Session 6-5 & 6-6

Intermediate/Advanced Crystal Reports

Presented By:
David Hardy
Progressive Reports

David.hardy@ProgressiveReports.com
WWW.PROGRESSIVEREPORTS.COM
971-223-3658

Tuesday, October 18
8:30am – 11:45am | Galleria III

Credits/Revision History

- David Hardy:
- April 3rd, 2016:
Overview

Sage CRE has incorporated Crystal Reports into almost all of its modules over the past 20 years. You find canned Crystal Reports being used in each and every module Sage 300 has. What started out mainly as a Billing invoice thing has morphed into the default tool used by Sage to produce their most complex reports, not to mention their most professional looking reports!

In this session we will cover grouping, filtering, Subreports, Formulas, and conditional formatting. We will build a Job Cost to Date report – based on JCT data table. We will select appropriate transactions by date range, group by Cost Code, add a subreport to display Group Cost Code Headers (‘divisions’), add formulas to Costs, Estimates and variances and add conditional formatting to display negative numbers as red. As time is available we will open the class up to your questions not related to the class.

Crystal Reports 2013

With the upgrade to Sage 300 14.1 version, Sage CRE has switched from using the Crystal XI R2 to using Crystal Reports 2013 version. This was a big jump in Crystal Versions. Sage leapfrogged over two other versions of Crystal to get on the latest and greatest version currently available.

There are many differences between Crystal XI and Crystal 2013. And though most of these differences are very good and will help you build better reports than ever before. It is important to know that some of your custom reports (Not canned) may not work without some modifications. For one thing, there is now an upgrade Crystal Reports feature that comes with Sage 300. You can find it by going to select All Programs> Sage> Sage300Construction and Real Estate > Reporting and Other Tools, right-click Upgrade Crystal Reports, and then select Run as administrator. Even though this is a class on canned reports it is important to know about this tool. Why is that? Because if you have already been modifying canned reports, most likely you have been naming them with a custom name so they wouldn’t get overwritten with Sage updates. Each of these formerly canned reports (Now custom) must be upgraded using this tool before using them with Sage 300 14.1 or higher. Once a report has been upgraded you will find a Formula in that report named SageCREReportVersion.
ODBC

We can’t have a discussion about modifying canned Crystal Reports for Sage 300 without first discussing ODBC. This point is often lost in the discussion and it is the foundation of a successful experience modifying custom reports.

ODBC stands for Open Database Connectivity and was introduced into the computer world many decades ago. It allows two programs to speak to each other and in some cases pass data back and forth between them. Think about it like this. A German and a Frenchman may not speak each other’s languages (similar to two propriety software packages) however if they both speak a third language such as English then the will be able to communicate with one another. ODBC is like a meeting place where diverse software such as Sage 300 and Crystal Reports that do not know anything about each other, can speak a common language.

That language is SQL or Structured Query Language. When Crystal Reports runs a report it actually sends to Sage 300 instructions written in SQL on what data it wants returned. Both the sender and the receiver can understand the request.

For us to be able to not only modify a canned Crystal Report but to preview them while in Crystal or to introduce new fields and tables we must first setup ODBC on our PC first. Therefore, when back at your PC where you have both Sage 300 and Crystal Reports installed you will need to setup ODBC so that you can do those edits and previews. If you cannot change your ODBC settings in the way you’re about to learn, then you may not have rights to do so and you will need to get your IT Dept. involved to give you those rights. You do not need to set this up each time you modify a report. You do this once for the most part and often don’t have to change it for years.

Almost everyone is now using a 64-bit machine (PC) with a 64 bit operation system such as Windows 7 or Windows 8. Sage 300 still uses a 32-bit ODBC driver. They are not alone, many other software companies including some of Sage 300 competitors still use a 32-bit ODBC driver. By default when you navigate to your computers control panel looking for the ODBC setup area you will find the 64 bit in almost every case. You won’t realize it is the 64 bit spot but it will be. Therefore you need to remember that you must navigate to the 32-bit ODBC administrator area. This is hidden a bit but you can find it.

There are two ways to find the 32-bit ODBC setup area. The first is to simple navigate to by going to Start menu, select All Programs> ODBC > 32bit ODBC Administrator to open the Data Source Administrator window. However, my favorite way is to simple type in 32 at the Windows start search area. That almost always brings up a list of files that start with 32 and the 32 ODBC Administrator will be at the top of the list.
Setting up ODBC

What I’m about to show you is also available inside Sage 300 in the User’s Guide. The User Guide is found in most modules under Help.

Activity One: Setup ODBC for use with Crystal Reports

Step 1:
From a machine that has both Sage 300 and Crystal Reports installed on it, open up the 32 bit ODBC Administrator using the following methods: From the Start menu, select All Programs > ODBC > 32bit ODBC Administrator to open the Data Source Administrator window. Or, my favorite way is to simple type in 32 at the Windows start search area.

Once open your screen should look something like below.
Step 2:

Double click on the “Timberline Data Source” or highlight it and click on the Configure button. If this is the first time you or anyone has been in the Timberline Data Source setup you will notice that no data folder has been selected.

Step 3:

Click on “Select Folder...” and pick your live data folder. During class you may pick the Timberline Sample data or whatever data folder your instructor asks you to pick.

Click OK once you have selected a data folder.
Step 4:

Click the Options Button on the right side, half way down. Make sure you have select “Standard Descriptions” under the Table and Field Naming. Most likely it will be there by default. Also make sure the “Use Maximum table segment size” is check and that “Shorten field and table names” is NOT checked. At this point your setup should look similar to the one shown below.

![Timberline Data ODBC Setup](image)

Step 5:

Click OK.

We could spend the entire class on what these settings mean. However that is not what this class is about. Therefore let me remind you that the Users Guide found in Sage 300 will go over these settings in detail. For this class let me just tell you that using “Standard Descriptions” mode instead of the other two modes is how Sage has been able to write reports for thousands of clients and have the report work for all of them regardless of what renaming each client may have done in their own copy of Sage 300. In other words it doesn’t matter what you name a field back at the office Sage will look for an internal Standard name and use that. For example if you rename “Vendor” to be “Supplier” such as they often do down in Australia, all Canned Crystal Reports will still look for the Sage internal name Vendor and ignore what you may have renamed it to.

As for the Check boxes, that is because ODBC has improved over the years and Sage wants to make sure you use all the fields on a table with their intended names. In the old days of ODBC we often had to break a table into two separate tables for reporting and truncate field names to short names. By checking the top box we don’t have to do that anymore.
What's so Magical about the “Timberline Data Source”

In almost every case you want to build Crystal Reports using the DSN (Data Source name) “Timberline Data Source”. By using this DSN, Sage 300 takes over when the report is run from the reports menu and does the following.

- Automatically switches to the data folder you are running the report from. This is critical for companies who have more than one data folder.

- Allows the use of all the Sage 300 functions such as:
  - Asking which file to use (New, Current, History)
  - Letterhead style
  - Automatically sign you in so you are not asked your User name and password.
  - And many more functions described in the Users Guide.PDF installed with Sage 300.

- Automatically sets itself back up if the DSN is ever deleted or a new version of Sage 300 is installed.

- Create Range Button when running reports from the Sage 300 reports menu.
  - Takes a couple of formulas and a parameter setup in Crystal Reports.

- You can use other DSN’s but the Magic listed above will not work!
Starting a New Report

Activity One: Create a New Report from Scratch
In this activity we will start a new report that will pull data from Job Cost.

3. Select Create New Connection.
4. Select ODBC.
5. Select Timberline Data Source.
6. Type in the Timberline User Name and Password.
   a. Ask your instructor for the User Name and Password if you are not using your own data.
7. Select Finish.
8. Select the “JCT_CURRENT__TRANSACTION” Table from the list provided.
   a. Make sure the JCT Current Table is pushed over to the right side of the screen.
9. Click OK.
10. Save the report as My Job Cost Report.

So far we have started a new report and selected the first but not the last table we will need. From here we will work forward to build each column and our rows.

Activity Two: Add fields to the Detail Section
Add the most detailed information available in the detail section. Though we will later hide this detail we will make it drilldown-able before we are done.

1. Continue working with the report from the first activity.
2. Add Category, Accounting Date, Transaction Type, Description and Amount to the detail section. *(Notice that fields placed directly into the Detail section automatically add field headings (column headings to report in the Page Header)).*
3. Type in the Timberline User Name and Password.
   a. Ask your instructor for the User Name and Password if you are not using your own data.
4. Select Print Preview from the View Menu.
5. Select File > Save.
Lining:

Very few reports have only one table. Though it does happen most of the time you’Il have multiple tables linked together. Crystal doesn’t understand the Sage 300 data therefore you need to. Linking is incredibly powerful and the more you practice linking the more fun it will become and the more creative you can become when you need to. However, linking is also the first place you can make a serious mistake. So it is important that you understand what you’re doing. Creating too few links can cause too much data and non-related data to print on your report and too many or incorrect links can cause too little or no data to be returned. I know that sounds confusing and maybe a bit scary but we I promise the more you work with it the easier it becomes.

Linking is the process of connecting related data together. For example if you want to print a list of Jobs and you want to see each jobs transactions you must link the transactions table to the job table. Another way of thinking about linking is what do they have in common? So what would link or tie transactions to jobs. In this example it would be the Job ID or Job Number.

Linking will allow us to get the data from other related tables. For example we will need the Job, Extra and Cost Code names.

Activity Three: Pull in related tables

1. From the Database Menu click on Database Expert.
2. Select Timberline ODBC on the left side.
3. From the drop down list of tables select the following tables and move them to the right box:
   a. JCM_MASTER__JOB
   b. JCM_MASTER__EXTRA
   c. JCM_MASTER__COST_CODE

4. Click on the Links Tab at the top left corner.
5. Link the Transaction table to the Job table using just the Job field.
6. Link the Transaction table to the Extra table using the Job and Extra Field.
   a. Make the link a **left outer join** between the Transaction table and the Extra table.
      i. We make this a Left Outer Join because not all jobs use Extras.

8. Click **OK** once you have linked the Tables correctly to exist Linking and Database Expert.
9. **Save** The Report.
Creating Groups

Often we use Group and Sort interchangeably but in Crystal they really are two separate things. Sort doesn’t Group while Group does Sort. So in practicality we do not use sort as often as we use Group. That is because we can group our data and turn it into eye pleasing blocks of data and in doing so we have also sorted our data in the process.

Grouped data is data that is sorted and broken up into meaningful groups. In a customer list, for example, a group might consist of all those customers living in the same Zip Code, or in the same Region. In a sales report, a group might consist of all the orders placed by the same customer, or all of the orders generated by a particular sales representative.

Most reports have at least one group but many have several, some up to a dozen or more groups. Groups give us the ability to do the following items:
- Sort as mentioned above.
- Filter our data based on Group totals.
- Drill down to see only one group worth of data.
- Subtotal our data.

Groups can be sorted in Ascending, Descending, Specified Order and Original Order. We will discuss this more later.

Activity Four: Create Grouping
1. Continue working with the report from the prior activity.
2. Using the Insert Group expert icon.
   a. You can use Insert > Group from the menu but it doesn’t work as well as the Group Expert icon.
3. We will group on Job, Extra and Cost Code.
4. Click OK
5. Print Preview your report.
a. Notice that you now have a group tree over on the right side of the screen. This tree allows you to jump to any group (Job, Extra or Cost Code).

b. Other than the groups there isn’t much too look at yet.

6. Using Field Explorer, drag the Job Description/Name to the Group Header 1.

7. Drag the Extra Description/Name to Group Header 2.

8. Drag the Cost Code Description to Group header 3.

9. Allow room for the Job, Extra, and Cost Code numbers to print to the left of the Descriptions.

10. Delete all items in the Detail section. We will shortly be adding more fields into the detail section.

11. Save your report.

Formulas:

Very few reports have no formulas, in fact not one canned Crystal Report for Sage 300 is every shipped out without at least 1 formula and most have many more than that. Why is that? Well Sage requires at least one formula on all its Accounting and Project Management reports called “TSDesignMode”. But beyond this mandatory formula we commonly use formulas to calculate things that Sage 300 is not storing for us. To understand why this is, think back a decade or two when accounting databases were becoming more common. Hard drive storage was very limited so software developers often left certain things out but made sure they could be figured out by the end users. For example, in Accounts Payable each invoice has an Original Amount and Amount Paid field but no field called “Balance” or “Open Amount”. That was not an accident but the software developers knew you could write a formula subtracting the Amount Paid from the Original Amount and come up with the Open Balance.

So formulas are often used to figure out things that are not stored in the database but also to manipulate what is there. For example, Sage 300 stores the Employee Name in Payroll this way “Last Name; First Name” but many people want to see the first name to print before the last name. We would use a formula to re-arrange the name field and split those two names out into two separate fields.

There are two major types of formulas in Crystal and they are broken down into Formula fields and SQL Expression fields. You can write far more formulas in the regular Formula fields but when possible writing SQL Expression style formulas will speed up the report and are processed on the Sage 300 server not on your local workstation. That increases speed and filtering tremendously.

In the next few activities we will write both regular formulas and sql express formulas.
Activity Five: Formulas

1. Continue using the same report from the prior activity.
2. Using the Field Explorer, right click on the Formula fields section and pick New.
3. Name this first formula “Date”. Type the following in the formula window:
   
   ```
   if {JCT_CURRENT__TRANSACTION.Accounting_Date} = cdate(0,0,0)
   then {JCT_CURRENT__TRANSACTION.Transaction_Date}
   else {JCT_CURRENT__TRANSACTION.Accounting_Date}
   ```

4. When finished click on Save and Close.
5. Using the Field Explorer, right click on the Formula fields section and pick New.
6. Create a new formula called “Original Estimate”. Type the following:
   ```
   if {JCT_CURRENT__TRANSACTION.Transaction_Type} = "Original estimate"
   then {JCT_CURRENT__TRANSACTION.Amount}
   else 0
   ```

7. Create a new formula called “Approved Estimate Changes”. Type the following:
   ```
   if {JCT_CURRENT__TRANSACTION.Transaction_Type} = "Approved est changes"
   then {JCT_CURRENT__TRANSACTION.Amount}
   else 0
   ```

8. Create a new formula called “Revised Estimate”. Type the following:
   ```
   if {JCT_CURRENT__TRANSACTION.Transaction_Type} in [ "Original estimate","Approved est changes"]
   then {JCT_CURRENT__TRANSACTION.Amount}
   else 0
   ```

9. Create a new formula called “Cost To Date”. Type the following:
   ```
   if {JCT_CURRENT__TRANSACTION.Transaction_Type} like "???cost"
   then {JCT_CURRENT__TRANSACTION.Amount}
   else 0
   ```

10. Create a new formula called “Remaining Budget”. Type the following:
    ```
    {@Revised Estimate} - {@Cost To Date}
    ```

11. Go to Design View and Place each of these formulas in the Details section in this order: Original Estimate, Approved Estimate Changes, Revised Estimate, Cost To Date and Remaining Budget.

12. Right click on each of these formulas (one at a time) in the Detail section and select Insert > Summary. Make sure you check the box “Add to all group levels”. This will put subtotals in each group footer.
13. **Save** your Report.

**Select Expert**

Select Expert in Crystal Reports can be thought of similar to “Conditions” in Sage 300. Select Experts filters the report to narrow down the number of records that are returned to just those we are interested in. There are many ways to use Select Expert. We can use the built in wizard, which is easy and helps make sure that each filter we use is optimized to return the requested data as fast as possible. However, if you go to the Reports menu and select **Select Formulas/Record** you can write the formula as you want it to be without the assistance of the wizard. That makes it more flexible and you have complete control. However, it can also make it so the filtering is not optimized for speed. We will talk about this more today in the class.

Now let’s add a filter to the report to only print certain transaction types.

**Activity Six: Filter the Report**

1. Continue using the report we saved in the prior activity.
2. From the **Report** menu option select **Selection Formulas > Record**. This brings you past the Selection Expert wizard and directly into the formula that is created when you do use the Select Expert wizard. Here you are free to write your own formulas and go beyond that which Select Expert can help with. We will write our own formula.
3. **Type** the following formula. *You can browse data so you don’t have to type the entire sentence.*

   `{JCT_CURRENT__TRANSACTION.Transaction_Type} in ["AP cost", "JC cost", "EQ cost", "PR cost", "SM cost", "IV cost", "Approved est changes", "Original estimate"]`
4. The above formula will filter the report so as to only bring backs those types of transactions that we are interested in and remove or throw out those that we don’t need. This also speeds up our report since we will not need to churn through transactions that we eventually will not use.

5. When done click on **Save and Close** at the top left of your formula screen. If there were no errors found the window will close and you have completed the filter we need for this report.
   a. If you get an error message, stop and read the message closely. Often the error does tell us exactly what is wrong. Also notice where Crystal has placed the cursor, as that is where the error begins.

6. **Print Preview** the Report. It will look something like the screen print below:

7. **Save** the Report.

<table>
<thead>
<tr>
<th>Original Estimate</th>
<th>Approved Changes</th>
<th>Revised Estimate</th>
<th>Cost To Date</th>
<th>Remaining Budget</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,895.41</td>
<td>.00</td>
<td>1,895.41</td>
<td>.00</td>
<td>1,895.41</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>273.30</td>
<td>273.30</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>273.30</td>
<td>-273.30</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>273.30</td>
<td>-273.30</td>
</tr>
<tr>
<td>151,629.17</td>
<td>.00</td>
<td>151,629.17</td>
<td>.00</td>
<td>151,629.17</td>
</tr>
<tr>
<td>-109,890.00</td>
<td>.00</td>
<td>-109,890.00</td>
<td>.00</td>
<td>-109,890.00</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>586.50</td>
<td>-586.50</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>688.50</td>
<td>-688.50</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>918.00</td>
<td>-918.00</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>1,402.50</td>
<td>-1,402.50</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>1,734.00</td>
<td>-1,734.00</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>1,657.50</td>
<td>-1,657.50</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>1,632.00</td>
<td>-1,632.00</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>1,530.00</td>
<td>-1,530.00</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>999.60</td>
<td>-999.60</td>
</tr>
<tr>
<td>13,500.00</td>
<td>.00</td>
<td>13,500.00</td>
<td>.00</td>
<td>13,500.00</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>117.79</td>
<td>-117.79</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>138.29</td>
<td>-138.29</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>212.51</td>
<td>-212.51</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>321.43</td>
<td>-321.43</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>372.70</td>
<td>-372.70</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>352.09</td>
<td>-352.09</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>346.67</td>
<td>-346.67</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>325.02</td>
<td>-325.02</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>212.34</td>
<td>-212.34</td>
</tr>
<tr>
<td>57,134.58</td>
<td>.00</td>
<td>57,134.58</td>
<td>14,367.34</td>
<td>42,767.24</td>
</tr>
<tr>
<td>5,050.00</td>
<td>.00</td>
<td>5,050.00</td>
<td>.00</td>
<td>5,050.00</td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td>4,895.00</td>
<td>-4,895.00</td>
</tr>
<tr>
<td>5,050.00</td>
<td>.00</td>
<td>5,050.00</td>
<td>.00</td>
<td>155.00</td>
</tr>
</tbody>
</table>
Activity Seven: Create Group Cost Code Formula

In this activity we will be creating a SQL Expression Formula that creates a Group Cost Code ID. Group Cost Codes are rarely found in the transaction file. But we want to sort and group our data by this cost code so we will make it up using SQL Expressions.

1. Continue working with the report from the first activity.
2. Using Field Explorer go to the SQL Expressions section.

3. Right click SQL Expressions and pick New.
5. Then either type what is below or use the string functions and field names similar to what the instructor has shown.

\{fn SUBSTRING("JCT_CURRENT__TRANSACTION"."Cost_Code",1,{fn LOCATE('-','JCT_CURRENT__TRANSACTION"."Cost_Code")-1})+'-000'\}

6. Once typed click on Save and Close.
7. From the Group Expert or the Menu Insert Group we want to add a new Group based on the SQL Expression we just created.
   a. Create a new Group based on the SQL Expression “Group Cost Code”. Make this your third group right after Job and Extra.
8. Print Preview your report.
Subreports

Subreports are one of the main features that separate Crystal Reports from so many other reporting tools including the Report Designer in Sage 300. Subreports allow us to pull data from other modules within Sage 300 or even from sources outside of Sage 300 such as Excel, Access or practically any other database and source. Think about a report that pulls in content from an Estimating program or a database stored on a website that has no affiliation with Sage 300. Now that may be an extreme situation but it is used every day. For a more regularly used solution we use Subreports in Sage 300 to accomplish the following: Recap data in a different grouping than the original report (known as the Main Report or Container Report) Perhaps you have sorted your original report in Cost Code Order but at the end of the report you want to see a recap of all Costs and Estimates in Category order. That can’t easily be done in the main report because you can only have subtotals where you have grouped your data. However, a subreport allows us to run through the same data a second time but group it differently than the main report; in this case in Category order. Often we need to pull data from another module that may not directly link up to the tables in the main report. One example is a Job Cost report that needs to pull data from General Ledger or Property Management or even Contracts. Subreports allow us to link the Main Report and Subreport together using formulas whereas all linking in the Main Report must be done in the Visual Linking expert and it can only entail one field equaling another.

Last Subreports can be on-demand. In other words they don’t run when you launch the report but rather, they only run if someone clicks on them. These are often visible as blue words such as you see on a web page. If you click it will run!

So you may have noticed that we did not put a group header Name/Description in Group 3 yet. As you will recall we created a SQL expression to slice the first two digits of the cost code. Now we need to description for that major grouping...i.e. (Site work, General Conditions etc.).

We will build a subreport to pull just the major grouping names into our report. We can either use the Standard Cost Code List or the Cost Codes related to each job.

Activity Eight: Subreport

1. While keeping your existing report open, go ahead and click on File > New > Blank Report to start a new report.
2. Pick ODBC as your connection type and Timberline Data Source as your DSN.
3. Pick the JCM_MASTER__COST_CODE table from the list and move it over to the right.
4. Click OK.
5. Select the Description field from the Cost Code Table and place that in the report header at the top left corner.
6. Save this report with the name Group Cost Code Subreport.
7. Then close the Group Cost Code Subreport report.
8. You should now have open just your existing Job Cost report that we have been building for the past several activities.
9. Using the Insert Menu select **Subreport**.
10. When prompted, select **choose an Existing Report** and browse for the report Subreport you previously saved.
11. Once selected, go ahead and click on the **Link** tab at the top left of the Insert Subreport dialog box.
12. This is important! Using the Transaction table in the **Top Left Box**, **double click** on Job moving it the **top right** and link it to the **job** in the subreport at **bottom right**. Do the same thing for **Extra**.
13. Now in the top left box find your **SQL Expression “Group Cost Code”** and **double click** on it. **Link** it to the **Cost Code** in the bottom right. It may try to link to the Cost Code Description, but you can click on the drop down and force it to link to the Cost Code field instead.
14. Once you’ve linked the subreport click **OK**, it will be attached to your mouse. You want to **place** the subreport in **Group Header 3**, just to the right of the Group name field.
15. After doing this **remove** the border around the subreport. You can do this with the **Border** button or you can right click on the subreport and chose format subreport.
16. **Double click** into the subreport and **suppress** all sections other than the **Report header**.
17. **Preview** the report it should look something like this.

18. **Copy** the all the fields in Group Footer 4 to Group Footer 3. They should line up directly below the same fields in Group Footer 4.
19. **Save** your work.

So why did we need a Subreport? Because there are generally no transactions in the current file aimed directly at the group cost codes, those that end in the triple zero, it was best to use a subreport that could be linked to using a formula or SQL Expression. The beauty of Subreports is that we can link on fields, formulas and parameters and as you all know we can’t do that with regular report linking.

**Parameters**

Many of you are familiar with Parameters and you don’t even know it. In Sage 300 they are called Prompts or Prompt Fields. Parameters in Crystal work the same way. They can be used for several things in Crystal Reports. Most often they are thought of as a way to allow the user to filter the report the way they want to. For example, we can create a Parameter that asks what Job ID we want to run the report for. That is filtering the report for only that Job. Or we can ask for a cutoff date or date range. Again, those parameters are simply allowing the person running the report to filter the report for a date range or a cutoff date.

There are other uses for Parameters though. One of the more common uses would be to change the level of detail that prints or the order in which the report prints. For example, we could create a Parameter that asks if the person running the report would like to see a summary (one line per job) or a semi-summary report (one line per cost code) or detail report where they see all the transactions that make up the entire report. That is one report with one Parameter (Prompt) that asks what level of detail and each time a person runs it they could choose differently. Pretty cool huh!

There is still another way they can be used and that is to interact with the data. We could have a Parameter that asks what markup percent you want. Based on the value typed in at run time the markup to the cost entries would be different each time. That makes a very interactive and dynamic report.

There are many other uses for Parameters but these are some of the most common uses. Remember they are essentially no different than those you may have been using for years in Sage 300 Report Designer; with the exception that they are more powerful and generally easier to use.

In fact they are so powerful that there are several features within Parameters that will not work once the report is placed on the Sage 300 menu. One of those features is **Dynamic Values** and another is **Allow Multiple Values**, which we will discuss more in class.

If class time permits we will discuss ways of running Crystal Reports without placing them on the Sage 300 menu. One reason we would do that is to take advantage of features in Crystal that are
turned off when run from the Sage 300 program. In addition to Dynamic Parameters, Alerts is a valuable feature not allowed when run from within Sage 300.

We want to create a new parameter to be used as a cutoff date. This date will then be used to look at each transaction to determine if its date is on or before this date or after the cutoff date. If the date is after the cutoff date we will exclude it from the report.

Activity Nine: Parameter

1. Continue using the same report from the prior activity.
2. Using the Field Explorer, right click on the Parameter Fields section and pick New.
3. Give the Parameter a name of “Cutoff Date”.
4. Change the default type to Date.
5. Click OK as you are done setting up this parameter.

Though we are done setting up the parameter we want to filter the report to only pull entries that are on or before the cutoff date. To do this we will engage Select Expert again.
6. Go to the Reports Menu and select Selection Formulas > Records. Type the portion of the formula below that is in the box. Be sure to put an “and” between the first half and the second half.
7. Click on **Save and Close** when done.

8. **Save** your report.

What the formula above accomplishes for us is to filter out costs and estimates that happened after the cutoff date while filtering out Estimates and Contract values that happened after the cutoff date. Sage 300 doesn’t use Accounting dates for things like Estimates and Contracts. Basically anything that doesn’t have an effect on General Ledger will not have an Accounting Date.

<table>
<thead>
<tr>
<th>Original Estimate</th>
<th>Approved Changes</th>
<th>Revised Estimate</th>
<th>Cost To Date</th>
<th>Remaining Budget</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-000 GENERAL CONDITIONS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1-040</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>151,029.17</td>
<td>0.00</td>
<td>151,029.17</td>
<td>0.00</td>
<td>151,029.17</td>
</tr>
<tr>
<td>103,850.00</td>
<td>0.00</td>
<td>103,850.00</td>
<td>0.00</td>
<td>103,850.00</td>
</tr>
<tr>
<td>103,850.00</td>
<td>0.00</td>
<td>103,850.00</td>
<td>0.00</td>
<td>103,850.00</td>
</tr>
<tr>
<td>13,500.00</td>
<td>0.00</td>
<td>13,500.00</td>
<td>0.00</td>
<td>13,500.00</td>
</tr>
<tr>
<td>13,500.00</td>
<td>0.00</td>
<td>13,500.00</td>
<td>0.00</td>
<td>13,500.00</td>
</tr>
<tr>
<td>1-045</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5,050.00</td>
<td>0.00</td>
<td>5,050.00</td>
<td>0.00</td>
<td>5,050.00</td>
</tr>
<tr>
<td>5,050.00</td>
<td>0.00</td>
<td>5,050.00</td>
<td>0.00</td>
<td>5,050.00</td>
</tr>
<tr>
<td>1-065</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20,239.00</td>
<td>0.00</td>
<td>20,239.00</td>
<td>0.00</td>
<td>20,239.00</td>
</tr>
<tr>
<td>20,239.00</td>
<td>0.00</td>
<td>20,239.00</td>
<td>0.00</td>
<td>20,239.00</td>
</tr>
<tr>
<td>20,239.00</td>
<td>0.00</td>
<td>20,239.00</td>
<td>0.00</td>
<td>20,239.00</td>
</tr>
<tr>
<td>1-510</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2,185.96</td>
<td>0.00</td>
<td>2,185.96</td>
<td>0.00</td>
<td>2,185.96</td>
</tr>
<tr>
<td>27,701.47</td>
<td>0.00</td>
<td>27,701.47</td>
<td>0.00</td>
<td>27,701.47</td>
</tr>
<tr>
<td>16,000.00</td>
<td>0.00</td>
<td>16,000.00</td>
<td>0.00</td>
<td>16,000.00</td>
</tr>
</tbody>
</table>
14. From the Design Mode right click on the word Details and select Suppress (No Drill-down).

15. Right click on the Remaining Budget formula in Group Footer 4 and select Format Field. On the Font tab select the X+2 formula to the right of color. When the formula editor opens up, write the following formula. You will have to remove the writing that is already there first. That writing is automatic by Crystal to give you ideas and help.

```
if currentfieldvalue < 0 then red else black
```

This conditional formatting formula says that if the value of the field I have right clicked on is less than zero then turn the field red otherwise stay the normal black. You can use this logic on any field displayed on the report.
Your report will look something like mine above hopefully! There are a couple things we can do to improve both the cosmetic looks of the report. We can add Better-looking column headers and add drop shadows and other features that will enhance the look of the report. We will get to the cute factor if class time is available but I want you to continue to see more of the formatting formula features and on-demand Subreports first.

Field Formatting

Field formatting gives us the ability to not only change the font size and color of a field or to change the direction in which fields print but we can do almost anything sometimes. The power to do things sometimes is another feature that takes Crystal Reports far beyond many other report writing programs including Sage 300 Report Designer.

To format a field or object (such as a line box or image) we simply need to right click on the field and select Format Field.
Activity Ten: Field and Section Formatting with Formulas

1. Continue using the same report from the prior activity.
2. Create a new Parameter. This one called “Detail Level”.
   a. The Parameter Type will be String.
   b. And it will have two default values. (Group Cost Code and Cost Code).
3. Make sure to type in both Values in the middle left box.
4. Make sure to set the “Allow custom values” to False.
5. Click on OK when done.
6. Copy all the numeric fields in Group Footer 3 and paste them direct above where they are in Group Header 3.
7. You may have to shorten the subreport in Group Header 3 to make all the numbers line up where they should.
8. Highlight or lasso all the numeric fields in Group Header 3 and then right click on one of them. Go to the Common tab.
9. On the Common tab click on the Suppress conditional Formatting formula icon. In that formula write the following formula.
   
   {?Detail Level} <> "Group Cost Code"
10. Click on Save and Close and then on OK at the bottom of the screen.
11. Move the Group 4 Header field to the Group Footer 4 section.
12. Right click on Group Header 4 and select Suppress (No Drill-down).
13. Right click on the Word Group Footer 4 and select Section Expert.
14. In the Suppress formatting formula type this formula.
a. \{?Detail Level\} = "Group Cost Code"

15. **Right click** on the Word **Group Footer 3** and select **Section Expert**.
16. In the Suppress formatting formula **type** this formula.
   a. \{?Detail Level\} = "Group Cost Code"

17. **Preview** the Report and **pick** Group Cost Code as your Level of Detail.
18. **Save** the report when done.

So you can see that based on the Parameter answer we give we can show and hide sections and fields to better format our report. This makes it much easier to build one report that can be used by many different people in the company and for different uses. When you build one report instead of several you are not only being more efficient but later on when you have to modify a report you don’t have to modify 3 different version of the same report. That is always a recipe for disaster!

### Final Formatting and cleanup

The report still doesn’t look good but let’s change that a bit. In this next exercise we will add Job Names. We will add Cost Code names and Extra names. We will also add a Title and subtotal descriptions. I jokingly often say during classes that we can make a report look so good that the numbers don’t have to be right. In reality we can do both!

**Activity Eleven: Cleanup**

19. Insert a text Object in the Report Header. Inside this text Object type a report title name such as **Job Cost Budget Report**.
20. **Increase** the size of the font for your new Report Title and Center the title on the page.
21. Using the **Field Explorer**, pull the **Job Description** into **Group Header 1**, just to the right of the Job ID.
22. Using the **Field Explorer**, pull the **Extra Description** into **Group Header 2**, just to the right of the Extra ID.
23. Using the **Field Explorer**, pull the **Cost Code Description** into **Group Footer 4**, just to the right of the Cost Code ID.
24. **Bold** all three of these **description** fields you have just brought in.
25. **Print Preview** your report. It should look something like the printout below:
26. Add an empty text object to each of these sections, Report footer, Group 1 Footer, Group 2 Footer. And Group 3 Footer. Each just to the left of the Original Estimate number.

27. In each of these four text objects type the word “Total:”, then right justify the text object.

28. Using the Field Explorer drag the group name 1 into the text object for group 1. Do the same for group 2 & 3. Bold each of those text objects.

29. Stretch your columns out across the page as needed. Add bold and color to your column headings and title to give it a bit of flair.
Cross Tabs
Cross-tabs are an under used feature within Crystal Reports. Very few people even know they exist. However, they are a very power tool for summarizing your data similar to a Pivot table in Excel.

For our example let’s say that we want to recap just our costs at the bottom of the report showing one line per job with a cost total broken out by Cost Category. As you recall we did not group our report by category therefore normally we would not be able to summarize our data in a way that did not coincide with our grouping. But Cross tabs allow this to happen.

Activity Twelve: Cross tab
1. From Design view Right click on the word Report Footer section.
2. Pick insert new section below.
   a. Now you have a Report Footer a and b.
3. From the Insert menu pick Cross Tab.
4. Place it in Report Footer b to the left side.
5. Right click on the Cross tab and pick Cross Tab expert
6. You’ll notice it is basically a wizard.
7. Drag the Job into the bottom left box
8. Drag the Category into the top right box
9. Drag the Cost To Date formula into the right bottom box.
10. Print Preview the report and go to the last page. You’ll now see a recap of your data by job and by category.

You can have more than one cross tab and they can be in your Report Header and Footers as well as Group Header and Footers. Play with them you will get addicted to them.
Summary

I hope you’ve had a good time in this class and found the information beneficial. As with any other software package if you do not use it regularly you will quickly forget the basics and even easy steps will become hard again. For the best results make it a habit to practice working with Crystal Reports each day for several months. If you will do that, you will master all these concepts and even those things that may seem hard to you now such as linking will become second nature. And at that point you can take a week off here or there and not fear that you have lost valuable knowledge and skills.

We’ve just scratched the surface of what Crystal Reports can do. Progressive Reports is committed to bringing you more classes in the future. We have classes for building Financial Statements in Crystal and how to build reports using SQL commands in an advanced class each year at TUG and of course all year long from Progressive Reports.

Please do not hesitate to reach out to me if you have any questions or need further training. Of course I hope each of you will take what you’ve learned and move forward. The skills you have acquired will be valuable for the remainder of your career.