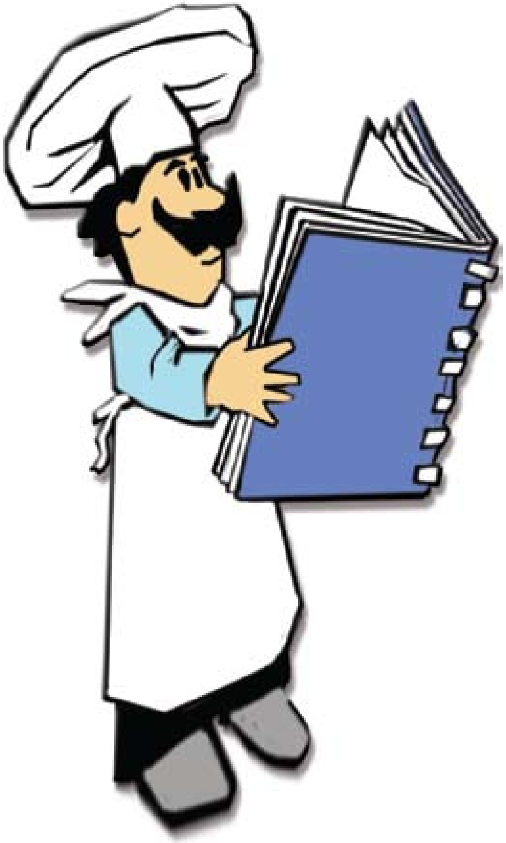
Recipes for Successful Report Writing

iReport is an easy-to-use open source report writer that allows BBj® developers to create custom reports in a quick and efficient manner. Built using the Netbeans framework, the iReport interface is one with which most of you will be familiar and can quickly and easily create custom reports. You can also achieve a more sophisticated look or capability that is not obvious with very little extra effort. This article shares recipes for several eye-pleasing enhancements that you can apply to your reports. Read more about the BBj-specific iReport integration points supplied by BASIS in *Jazz up Your Applications* on page 16 of this issue.

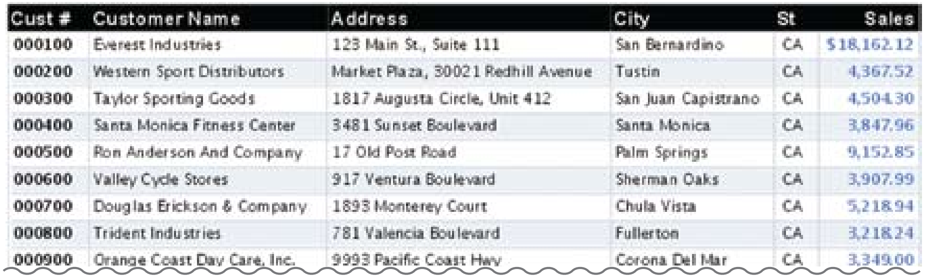


Alternating Row Colors

To easily achieve alternating row colors using the rectangle report element, create a rectangle around a row in the iReport Designer and size it so that it covers the entire row. Next, change the rectangle to the desired alternating row color and check the "Opaque" box in the rectangle properties. At this point every row will use the colored background. To tell iReport that it should only use the background for alternating rows, modify the "Print when Expression" property, a Java expression that must resolve to either True or False to determine whether to include the component in the generated report. To show the background on the even rows, use the variable **$V{PAGE_COUNT}** that provides the current number of records that have been processed in the current page. In other words, it gives the row number, in real time, as the report is created. Next, create the following expression to determine if the current row is even, resulting in a Java Boolean that iReport uses to determine whether or not to draw the background color:

```
new Boolean( $V{PAGE_COUNT}.intValue() % 2 == 0)
```

The conditional expression causes even rows to display the colored rectangle as shown in **Figure 1**.



Cust #	Customer Name	Address	City	St	Sales
000100	Everest Industries	123 Main St., Suite 111	San Bernardino	CA	$18,162.12
000200	Western Sport Distributors	Market Plaza, 30021 Redhill Avenue	Tustin	CA	4,367.52
000300	Taylor Sporting Goods	1817 Augusta Circle, Unit 412	San Juan Capistrano	CA	4,504.30
000400	Santa Monica Fitness Center	3481 Sunset Boulevard	Santa Monica	CA	3,847.96
000500	Ron Anderson And Company	17 Old Post Road	Palm Springs	CA	9,152.85
000600	Valley Cycle Stores	917 Ventura Boulevard	Sherman Oaks	CA	3,907.99
000700	Douglas Erickson & Company	1893 Monterey Court	Chula Vista	CA	5,218.94
000800	Trident Industries	781 Valencia Boulevard	Fullerton	CA	3,218.24
000900	Orange Coast Day Care, Inc.	9993 Pacific Coast Hwy	Corona Del Mar	CA	3,349.00

Figure 1. Example of even rows displaying the colored background

Hyperlinks

Today, the need to include hyperlink components in reports is on the rise. These hyperlinks can be one of three distinct types in iReport and BBJasper, each with a different purpose – the Reference type, Command type, and ReportName type. All of these types are specified in the hyperlink properties of a report element. To access the Hyperlink Properties dialog box, right click the report element inside the iReport designer and choose "Hyperlink" from the popup menu.

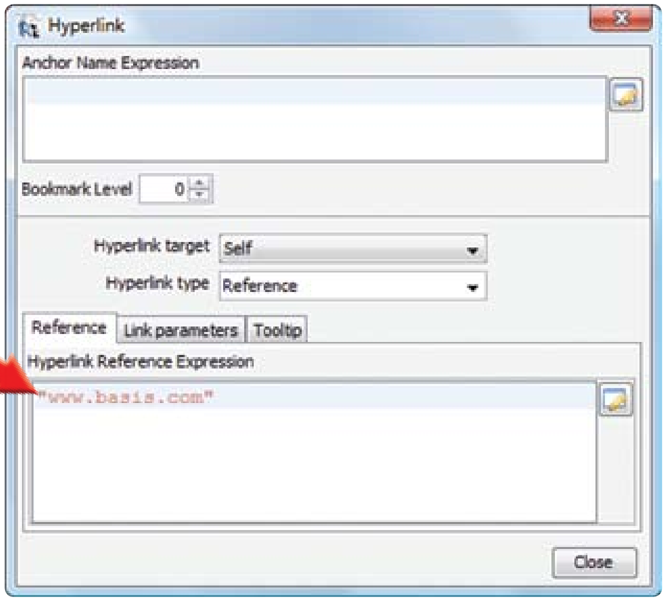
Reference Hyperlink

The Reference type is the kind of link that most users associate with the word hyperlink. When the report displays via the BBJasper utility, clicking the link causes the BBInvoker utility to open the system's default Web browser and display the target Web page that is determined by the URL specified in the hyperlink definition.

To create a Reference hyperlink, enter the URL in the Hyperlink Reference Expression box of the Reference tab. The value is a string and must be surrounded by quotes as shown in **Figure 2.**

>>

By Robert Del Prete
Quality Assurance Engineer



Anchor Name Expression

Bookmark Level 0

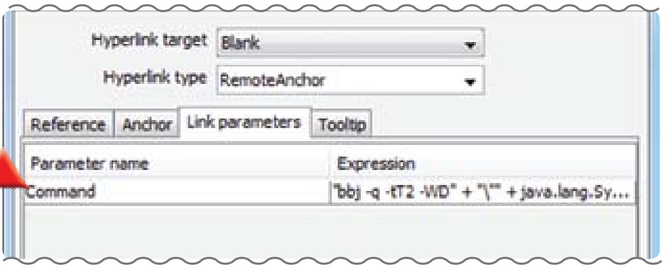
Hyperlink target Self

Hyperlink type Reference

Hyperlink Reference Expression

"www.basis.com"

Figure 2. Reference hyperlink example

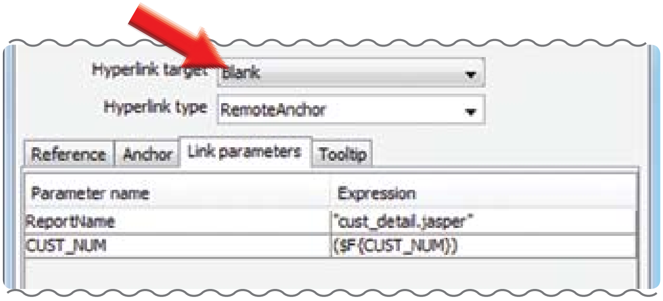


Hyperlink target Blank

Hyperlink type RemoteAnchor

Parameter name	Expression
Command	"bbj -q -tT2 -WD" + "\"" + java.lang.Sy...

Figure 3. Command hyperlink example



Hyperlink target Blank

Hyperlink type RemoteAnchor

Parameter name	Expression
ReportName	"cust_detail.jasper"
CUST_NUM	($F{CUST_NUM})

Figure 4. ReportName hyperlink example

Command Hyperlink

The Command hyperlink type takes advantage of BBj's SCALL capability to run another program or BBj application. The Command parameter passes the expression given to the command shell. The expression is a string and also requires quotation marks. **Figure 3** shows the parameter name Command and the following expression that launches the Digital Dashboard demonstration program to detail sales figures for the desired month:

```
"bbj -q -tT2 -WD" + "\"" + java.lang.System.getProperty("basis.BBjHome") +
"/demos/digitaldashboard" + "\"" + " digitaldashboard.src - " +
$P{ORDER_MONTH} + "&"
```

ReportName Hyperlink

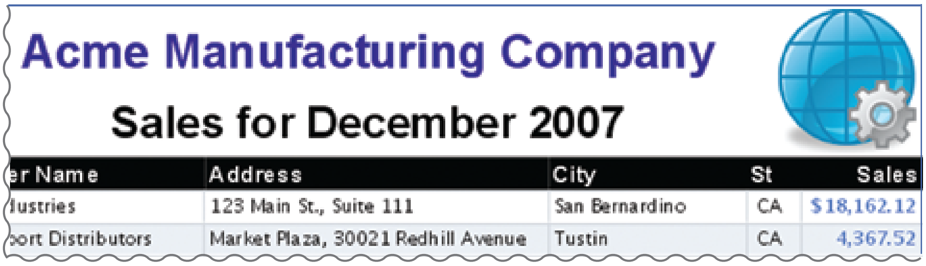
The ReportName hyperlink instructs BBJasper to launch another Jasper report. Use the ReportName parameter to specify the path and name of the new report, as well as any additional parameters the report requires. The report author may also control the resultant window's sizing, placement, and title via additional parameters. If the Hyperlink target is set to Self, the new report will replace the current report in the viewer window. If the Hyperlink target is set to Blank (see **Figure 4**), the report appears in an additional viewer window.

Expressions

Expressions can be used in many situations when designing a report. For example, it is possible to display a readable month name when the input is a numeric value. An additional field in the database can take care of this but why bother when iReport has the capability of utilizing programmable expressions? To change the parameter $P{MONTH} into a readable month value, use the expression below. >>

```
"Sales for " + ( $P{MONTH}.equals("1") ? "January" :
( $P{MONTH}.equals("2") ? "February" :
( $P{MONTH}.equals("3") ? "March" :
( $P{MONTH}.equals("4") ? "April" :
( $P{MONTH}.equals("5") ? "May" :
( $P{MONTH}.equals("6") ? "June" :
( $P{MONTH}.equals("7") ? "July" :
( $P{MONTH}.equals("8") ? "August" :
( $P{MONTH}.equals("9") ? "September" :
( $P{MONTH}.equals("10") ? "October" :
( $P{MONTH}.equals("11") ? "November" :
( $P{MONTH}.equals("12") ? "December" : "" )))))))))))) + " " +$P{YEAR}
```

This expression converts the **“12”** in **${P_MONTH}** to the string December, provides the year from the $P{YEAR} parameter, and adds additional text to create the report's title as shown in **Figure 6.**



Acme Manufacturing Company

Sales for December 2007

er Name	Address	City	St	Sales
lustries	123 Main St., Suite 111	San Bernardino	CA	$18,162.12
ɔort Distributors	Market Plaza, 30021 Redhill Avenue	Tustin	CA	4,367.52

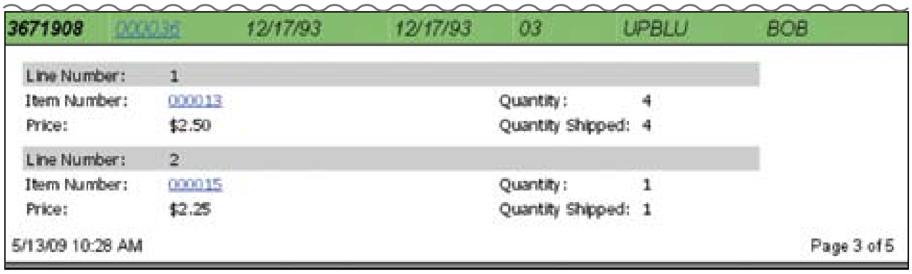
Figure 6. Example of month and year

Evaluation Time

Report Elements in iReport have an “Evaluation Time” text field property. This property is particularly useful for parameters or variables that need evaluation at different times during report creation. Useful values for this field include:

Now	Evaluates the expression immediately
Report	Evaluates the expression at the end of the report
Page	Evaluates the expression at page end
Column	Evaluates the expression at column end

When creating a new report, iReport automatically includes the $V{PAGE_NUMBER} variable by default. With the Evaluation Time set to **Report**, it evaluates the variable once the report is complete and gives the total number of report pages. However, with the Evaluation Time set to **Now**, it displays the page number that the report is currently creating. Therefore, the same variable can be used to create a report footer that shows the current page and total number of pages for the report as shown in **Figure 7,** just by modifying the Evaluation Time property.



3671908	000036	12/17/93	12/17/93	03	UPBLU	BOB

Line Number:	1		
Item Number:	000013	Quantity:	4
Price:	$2.50	Quantity Shipped:	4
Line Number:	2		
Item Number:	000015	Quantity:	1
Price:	$2.25	Quantity Shipped:	1

5/13/09 10:28 AM

Page 3 of 5

Figure 7. Current and total page count example

While the page number example serves as a good illustration for the use of the Evaluation Time property, it is so common that the creators of iReport made this property even easier to access via the Tools section of the component palette. Expanding the Tools section reveals new options that provide a shortcut to some of the more popular but previously obscure features such as adding the page number, total pages, or the 'Page X of Y' as shown in **Figure 8**. Dragging and dropping this option into the report does everything covered previously in one easy step, as iReport creates the two text fields referencing the page variable and sets the evaluation time appropriately for each field.

Calculation Property of Variables

Calculating values within report fields using the Calculation property of a parameter is often times much easier than creating a subreport to do the same thing via SQL. An additional parameter in the report easily accomplishes this. In the properties of the added parameter, change the Variable Class to **java.math.BigDecimal** and enter the >>

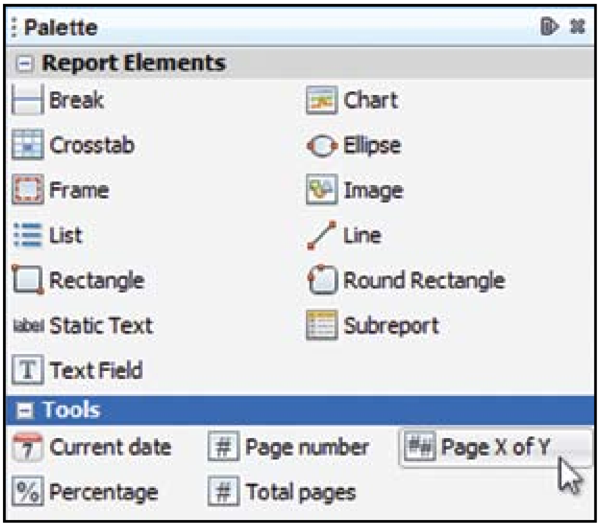
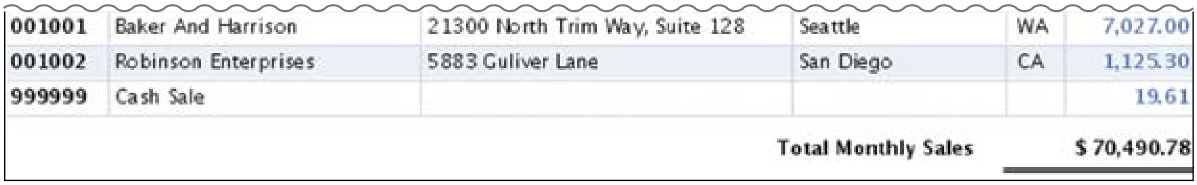


Figure 8. Page X of Y element from the Tools section of the Component Palette

calculation to perform in the Variable Expression property. The Calculation property contains several options including:

Nothing	No calculation is performed
Count	Returns the number of times the result is different from nu
Distinct Count	Returns the number of unique results returned
Sum	Adds each expression value to the current value
Average	Returns the average of all expressions
Lowest	Returns the lowest value received
Highest	Returns the highest value received

For example, to total all of the columns of the Invoice totals in a report, create a new variable called **TOTAL_SALES**. Modify the Variable Class and change the Variable Expression to the Invoice total field $F{INVOICE_TOTAL}. Since we are summing the results, change the calculation type to **Sum**. This variable will now return the total of all Invoices that we received in the report (see **Figure 9**).



001001	Baker And Harrison	21300 North Trim Way, Suite 128	Seattle	WA	7,027.00
001002	Robinson Enterprises	5883 Guliver Lane	San Diego	CA	1,125.30
999999	Cash Sale				19.61
			Total Monthly Sales		$ 70,490.78

Figure 9. Calculation example

Summary

iReport caters to all levels of users to quickly and easily develop new reports for their data. After creating a few reports with the built-in wizards, most report authors quickly discover that it whets their appetite for creating more advanced reports that utilize custom features and a more polished presentation. By implementing the techniques covered in this article, developers will have a head start in creating better looking and more functional reports.

For more information about iReport, check out the many books and online PDFs for sale such as www.amazon.com/Definitive-Guide-iReport-Experts-Voice/dp/1590599284