R&R Report Writer® *Technical Bulletin*

Conditional Counts

Product:	R&R Report Writer®	**Oper Sys:**	DOS®
Version:	All		

Summary

We've received many calls from R&R Report Writer users asking if there is a way to count records that contain a particular value in one of the fields.

To create a conditional count, you will need to:

1. Create a calculated field for each value you want to count.

2. Create one total field for each value you want to count.

3. Insert the counts in your report.

For example, you may want to know how many orders are for dBR and how many are for R&R Report Writer. Using the PRODUCT field, how can you produce a separate count for each product?

If you are grouping on PRODUCT, you can produce counts by simply creating a group subcount. If you aren't grouping on PRODUCT the answer is not as obvious, but it is relatively easy. In this example we show how to create a grand count, but the same technique can be used to produce a subcount or page count.

Creating Calculated Fields

First, create one calculated field for each value you want to count. You enter an expression that causes the field to contain a 1 if the record contains the desired value and a 0 otherwise. Using the /Field Calculate Create command, enter the following calculated field expressions.

```
dBR = IIF(DETAIL->PRODUCT="dBR", 1, 0)
RnR = IIF(DETAIL->PRODUCT="R&R", 1, 0)
```

If you are using R&R Report Writer version 2, you can simplify the calculations by creating a logical instead of a numeric field. This is because R&R Report Writer version 2 treats logical true as a numeric 1, and logical false as a numeric 0. Use these simplified expressions instead.

```
dBR = DETAIL->PRODUCT="dBR"
```

RnR = DETAIL->PRODUCT="R&R"

Create One Total Field for each Value

The next step is to create one total field for each value you want to count. You create a sum total to add up all of the ones, which is the same as counting records with the desired value. Using the /Field Total Create command, enter the following total field definitions.

dBR_Sum = Grand Sum of dBR
RnR_Sum = Grand Sum of R&R

Insert the Counts in your Report

The final step is to insert the counts in your report. Use the /Field Insert command to place dBR_Sum and RnR_Sum in the summary area. The calculated fields dBR and RnR do not appear on the report. They are only used as memory variables to record whether or not the PRODUCT field contains a particular value.

Conditional Counts in Version 3

Creating conditional counts involves creating a calculated field and a total field for each count. With the advent of self-referencing and user-defined functions in R&R Version 3, it is possible to create a conditional count using a single calculated field.

First let's review what is meant by "conditional count" using an example. Suppose a report lists all of your customers grouped by the state in which they're located. Using a total field you could easily count the number of customers per state. But suppose you want a separate count of customers who are resellers and those who are end-users. See the sample report in Figure 1.

```
              Conditional Customer Count by State

     State    Name                  Customer Type

     CA       Jughead Software    Reseller
              Sandy Eggo             End-user
              Beverly Hill           End-user
              Louis Obispo           End-user
              SoftSurf Systems        Reseller

         Number of End-users:        3
         Number of Resellers:        2
                             ---
             Total Customers:        5

     NY       Chester West           End-user
              SNAFU Consultants   Reseller
              Sarah Cuse             End-User

         Number of End-users:        2
         Number of Resellers:        1
                             ---
             Total Customers:        3
```

Figure 1. Conditional Count Report

This application requires two conditional counts, one that counts customers on the condition they are a reseller, and one that counts customers on the condition they are an end-user. In such an application, the database structure would be similar to the following simplified structure.

```
.display structure
Structure for database: \RR\CUSTOMER.DBF
Number of data records:      8
Date of last update   : 07/23/88
Field  Field Name  Type       Width    Dec
   1  NAME        Character    20
   2  STATE       Character     2
   3  TYPE        Character     1
** Total **                   24
```

The fields are self-explanatory except for TYPE which contains the letter "R" for resellers and "E" for end-users. In this example, the database contains the following records.

```
.list
Record#    NAME                  STATE      TYPE
     1     Jughead Software      CA         R
     2     Sandy Eggo            CA         E
     3     Chester West          NY         E
     4     Sarah Cuse            NY         E
     5     Louis Obispo          CA         E
     6     SoftSurf Systems      CA         R
     7     SNAFU Consultants     NY         R
     8     Beverly Hill          CA         E
```

Let's build the conditional count user-defined formula (UDF) in two steps. First we'll create one that works for grand counts, and then we'll enhance the formula to work for subcounts. As you may have guessed, the formula for subcounts is a bit more complex.

Enter the conditional grand count UDF declaration and formula using the /Field Calculate User-Function Create command.

```
Declaration:          CGCount(l_condition,n_count)
Formula:              IIF(condition,count+1,count)
```

This formula is pretty simple. It says that if the condition is true, then add one to the count. To use the CGCount() function in our example report, create two calculated fields named R_Count (reseller count) and E_Count (end-user count) as follows.

```
R_Count = CGCount(TYPE="R",R_Count)
E_Count = CGCount(TYPE="E",E_Count)
```

Note the use of "self-referencing" in these expressions. Self referencing occurs whenever the name of the field being defined appears in the expression. Self-referencing is used so that for each record, the new value of E_Count equals the current value of E_Count, plus one if the condition is true.

As it now stands, CGCount() will calculate a grand count. To modify the UDF formula to calculate a subcount, you need to pass another argument to the function— the group field. Using the PREVIOUS() function, you can reset the group count whenever the group field changes.

Using the /Field Calculate User-Function Create command, enter the conditional subcount UDF declaration and formula. A shortcut is to use the Choices Menu [F10] to copy the CGCount() user-function formula. Then modify the formula as follows.

```
Declaration:        CSCount(c_group,l_condition,n_count)
   Formula:
         IIF(group=PREV(group),
           IIF(condition,count+1,count),
             IIF(condition,1,0))
```

The conditional subcount expression works by comparing the contents of the group field in the current record with that of the previous record. If they are not equal a group break has occurred. There are four possible cases and results as shown in the following table.

Group Break?	Condition True?	Count=
No	Yes	Count+1
No	No	Count
Yes	Yes	1
Yes	No	0

In the subcount UDF it is important to note that the group field in this example is character data type. You may need to modify the formula to work with a group field of a different data type. An alternative is to create a calculated field that converts a numeric or date group field value to character data type and use it in place of the group field.