

Crystal Reports Technical Document

Subject: Using variables from VB in CR Formula Properties.
Date: Sept 3, 1993
Versions: 2.0(Pro), VB

Solution:

The secret to this is to remember that formulas have to conform to Crystal Reports formula standards. (ie strings need to be embedded in quotes whereas numbers do not)

For example, a variable string in VB is often assigned like so:

```
charstr As String  
charstr = "This is a string"
```

VB will recognize that this is a string. However, if passed to Crystal Reports the string will not be recognized. For CR to recognize a VB string for what it is you need to embed quotes into the string.

In VB, you do that by using 2 double quotes or 1 single quote. Therefore the following statements, have embedded double quotes in the strings.

```
charstr = """"This is a string"""  
or  
charstr = "'This is a string'"
```

Now you can concatenate this string to the Formulas property like so:

```
report1.Formulas(0) = "formulaname=" +charstr
```

This has now set @formulaname to be equal to the string and when the report is printed out it will print out that string where ever the formula was placed.

For numbers you do not have to embed quotes around it. Therefore, if you have the following statements:

```
num = 1234
```

```
report1.Formulas(0) = "formulaname=" +num,
```

the number 1234 will be printed out, but as a Number field and not as a text field.

Dates require a little more work. To compare a date field to a date IN CR you would use a formula like this

```
{datefield} = Date(1993,08,24)
```

To set up this formula from VB you will have to do the following:

- 1) Break down the date input by the user or given by the computer into its component parts of Year, Month, Day.

{file name}

2) Set up the formula like so:

```
report1.Formulas(0) = "formulaname= {datefield} = Date("+Year+", "+Month+", "+Day+")"
```

where Year is the year like 1993, Month is the numeric representation of the month, and Day is the day like the 24th or the 1st.

{file name}