ LANGUAGE="VBSCRIPT" %>

<%
    Dim bGetCriteria
    Dim eGetCriteria
    Dim reportName
    Dim strUserID
    Dim strDBName
    Dim strdbSrv
    Dim strSrvName
    Dim rsSubDocuments
    Dim i
    Dim strParse()
    Dim strDBUser
    Dim strDBPassword
    Dim RptCnt

    Dim objDBTable
    Dim objCPProperties

    reportname = trim(Request.QueryString("Rpt"))
'reports\Reportname.rpt"
%>
<!-- #include file="reports\AlwaysRequiredStepsv8.asp" -->
<%
on error goto 0

strUserID = trim(Request.QueryString("uid"))
strDbName = trim(Request.QueryString("db"))
```

```

strDbSrv = trim(Request.QueryString("dbSrv"))
strSrvName = trim(Request.QueryString("Srv"))

Set busObj = Server.CreateObject("CAMSGeneral.dbCAMReports")

Set objDBTable = session("oRpt").Database.Tables(1)
Set objCPPProperties = objDBTable.ConnectionProperties

objCPPProperties.Item("Provider") = "SQLOLEDB"
objCPPProperties.Item("Data Source") = strDbSrv
objCPPProperties.Item("Initial Catalog") = strDbName
Call busObj.SetCRConnProp(objCPPProperties, strDbSrv, strDbName)

RptCnt = session("oRpt").Database.Tables.Count

For i = 1 to RptCnt
    Set objDBTable = session("oRpt").Database.Tables(i)
    objDBTable.Location = objDBTable.Name
Next
%>

```

**If this line is un-commented it will limit the report data to students who have a last name of Turk.** →

‘You can use formulas and variables directly in the Crystal Report or in this ASP page as shown below. Uncomment both lines.

```

'session("oRpt").FormulaSyntax = 0
'session("oRpt").RecordSelectionFormula = "{student.lastname}=""Turk""

```

‘If a subreport is used then for each subreport un-comment the lines below including the data source lines. It should be noted that if a subreport is used it will severely add to the amount of time it takes to display the report. **Therefore, subreports are not recommended.**

```

Set CRSubreports = session("oRpt").OpenSubreport("gg_AddressSUBReport")

objCPPProperties.Item("Provider") = "SQLOLEDB"
objCPPProperties.Item("Data Source") = strDbSrv
objCPPProperties.Item("Initial Catalog") = strDbName

```

```
'Call busObj.SetCRConnProp(objCPProperties, strDbSrv, strDbName)
```

```
'objDBTable.Location = objDBTable.Name
```

```
'Set objDBTable = session("oRpt").Database.Tables(2)
```

```
'objDBTable.Location = objDBTable.Name
```

```
<!-- #include file="reports\MoreRequiredSteps.asp" -->
```

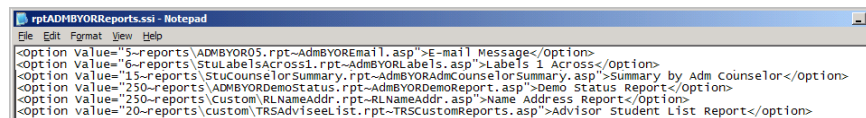
```
<!-- #include file="reports\SmartViewerActivev8.asp" -->
```

9. Save the ASP page.

10. Add the report to a dropdown

Once the report is finished it can be added to any available report dropdowns that are listed below in the **SSI Files** section of this document.

For this example the report will be added to the Admissions >Student >BYOR report screen.



```
rptADMBYORReports.ssi - Notepad
File Edit Format View Help
<option value="5-reports\ADMBYOR05.rpt~AdmBYOREmail.asp">E-mail Message</option>
<option value="6-reports\StuLabelsAcross1.rpt~AdmBYORLabels.asp">Labels 1 Across</option>
<option value="15-reports\StuCounselorSummary.rpt~AdmBYORAdmCounselorSummary.asp">Summary by Adm Counselor</option>
<option value="250-reports\ADMBYORDemoStatus.rpt~AdmBYORDemoReport.asp">Demo Status Report</option>
<option value="250-reports\custom\RLNameAddr.rpt~RLNameAddr.asp">Name Address Report</option>
<option value="20-reports\custom\TRSAdviseeList.rpt~TRSCustomReports.asp">Advisor Student List Report</option>
```

Figure 13: Adding New Custom Report

The last line in the SSI file is used to display the report. Use the following format all on one line no spaces except for what the user will see in the dropdown:

```
<Option
value="20~reports\custom\TRSAdviseeList.rpt~TRSCustomReports.asp"
>Advisor Student List Report</option>
```

- You can use **<Option value="20** for all entries using the TRSCustomReports.asp.
- This is the location of the custom report, **~reports\custom\TRSAdviseeList.rpt**.
- This is the reference to the ASP page that is used to display the custom report, **~TRSCustomReports.asp" >**.
- This is what the user will see displayed in the dropdown, **Advisor Student List Report**. Spaces are allowed here.
- This closes the Option, **</option>**.

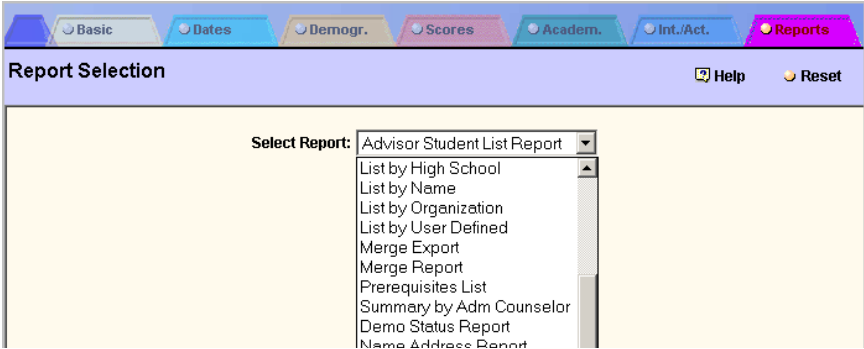


Figure 14: Custom Report Available in Dropdown

## Create a Custom Report for BYORs (Intermediate)

Three Rivers Systems calls these types of reports Direct Connect Reports. They use reports shells, known as Direct Connect Shells, which utilize a permanent temporary table that will contain the data set that matches selected report criteria. These “permanent” tables are placed in the database to handle temporary transactions.

When the BYOR report is printed the selected criteria first populates the associated Direct Connect Shell temporary table with the resulting prospects, students, or faculty. The tables or views that you have added to the custom report are linked to this temporary table and display the correct report data.

For the purpose of demonstration, the following steps will provide instructions on creating a new report in the Student BYOR. However, this same process can be applied to all areas where the Direct Connect Shells have been placed. See the table included in this document for list of such reporting areas.

The purpose of this report is to display a Student and all of their defined contacts.



### Step-By-Step: Open the Starter Report

1. From the **CAMS Enterprise Home** page, click **Admissions >Students >Report BYOR**. Click the **Reports** tab and at the bottom of the list is the **Tmp Table Starter** report.
2. Click **Print**. The report opens. Look at the status bar at the bottom of the report for the actual report name, which is **StuAdmStarter.rpt**.

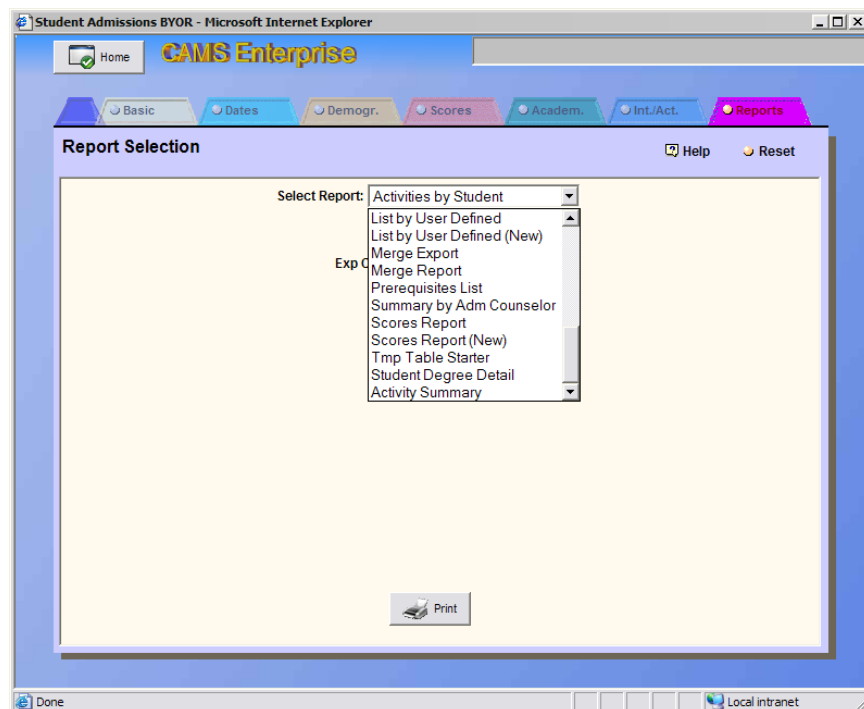


Figure 15: TMP Table Starter Report

Open Crystal Reports and open the **StuAdmStarter.rpt** report.

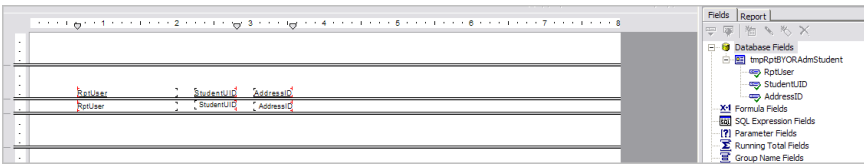


Figure 16: Student Starter Report

The temporary table for this report is called tmpRptBYORAdmStudent and notice there are only three fields available, RptUser, StudentUID, and AddressID.

- RptUser – used to define specific results for the current user running the report. Allows other users to run custom reports at the same time.
- StudentUID – used to identify the student.
- AddressID – used to identify the specific address requested.

Immediately save this report in the custom directory with a custom name. This report will be called TRS\_StudentContact.rpt.

To add additional fields you need to add and link to other tables or views, including custom tables or views.



### Step-By-Step: Link Other Tables or Views

1. On the report access the **Database Expert**. Use **Create New Connection** and select **OLE DB (ADO)** and then **Microsoft OLE DB Provider for SQL Server**.

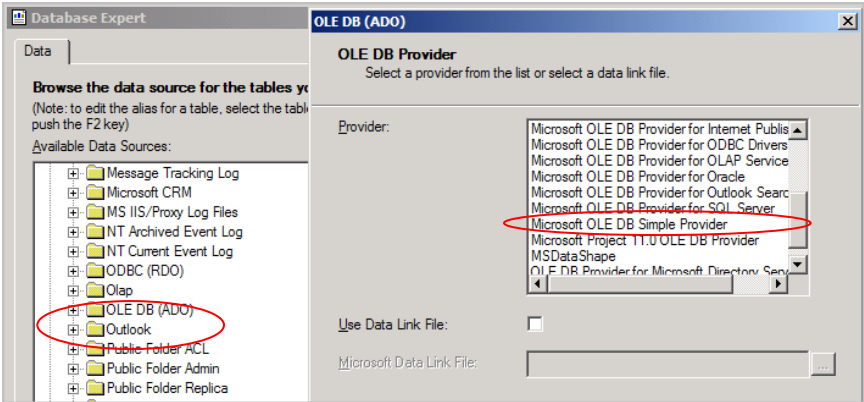


Figure 17: Set New Connection

2. Click **Next**.

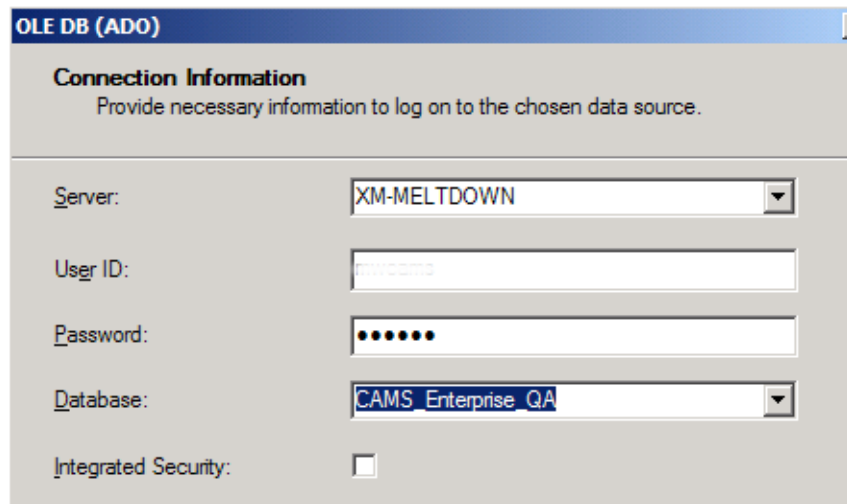


Figure 18: Connection Information

1. Select the CAMS Enterprise SQL server and enter a valid SQL User ID and Password Do not use Integrated Security. Select the CAMS\_Enterprise database.
2. Click **Finish**.

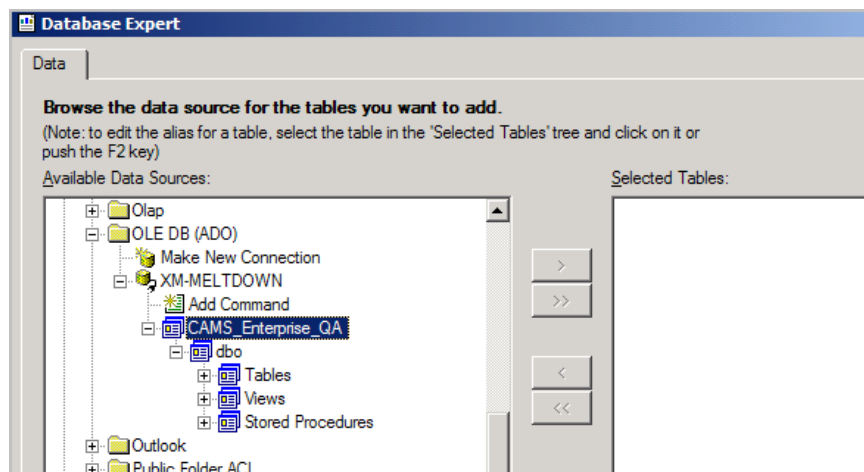


Figure 19: Access to Tables and Views

You can now select the tables and/or views that you would like to use in the report. If possible, create views to provide all the data and use that as the only data source.



## Step-By-Step: Adding Data Sources

1. Some knowledge of the database and its objects is required to pick the proper CAMS supplied tables/views, or you can simply create one of your own. In this example the following TRS supplied view will be used and the appropriate Direct Connect Shell:
  - a. CAMS\_StudentContactList\_View

b. CAMS\_Student\_View



**Note:** If the views/tables that you want to use are not available to you, you need to contact your CAMS Manager, Network Administrator, and/or Database Administrator to see if the SQL User name you used has a minimum of Select rights to those objects (at Runtime CAMS uses it's predefined SQL user that has full rights to all objects in our database so there is no worries about that).

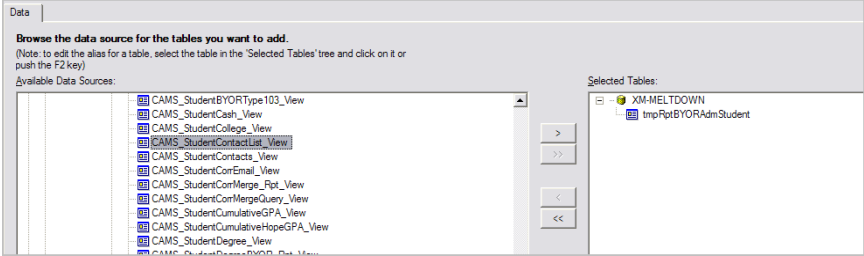


Figure 20: Student Contact View

Click the greater than arrow to add the views to the Selected Tables: column.

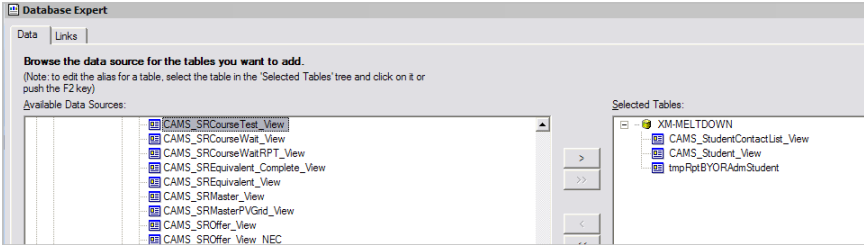


Figure 21: Views Added



**Step-By-Step: Link Data Sources**

1. Once the views have been added to the report, click on the **Links** tab at the top of the Database Expert window. The default join type is an Inner Join. By default, Crystal Reports links by all matching field names/types with an inner join.

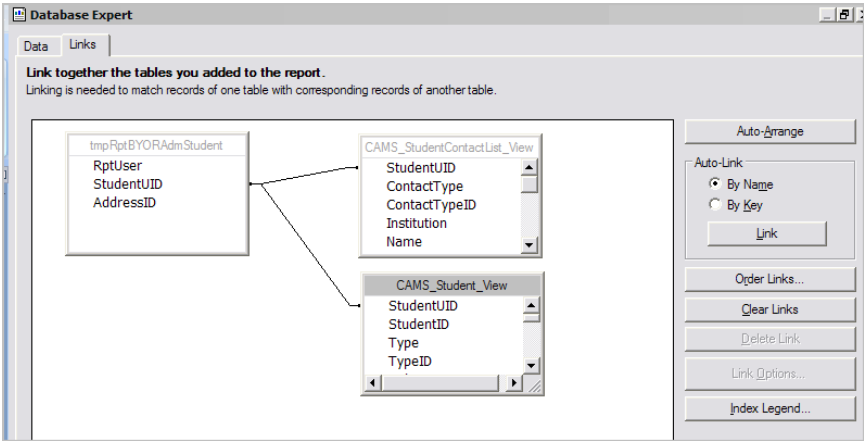


Figure 22: Linked Data Sources

2. Click **OK** and the report will be displayed in the preview tab.
3. Create the report. Notice that there are no TTX files in use. You are working directly from the tables and views. You can delete the existing fields from the report and then add the fields you want.

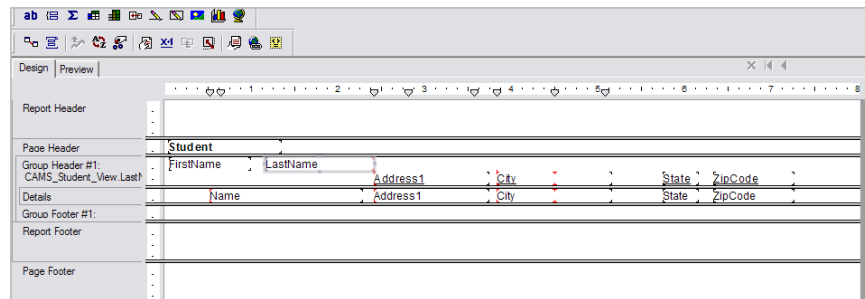


Figure 23: Sample Report Design

The views that were added will display every student in the database and their contacts. By adding the temporary table you will be able to use the BYOR to limit the list of students to only those that meet the selected criteria.



### Step-By-Step: Add Report to CAMS Report List

1. The next step is to add this report to the Student BYOR report format dropdown. Open the rptADMBYORReports.ssi in a text editor. This file is located in the SSI folder for the College database and in the SSI2 folder for the training database.



Figure 24: rptADMBYORReports.ssi file

2. Find the line that contains the Tmp Table Starter text. You must copy this line exactly as written and paste it at the bottom of the list. (Where ever this line is located determines where it displays in the drop down list.)

You will only change the report name and location and the drop down description that your users will see. Everything else in the line must remain the same.

```
<option value="1000~reports\StuAdmStarter.rpt~ADMStudentBYORType1000.asp">Tmp Table Starter</option>
```

Change the StuAdmStarter.rpt text to **custom\TRS\_StudentContact.rpt** and change the Tmp Table Starter to **TRS Student Contacts**.

The line should now read:

```
<option value="1000~reports\custom\TRS_StudentContact.rpt~  
ADMStudentBYORType1000.asp">TRS Student Contacts</option>
```

By adding the college initials to the beginning of the report name and the drop down display Three Rivers Systems will recognize that it is a custom report and your users will know that that is the report they should use.

3. Save this file.

You can now go to the Student BYOR and your report will be listed. Select your BYOR criteria and then print your custom report.

## Create a Custom Report (Advanced)

The following information will provide the necessary steps to add a custom report, using a custom stored procedure, custom TTX file and the User Defined Routine section of CAMS Enterprise.

Only basic steps will be listed in creating a store procedure, ttx file, and custom report. Additional information about these items can be found using your SQL and Crystal Reports documentation.

### Data for the Custom Report

Before creating the custom report you will need to determine what the report is used for, what data should provided, the names of all the fields and their data types, and the layout of the report.

The custom report used for this documentation displays a list of all ethnicities and total count of students who are registered for a specific term.

<b><u>Ethnicity Report</u></b>	
8/2/2005	
<b><u>SP-04 (Official)</u></b>	
<b><u>Ethnic Group</u></b>	<b><u>Student Count</u></b>
American Indian / Alaska Native	17
Asian	3
White	1
	4
<b>Totals:</b>	<b>25</b>

Figure 25: Ethnicity Custom Report

### Create the TTX file

To create a custom report that uses a data set you need to create a TTX file that contains all the needed fields, their data type, length and sample data.

```

Ethnic_Group      String  40      OTHER
Ethnic_Count      Long    8
S_Status          String  40      official
    
```

Figure 26: Report TTX file

Add the TTX file as the data source to your blank custom report.

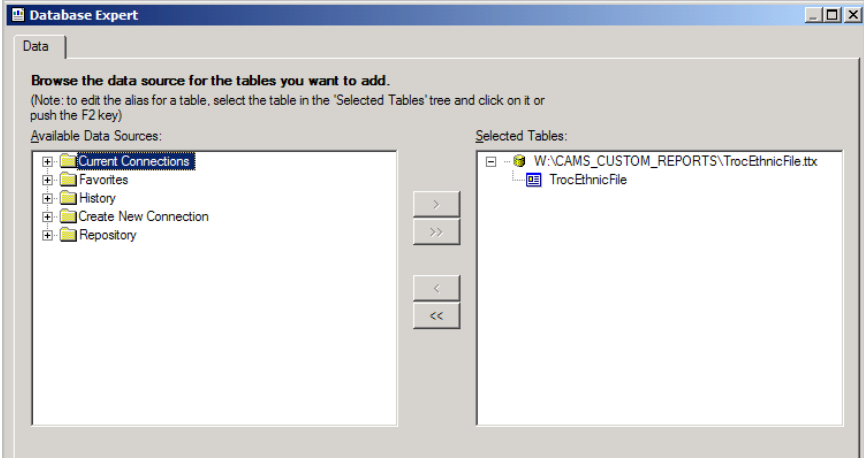


Figure 27: Adding TTX As Data Source

### Create The Report

Once the data source is attached to the report you can design the report.

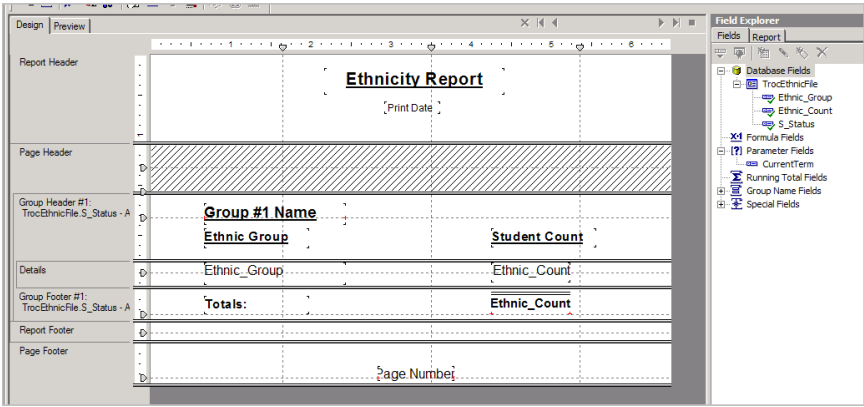


Figure 28: Designing The Report

By clicking on the preview tab you can view the sample data from the TTX file.

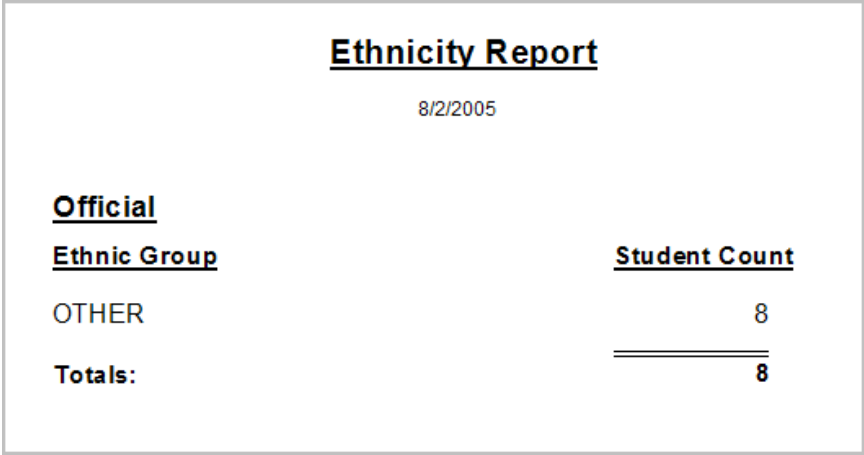
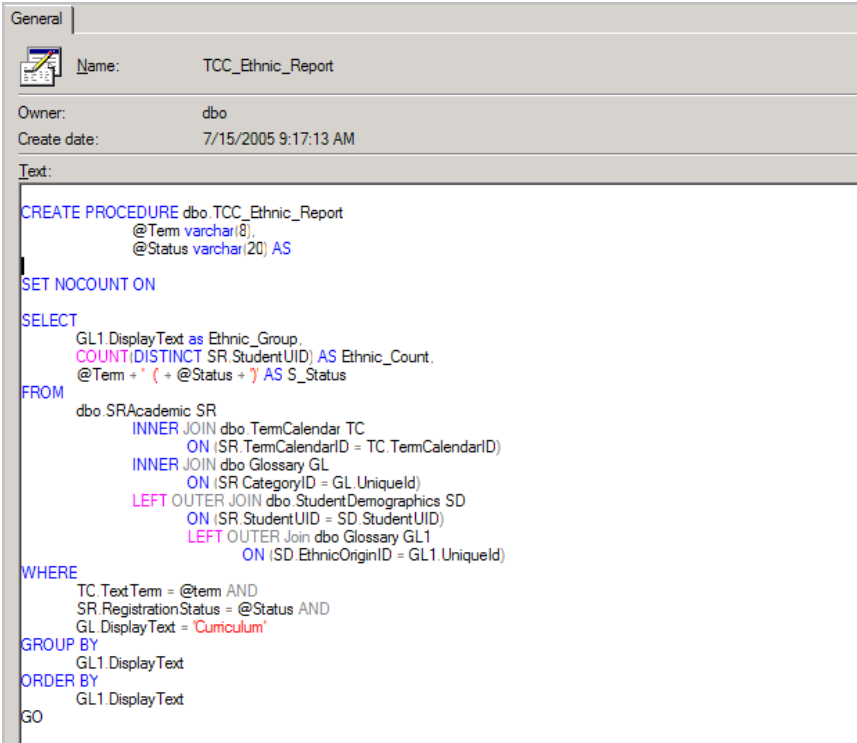


Figure 29: Previewed Report

## Create Stored Procedure

A stored procedure is used to build the data set directly from the database that will be displayed when the report is run.



```

CREATE PROCEDURE dbo.TCC_Ethnic_Report
    @Term varchar(8),
    @Status varchar(20) AS
SET NOCOUNT ON
SELECT
    GL1.DisplayText as Ethnic_Group,
    COUNT(DISTINCT SR.StudentUID) AS Ethnic_Count,
    @Term + ' (' + @Status + ')' AS S_Status
FROM
    dbo.SRAcademic SR
        INNER JOIN dbo.TermCalendar TC
            ON (SR.TermCalendarID = TC.TermCalendarID)
        INNER JOIN dbo.Glossary GL
            ON (SR.CategoryID = GL.UniqueID)
        LEFT OUTER JOIN dbo.StudentDemographics SD
            ON (SR.StudentUID = SD.StudentUID)
        LEFT OUTER JOIN dbo.Glossary GL1
            ON (SD.EthnicOriginID = GL1.UniqueID)
WHERE
    TC.TextTerm = @term AND
    SR.RegistrationStatus = @Status AND
    GL.DisplayText = 'Curriculum'
GROUP BY
    GL1.DisplayText
ORDER BY
    GL1.DisplayText
GO
  
```

Figure 30: Stored Procedure

## Displaying The Custom Report

To display the custom report using the Crystal Report viewer and the stored procedure the calling ASP page must be created. The following code is a sample of the ASP page used in designing the Ethnicity report used for this example. To simplify the coding some items are hard-coded, such as the connection information. Creating web pages with variables instead of hard-coding is beyond the scope of this document.

### **Ethnic\_Report.asp**

---

```

<%@language = VBScript %>
<%
Response.buffer = true
Dim strUserID
Dim strSemester
Dim strType
Dim strDBName
Dim strSvrName
Dim strDBSrv
Dim strError, CriteriaRS
Dim objCmd, objParam, objParam1, objRS, obj, busAdmitObj
Dim strOut
Dim reportname

reportname = "reports\TrocEthnicReport.rpt"

strUserID = trim(Request.QueryString("uid"))
Dim strConnection
Set obj = Server.CreateObject("CAMSAuthorizeDB.busConnect")
strConnection = obj.DBConnString("Servername", "CAMs_Enterprise")
Set objDBConn = Server.CreateObject("ADODB.Connection")
objDBConn.CommandTimeout = 600
objDBConn.open(strConnection)
%>
<!-- #INCLUDE FILE="reports\AlwaysRequiredStepsv8.asp" -->
<%
Set objRS = Server.CreateObject("ADODB.Recordset")

```

**Report Name** →

**Server Name** →

**Database** →

**Must include this line** →

**Run stored procedure**

```

objRS.Open "TCC_Ethnic_Report 'SP-04', 'Official'", objDBConn
Set session("oData") = objRS
if session("oData").BOF or session("oData").EOF then
else
    session("oData").MoveFirst
end if
session("oRpt").DiscardSavedData
' link the data to the report
Set Database = session("oRpt").Database
Database.Tables.Item(1).SetPrivateData 3, session("oData")
%>
<!-- #INCLUDE FILE="reports\MoreRequiredSteps.asp" -->
<!-- #INCLUDE FILE="reports\SmartViewerActiveXv8.asp" -->
<%
objRS.close
objDBConn.close
Set objRS = Nothing
Set objDBConn = Nothing
%>

```

**Stored procedure parameters**

**Must include these lines**



**Note:** The above ASP code can be used for any custom report by changing the following items:

- **Report Name**
- **Server Name**
- **Database**
- **Stored Procedure Name and Parameters** (Parameters might not be necessary if they are hard-coded in the stored procedure.)

# User Define Routines

The User Defined Routine module is a location on the CAMS Enterprise menu for institutions to place custom pages or routines. This allows the users access to these items without having to access another location outside of CAMS Enterprise.

There is a SQL table called CAMSMenu that contains a listing of all available menu items.



**Warning: Changing this table incorrectly can cause problems with CAMS Enterprise.**

This table lists the order of the menu items.

CAMSMenuID	MenuItemName	CAMSModulesID	MenuLevel	ParentOrder	ItemOrder	MenuAccess	ProgramName	ItemIconNum	ShowOnUserMenu
140	PCI Import	13	7	5	1	0	none	2	Yes
15	User Defined Routh	14	0	0	0	0	Tasks	0	Yes
28	User Audit	14	1	0	0	0	sysUserAuditTab.a	2	No
266	Adm Custom Rep	14	2	0	0	0	admrpts.asp	2	Yes
263	Custom 3	14	3	0	0	0	notyet.asp	2	Yes
235	Add	99	5	2	1	0		0	No
337	Change	99	5	2	2	0		0	No

Figure 31: CAMSMenu table

Only the section dealing with the User Defined Routines will be discussed. It is not recommended that any other menu item be changed.

The first column lists the menu item name. Based on a hierarchal model the MenuItemName “User Defined Routines” is the first CAMSModulesID 14. The successive fields, MenuLevel, ParentOrder, and ItemOrder are all set to zero, which makes this the top level menu item.

The “ADM Custom Reports” menu item name is also a CAMSModulesID 14 but it has a MenuLevel of 2, which makes it a submenu item to User Defined Routines.

The ProgramName field contains the page to be opened when the menu item is selected. For **Adm Custom Reports** this is **admrpts.asp**.

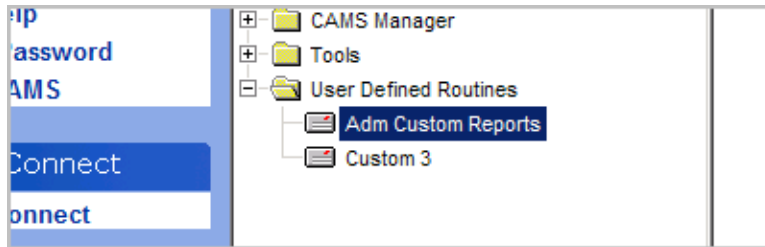


Figure 32: Admissions Custom Reports Menu Item

Clicking on **ADM Custom Reports** will open the **admrpts.asp** page.



Figure 33: ADMRPTS.ASP page

This page has all the custom report links. Clicking on a link will now use the Stored Procedure to produce the data set that the custom report will utilize when opening with the Crystal Reports viewer.

<b><u>Ethnicity Report</u></b>	
8/2/2005	
<b><u>SP-04 (Official)</u></b>	
<b><u>Ethnic Group</u></b>	<b><u>Student Count</u></b>
	17
American Indian / Alaska Native	3
Asian	1
White	4
<b>Totals:</b>	<b>25</b>

Figure 34: Ethnicity Custom Report

See **Adding a Customize Report** on page 43 to make the report available to users.

## Advanced Variation With Dropdowns

The previous custom reports uses static or hard-coded variable information, which requires the user to modify either the ASP page or the Stored Procedure anytime there is a change. For example, if the user wanted to run the report for one term and then run it again for a different term.

In the above Ethnic\_Report.asp page code the stored procedure parameters are hard-coded into the code. In this next example the term parameter will come from a dropdown on the **admrpts.asp** page.

As stated previously only basic information will be presented in this document. There are many ways to design an ASP page and/or call stored procedures.



Figure 35: Dropdown Fields

Figure **Figure 35** above displays two dropdown fields. These dropdown fields are created using SSI files that Three Rivers Systems, Inc creates for all CAMS Enterprise dropdown fields. The code to create a dropdown is:

```
<%
Dim ssiPath
Dim uiCommon
ssiPath = Request.QueryString("ssipath")
If Right(ssiPath, 1) = "\" Then
Else
    ssiPath = ssiPath & "\"
End If
Set uiCommon = Server.CreateObject("uiCECommon.ceCommon")
%>
```

```

<select id=uiTerm>
<%
    call uiCommon.Readssi(ssiPath & "TermCalendar.ssi")
%>
</select>

```

The uiCommon lines use an object created by Three Rivers Systems, Inc to retrieve the proper information from the selected SSI file. These dropdowns will be identical to the dropdowns in CAMS Enterprise.

The Select statement reads the information and creates the dropdown. And this dropdown is called **uiTerm**.

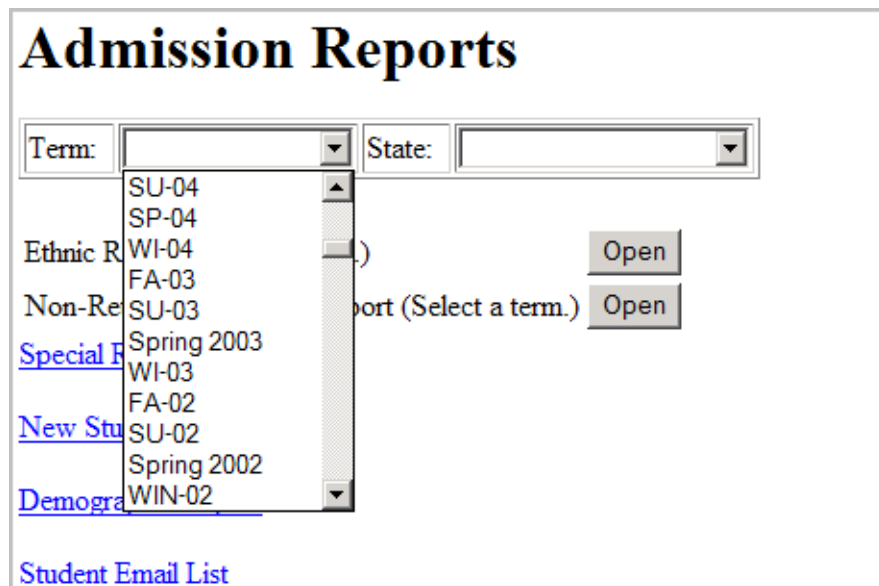


Figure 36: Selectable Dropdown

The TermCalendar.ssi file contains all the elements needed for a dropdown.

```

TermCalendar.ssi - Notepad
File Edit Format View Help
<OPTION VALUE="0~"></OPTION>
<OPTION VALUE="157~FA-12">FA-12</OPTION>
<OPTION VALUE="156~FA-09">FA-09</OPTION>
<OPTION VALUE="155~FA-08">FA-08</OPTION>
<OPTION VALUE="154~FA-07">FA-07</OPTION>
<OPTION VALUE="153~FA-06">FA-06</OPTION>
<OPTION VALUE="152~FA-05">FA-05</OPTION>
<OPTION VALUE="151~SU-05">SU-05</OPTION>
<OPTION VALUE="150~SP-05">SP-05</OPTION>
<OPTION VALUE="149~WT-04">WT-04</OPTION>
<OPTION VALUE="148~FA-04">FA-04</OPTION>
<OPTION VALUE="147~SU-04">SU-04</OPTION>
<OPTION VALUE="146~SP-04">SP-04</OPTION>
<OPTION VALUE="145~WT-03">WT-03</OPTION>

```

Figure 37: TermCalendar SSI file

Three Rivers Systems, Inc uses IDs when creating scripts or stored procedures. Looking at **Figure 37** the information after the Value= is the ID number of the particular term. Notice the ID is separated from the TextTerm by a tilde (~).

When using variables instead of hard-coded parameters there must be a way to pass the selected ID from the dropdown to the stored procedure used to populate the report. In this example a button is created and the OnClick option is used to pass the data.

Notice in Figure **Figure 35** that the first two reports are no longer hyperlinks that call the report asp page directly. There is now a button to the right of each report.

The code used for the Ethnic report button is:

```
Ethnic Report (Select a term.) <Input type=button id=Ethnicbtn value="Open">
```

This displays the text for the report name and a button.

To retrieve the ID number from the newly created dropdown you can set the script to find the tilde in the **uiTerm.value** (`delim = instr(1,uiTerm.value,"~")`) then only select the value to the left of the tilde (`Left(uiTerm.value,delim-1)`).

The word **key** is the variable used to hold the value retrieved from the dropdown.

The **OnClick** information created to pass the selected ID to the report page looks like this:

```
sub Ethnicbtn_Onclick()
    delim = instr(1,uiTerm.value,"~")
    key = Left(uiTerm.value,delim-1)
    window.location="ethnic_report.asp?uiTerm=" & cstr(key)
end sub
</script>
```

When the button is clicked the **Ethnic\_Report.asp** page is called and the uiTerm variable is passed along to that page.

Some changes need to be made to the Ethnic\_Report.asp page to handle the incoming variable.

A variable must be declared to accept the value of the uiTerm variable.

```
Dim TermID
```

The TermID variable must equal the uiTerm variable.

```
termID = trim(Request.QueryString("uiterm"))
```

The line calling the stored procedure must be able to accept the variable. Previously the line looked like this:

```
objRS.Open "TCC_Ethnic_Report 'SP-04', 'Official'", objDBConn
```

Now it should look like this:

```
objRS.Open "TCC_Ethnic_Report " & cstr(termID) & "', 'Official'",
objDBConn
```

Notice that the 'SP-04" hard-coded term has been replaced with our **termID** variable. This will allow users to now choose what term they want without having to manually change any code.

## Final Code

Code for the admrpts.asp and the ethnic\_report.asp page are displayed below to allow you to view all the code for each page in one location to assist you in understanding what has been done.

### ADMRpts.asp

---

```

<%
    Dim uiCommon

    Set uiCommon =
    Server.CreateObject("uiCECommon.ceCommon")
%>

<script Language="vbScript">
sub Ethnicbtn_Onclick()

    delim = instr(1,uiTerm.value,"~")
    key = Left(uiTerm.value,delim-1)

    window.location="ethnic_report.asp?uiTerm=" & cstr(key)
end sub
</script>

<Script>
sub NonRtnbtn_Onclick()

    delim = instr(1,uiTerm.value,"~")
    key = Left(uiTerm.value,delim-1)
    window.location="nonreturningstudents.asp?uiTerm=" &
cstr(key)
end sub
</script>
<BODY>

<B><font size=6>Admission Reports</font></B><P>

```

```

<table border = 1>
<tr>
    <Td align=right>Term: &nbsp;</td>
    <td><select id=uiTerm>
<%
        call uiCommon.Readssi("TermCalendar.ssi")
%>
    </select>
</td>

    <TD Align=right>State: &nbsp;</TD>
    <TD><Select id=uiState>
<%
        call uiCommon.Readssi("State_Codes.ssi")
%>
    </select>
</td>
</tr>
</table>

<table>
<tr><td>&nbsp;</td></tr>
<tr>
    <td>Ethnic Report (Select a term.)</td><td><Input type=button
id=Ethnicbtn value="Open"></td>
</tr>
<tr>
    <td>Non-Returning Students Report (Select a
term.)</td><td><Input type=button id=NonRtnbtn value="Open"></td>
</tr>
</table>

<A Href="ethnic_Report.asp">Special Report for President</A><p>

<A Href="ethnic_Report.asp">New Students by State</A><p>

<A Href="ethnic_Report.asp">Demographic Report</A><p>

```

```
<A Href="ethnic_Report.asp">Student Email List</A><p>
</BODY>
```

---

## **Ethnic\_Report.asp**

---

```
<%@language = VBScript %>
```

```
<%
```

```
Response.buffer = true
```

```
dim termID
```

```
Dim strUserID
```

```
Dim strSemester
```

```
Dim strType
```

```
Dim strDBName
```

```
Dim strSvrName
```

```
Dim strDBSrv
```

```
Dim strError, CriteriaRS
```

```
Dim objCmd, objParam, objParam1, objRS, obj, busAdmitObj
```

```
Dim strOut
```

```
Dim reportname
```

```
reportname = "reports\TrocEthnicReport.rpt"
```

```
strUserID = trim(Request.QueryString("uid"))
```

```
termID = trim(Request.QueryString("uiterm"))
```

```
response.write term
```

```
Dim strConnection
```

```
Set obj = Server.CreateObject("CAMSAuthorizeDB.busConnect")
```

```
strConnection = obj.DBConnString("xm-meltdown", "CAMSEnterprise_QA")
```

```
Set objDBConn = Server.CreateObject("ADODB.Connection")
```

```

objDBConn.CommandTimeout = 600

objDBConn.open(strConnection)

%>

<!-- #INCLUDE FILE="reports\AlwaysRequiredStepsv8.asp" -->

<%

Set objRS = Server.CreateObject("ADODB.Recordset")

objRS.Open "TCC_Ethnic_Report " & cstr(termID) & ", 'Official'", objDBConn

Set session("oData") = objRS

if session("oData").BOF or session("oData").EOF then
else
    session("oData").MoveFirst
end if

session("oRpt").DiscardSavedData

' link the data to the report
Set Database = session("oRpt").Database
Database.Tables.Item(1).SetPrivateData 3, session("oData")

%>

<!-- #INCLUDE FILE="reports\MoreRequiredSteps.asp" -->
<!-- #INCLUDE FILE="reports\SmartViewerActiveXv8.asp" -->

<%

objRS.close
objDBConn.close

Set objRS = Nothing

```

Set objDBConn = Nothing

%>

---

## Adding A Customized Report

Whether you modify a pre-existing CAMS Enterprise report or create a custom report making it available to users through a report dropdown list is the same.



### Step-By-Step: Add a Customized Report to a Drop Down

1. Ensure that you know the name and location of the report, the name of the ASP page that will get the data and the text that will display in the report dropdown list.
2. Put your custom report in the **~/reports/custom** folder. If you have copied a preexisting report, be sure to rename the custom report with a different name. Otherwise, there is a chance that it may be overwritten during a patch.

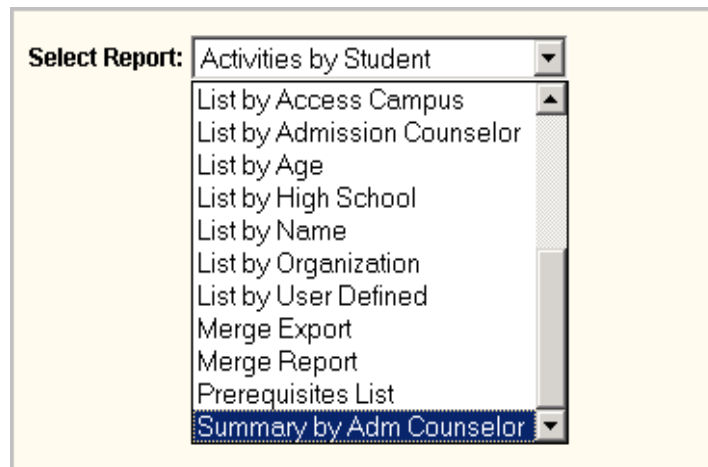


Figure 38: Original Report Dropdown list

For example, the **Summary by Adm Counselor** report is to be modified. The Crystal Report name is **stuCounselorssummary.rpt**. This report is copied, modified, and placed in the **~/reports/custom** folder and renamed to **ABCstuCounselorssummary.rpt**. (ABC could stand for the institution's initials.) By renaming it with something that identifies it as a custom report this will help Three Rivers Systems troubleshoot report problems and also prevent overwriting the report.

3. Open the appropriate SSI file. (A list of Report Dropdown lists and the SSI files that populate them is listed later in this document.) Modify as shown below.

In this case the SSI file is called **rptADMBYORReports.ssi**.

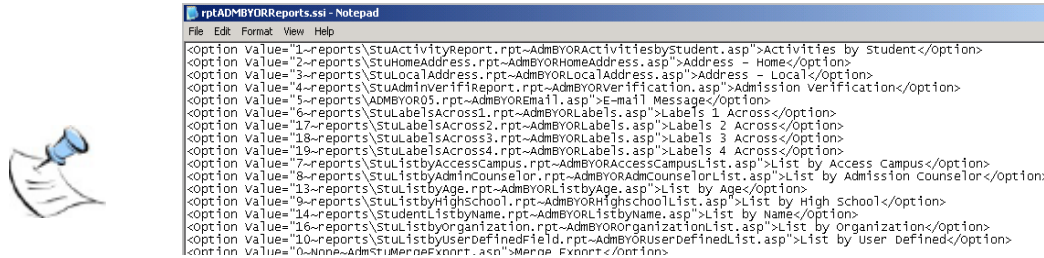


Figure 39: SSI file

### Modify an SSI File

Notice the last line in the file.

```
<Option Value="20~reports\StuCounselorSummary.rpt
~AdmBYORAdmCounselorSummary.asp"> Summary by Adm Counselor
</Option> (This is all on one line in the file.)
```

An explanation of the line is listed below.

The quoted portion of the line,

("20~reports\StuCounselorSummary.rpt~AdmBYORAdmCounselorSummary.asp"),

is what controls how the report is run. The tildes in this line are delimiters for the different parameters. This line will be broken down below by parameter (in bold).

```
<Option Value="20~reports\StuCounselorSummary.rpt
~AdmBYORAdmCounselorSummary.asp">Summary by Adm Counselor
</Option>
```

The number in the first position of the line is how CAMS Enterprise knows what data to fill on the report. This is an internal code, and for this purpose, should not be modified.



**Warning: When copying a line the Option Value should not be changed. This is not the list order number.**

```
<Option Value="20~reports\StuCounselorSummary.rpt
~AdmBYORAdmCounselorSummary.asp">Summary by Adm Counselor
</Option>
```

The next item in the line is the relative location of the Crystal report file itself. This should be changed to "reports\custom\ABCStuCounselorSummary.rpt". This tells CAMS Enterprise to go get this new custom report to run instead of the original.



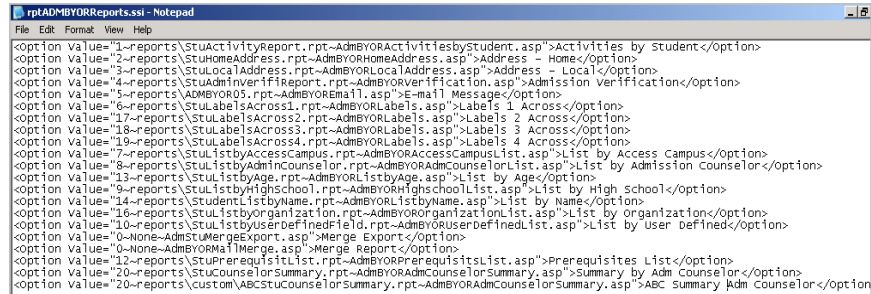
**Note:** The subdirectory \custom is shown as a reminder that you should place your customized reports in this directory to prevent you files from being overwritten during patching.

```
<Option Value="20~reports\custom\ABCStuCounselorSummary.rpt
~AdmBYORAdmCounselorSummary.asp">ABC Summary Adm Counselor
</Option>
```

This section of the line tells CAMS Enterprise which asp file to load the report with. For this purpose, this should not be modified.

```
<Option Value="20~reports\custom\ABCStuCounselorSummary.rpt
~AdmBYORAdmCounselorSummary.asp">ABC Summary Adm Counselor
</Option>
```

This section is what will be displayed in the report dropdown list on the screen. If you have kept the original CAMS Enterprise report, this should be modified to distinguish the two on screen.



```
rptADMBYORReports.ssi - Notepad
File Edit Format View Help
<Option Value="1~reports\StuActivityReport.rpt~AdmBYORActivitiesbyStudent.asp">Activities by Student</Option>
<Option Value="2~reports\StuHomeAddress.rpt~AdmBYORHomeAddress.asp">Address - Home</Option>
<Option Value="3~reports\StuLocalAddress.rpt~AdmBYORLocalAddress.asp">Address - Local</Option>
<Option Value="4~reports\StuAdmInverIfReport.rpt~AdmBYORVerification.asp">Admission verification</Option>
<Option Value="5~reports\ADMBYOR05.rpt~AdmBYOREmail.asp">E-mail Message</Option>
<Option Value="6~reports\StuLabelsAcross1.rpt~AdmBYORLabels.asp">Labels 1 Across</Option>
<Option Value="7~reports\StuLabelsAcross2.rpt~AdmBYORLabels.asp">Labels 2 Across</Option>
<Option Value="8~reports\StuLabelsAcross3.rpt~AdmBYORLabels.asp">Labels 3 Across</Option>
<Option Value="9~reports\StuLabelsAcross4.rpt~AdmBYORLabels.asp">Labels 4 Across</Option>
<Option Value="10~reports\StuListbyAccessCampus.rpt~AdmBYORAccessCampusList.asp">List by Access Campus</Option>
<Option Value="11~reports\StuListbyAdmCounselor.rpt~AdmBYORAdmCounselorList.asp">List by Adm Counselor</Option>
<Option Value="12~reports\StuListbyAge.rpt~AdmBYORListbyAge.asp">List by Age</Option>
<Option Value="13~reports\StuListbyHighSchool.rpt~AdmBYORHighSchoolList.asp">List by High School</Option>
<Option Value="14~reports\StuListbyName.rpt~AdmBYORListbyName.asp">List by Name</Option>
<Option Value="15~reports\StuListbyOrganization.rpt~AdmBYOROrganizationList.asp">List by Organization</Option>
<Option Value="16~reports\StuListbyUserDefinedField.rpt~AdmBYORUserDefinedList.asp">List by user defined</Option>
<Option Value="17~None-AdmStuMergeExport.asp">Merge Export</Option>
<Option Value="18~None-AdmBYORMerge.asp">Merge Report</Option>
<Option Value="19~reports\StuPrerequisitesList.rpt~AdmBYORPrerequisitesList.asp">Prerequisites List</Option>
<Option Value="20~reports\StuSummary.rpt~AdmBYORAdmCounselorSummary.asp">Summary by Adm Counselor</Option>
<Option Value="20~reports\custom\ABCStuCounselorSummary.rpt~AdmBYORAdmCounselorSummary.asp">ABC Summary Adm Counselor</Option>
```

Figure 40: Modified SSI File



**Note:** It is also possible to change the order in which items are listed in the drop down on the screen. The items are listed in the same order they appear in the SSI file. You only need to change the order of the items in the SSI file. Care should be taken to fully copy each line when moving it around in the file.

If everything is done correctly close the window that has the report dropdown list on it and then reopen the screen. This will refresh the dropdown. You should now see your custom report listed.

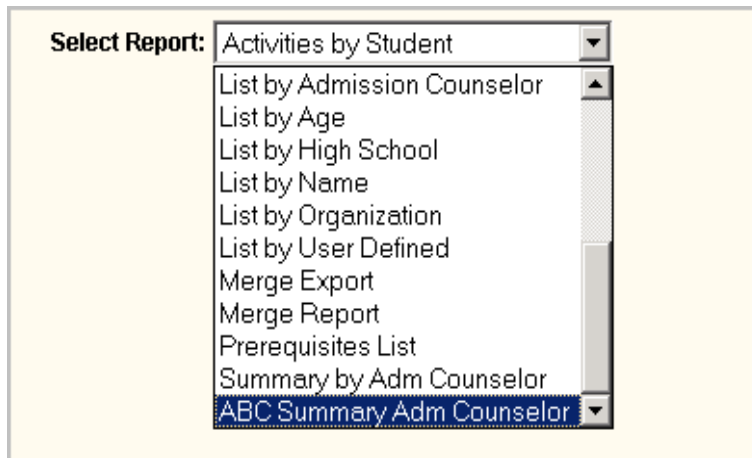


Figure 41: Modified Report Dropdown list

4. Highlight your custom report in the report dropdown list and click **Print**. Your custom report should appear.

# Report Items

## Subreports

Crystal Reports can have subreports as part of the main report. If a custom report, that has subreport(s), is created and placed in the ~\reports\custom\ folder an error may occur when the report is printed.

There are two ways to correct this problem.

- Place a copy of the subreport(s) in the ~\reports\custom\ folder.
- Remap the subreport(s) in the custom report to point to its current location under the ~\reports\ folder.

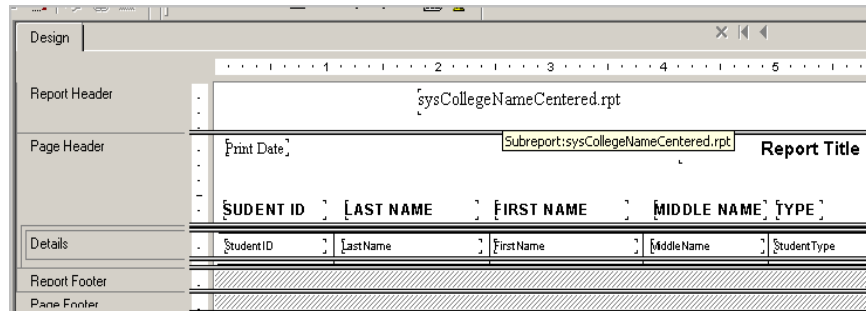
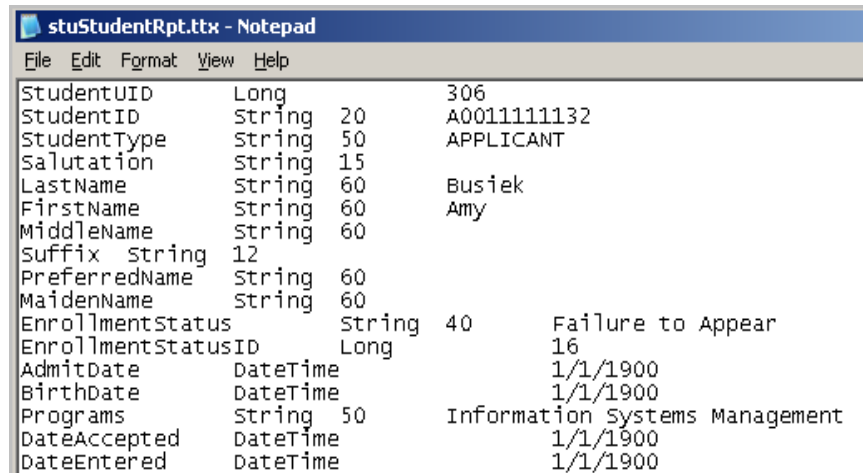


Figure 42: Crystal Report with Subreport (in Crystal Designer)

## TTX Files

A Data Definition file is a tab separated text file, with a TTX file extension. Basically, it is an outline of a table structure. Each line in a TTX file contains a field name, data type, length (if the data type is a string), and one piece of sample data. The contents of the sample Data Definition file, stuStudentRpt.ttx, are shown below.



Field Name	Data Type	Length	Sample Data
StudentUID	Long		306
StudentID	String	20	A0011111132
StudentType	String	50	APPLICANT
Salutation	String	15	
LastName	String	60	Busiek
FirstName	String	60	Amy
MiddleName	String	60	
Suffix	String	12	
PreferredName	String	60	
MaidenName	String	60	
EnrollmentStatus	String	40	Failure to Appear
EnrollmentStatusID	Long		16
AdmitDate	DateTime		1/1/1900
BirthDate	DateTime		1/1/1900
Programs	String	50	Information Systems Management
DateAccepted	DateTime		1/1/1900
DateEntered	DateTime		1/1/1900

Figure 43: Sample TTX file

Using TTX files allows you to develop your reports without worrying about where the database is located. A TTX file does not hold any data other than the example data in the file itself. Actual data is created in record sets that match the TTX file. All the TTX files used in CAMS Enterprise are located in the ~\Reports\ folder. They are designed to work with data produced from the ASP pages that have code similar to what was discussed earlier in this document.

You can use Crystal Reports or any text editor to see what fields are in each TTX file but you cannot just randomly use TTX files for custom reports. The TTX files are designed to work with the report that they are a data source for. What this means is that this TTX file was created specifically to work with the data that will be returned by the ASP page. Multiple reports may use the same ttx file. If you try to create a custom report from scratch and just use any TTX file you may not get the results you were expecting. Verify that the TTX file has the data fields you want to use on the custom report and also verify that the ASP page you use is returning the correct dataset.

It is recommended that you check out existing reports and see what TTX files they are using and if they have the data fields you will need for your custom report. If the report does, then copy the report and customize the new copy.

To see what TTX files are used for a report, open the report and click the **Database Expert** button.



## Type 1000 Reports

The following is a list of Type 1000 report ASP pages, Direct Connect Shells and what BYOR they are associated with.

ASP Page Used in SSI File	Associated BYOR
ADMStudentBYORType1000.asp	Admissions Student BYOR
REGAttendanceType1000.asp	Attendance Reports
Not available at this time	Financial Aid BYOR
devFundLedgerBYORType1000.asp	Development Reports Ledger BYOR
defFundConstituentType1000.asp	Development Reports Constituent BYOR
RegOfferingType1000Reports.asp	Registration Offering Print Offering
Not available at this time	Admissions Prospect Report BYOR
RegDirectoryType1000.asp	Registration Reports Directory
RegRosterType1000.asp	Registration Offering Roster Print Roster
RegScheduleType1000.asp	Registration Academic Schedule Matrix Format
RegScheduleType1000.asp	Registration Academic Schedule
Not available at this time	Registration Academic Transcripts

Direct Connect Shell	Associated BYOR
tmpRptBYORAdmStudent	Admissions Student BYOR
tmpRptBYORAttendance	Attendance Reports
tmpRptBYORFinAid	Financial Aid BYOR
tmpRptFundledger	Development Reports Ledger BYOR
tmpRptFundConstituent	Development Reports Constituent BYOR
tmpRptBYOROffering	Registration Offering Print Offering
tmpRptBYORProspect	Admissions Prospect Report BYOR
tmpRptBYORRegDirectory	Registration Reports Directory

tmpRptBYORRoster	Registration Offering Roster Print Roster
tmpRptCohortSchedules	?
tmpRptScheduleMatrix	Registration Academic Schedule Matrix Format
tmpRptSchedule	Registration Academic Schedule
tmpRptTranscriptAddresses	Registration Academic Transcripts
tmpRptTranscriptDegrees	Registration Academic Transcripts
tmpRptTranscriptSracademic	Registration Academic Transcripts
tmpRptTranscriptStudentList	Registration Academic Transcripts
tmpRptTranscriptStudentStatus	Registration Academic Transcripts

Note 1: The registration schedule reports have an additional issue to address. When adding a new custom report to the rptregScheduleReports.ssi file you must use one of three report types; 1001 if the schedule report utilizes the waiting list sub report, 1002 if the schedule report utilizes the book list sub report, and 1003 if the schedule report utilizes the Matrix schedule sub report.

# SSI Files

The following is a list of report dropdowns and their corresponding SSI file. If an SSI file in your CAMS Enterprise folder is not in this list then it should not be modified manually without guidance from Three Rivers Systems, Inc.

## Admissions

### Prospects >Reports

BYOR ..... rptAdmProspectBYORReports.ssi  
 Correspondence..... rptProsCorrRptOptions.ssi

### Students >Report

BYOR ..... rptADMBYORReports.ssi  
 Correspondence..... rptStuCorrRptOptions.ssi  
 Document Tracking Reports ..... rptADMStudentDocReport.ssi

### Organizations

Reports..... rptorgADMByor.ssi

### Contacts

Reports..... rptAdmContactsBYOR.ssi

## Registration

### Academic

Registration Save Options ..... rptregRegisterSaveOptions.ssi  
 Schedule ..... rptregScheduleReports.ssi

### Academic >Transcripts

Official Transcript ..... rptregTranscripts.ssi  
 Advisors Transcript ..... rptregTranscripts.ssi

### Offering

Print Offering ..... rptRegOfferingReports.ssi

### Offering >Roster

Print Roster ..... rptregRosterReports.ssi

### Degree Audit >Reports

Student BYOR..... rptDegreeAuditStudentBYORReports.ssi  
 Setup BYOR..... rptDegreeAuditSetupReports.ssi  
 Incomplete Courses ..... rptDegreeAuditIncompleteReports.ssi

**Attendance >Reports**

BYOR ..... rptregAttendBYOR.ssi

**Reports**

Grade Summary ..... rptregGradeSummaryReports.ssi

Directory ..... rptregDirectoryRpts.ssi

Full Time Equivalent ..... rptregFTEReports.ssi

Enrollment Verification ..... rptEnrollVerifReports.ssi

**Reports >Grade Reports**

Grade Cards ..... rptRegGradeCardsReports.ssi

Employer Reimbursement ..... rptEmployerReimbReports.ssi

**Reports >Statistics**

IPED Reports ..... rptRegIPEDRpts.ssi

By Field ..... RegIPEDFieldReports.ssi

**Reports >Hold/Degree Reports**

Hold List by Category ..... rptRegHoldList.ssi

Hold List by Name ..... rptRegHoldList.ssi

List by Degree ..... rptRegDegreeList.ssi

List by Student Name ..... rptRegDegreeList.ssi

**Billing**

**Billing Accounting**

Accounting Reports ..... rptBillingPrePost.ssi

**Billing Accounting >Post Options**

Accounts Payable ..... rptBillAPPost.ssi

**Billing Cashier Entry**

Statement Options ..... BillingStatementFormat.ssi

**Reports Menu**

Statements ..... rptBillingReportsStatements.ssi

BYOR ..... rptbillingbyorrpts.ssi

Aging Reports ..... rptBillingAgingReports.ssi

1098 Form ..... rpt1098Form.ssi

**Financial Aid**

Quick Awards ..... Hard Coded – No SSI file

**Maintenance**

Reports >Report Options ..... No SSI File.

**Report**

Award Letters ..... rptfinancialAwardLetters.ssi

BYOR ..... rptFinAidBYORReports.ssi

Document Tracking Reports ..... rptADMStudentDocReport.ssi

**Status and Load**

Enrollment ..... rptFinAidEnrollment.ssi

**Housing**

**Report**

Room BYOR ..... rptHousingRoomBYOR.ssi

Student BYOR..... rptHousingStudentBYOR.ssi

Document Tracking Reports ..... rptADMStudentDocReport.ssi

Confirmation Letters (Room Tab) ..... rptHousingRoomConfirmLtr.ssi

Confirmation Letters (Student Fee Tab) . rptHousingStudentConfirmLtr.ssi

**Health**

Reports..... rptHealthBYOR.ssi

**Parking**

**Reports**

Parking BYOR..... rptParkingBYOR.ssi

Ticket BYOR ..... rptParkingTicketBYOR.ssi

**Development**

**Reports**

Constituent BYOR..... rptFundConstituentBYOR.ssi

Ledger BYOR..... rptFundLedgerBYOR.ssi

**Reports >Correspondence**

Constituent ..... rptFundLedgerCorr.ssi

Ledger ..... rptFundLedgerCorr.ssi

**Accounting**

Accounting Reports..... rptFundPrePost.ssi

# Faculty

## Reports

BYOR ..... rptFacultyBYORReports.ssi

# Meeting

Reports..... rptMeetingBYOR.ssi

# Index

Adding A Customized Report.....	43	key.....	38
Advanced Variation with Dropdowns.....	33	Modify Pre-Existing Report.....	9
ASP Code.....	29	Modify SSI Files.....	44
ASP Pages.....	8	OnClick.....	38
Backup.....	6	Order	
Cashier Entry SSI.....	52	Of Dropdown List.....	45
Create a Custom BYOR Report.....	19	Overview.....	6
Create a Custom Report (Advanced).....	25	Pre-Existing Report Modify.....	9
Create a Custom Report (Intermediate).....	11	Remap.....	46
Create a Custom Report for BYORs (Intermediate).....	19	Remap TTX.....	48
Create Stored Procedure.....	28	Report Criteria.....	7
Create The Report.....	26	Report Does Not Work.....	48
Create TTX file.....	25	Report Items.....	46
Criteria.....	7	Report Name.....	29
Crystal Report.....	8	Reports Folder.....	6
Custom Folder.....	6	Request.QueryString() option.....	40
Custom Report.....	43	Security.....	6, 11
Custom Report Adding.....	9	Security Permissions.....	6
Custom Reports for BYORs.....	19	Server Name.....	29
Data Definition.....	47	Server Side Include.....	7
Data For The Custom Report.....	25	Setup Considerations.....	6
Database Expert.....	47	SSI	
Database Name.....	29	Admissions.....	51
delim.....	38	Billing.....	52
Direct Connect Shells list.....	49	Development.....	54
Display Custom Report.....	29	Faculty.....	54
Dropdown Order.....	45	Financial Aid.....	52
Elements.....	7	Health.....	54
Final Code Layout.....	38	Housing.....	54

Meeting .....	54	Open the Starter Report .....	19
Parking .....	54	Stored Procedure .....	30
Registration .....	51	Subreports .....	6
SSI Files.....	7, 51	TermID.....	40
Modify.....	44	trim option.....	40
Step-By-Step		TTX Files .....	47
Add Report to CAMS Report List.....	23	TTX Remap.....	48
Adding a Customized Report to a Drop Down .....	43	Type 1000 ASP pages list .....	49
Adding Data Sources.....	21	Type 1000 Reports .....	49
Create a Custom Report.....	13	uiCommon.....	38
Link Data Sources .....	22	Unknown Server.....	48
Link Other Tables or Views .....	20	User Define Routines .....	31
Modify a Report with TTX\.....	9	Verify Database.....	48
Modify a Report with Views .....	11		