

# BIRT Report Object Model – Lists & Tables

Functional Specification

Draft 1: November 29, 2004

## Abstract

*Describes the list and table elements of the Java Reporting Platform Report Object Model.*

## Document Revisions

<b>Version</b>	<b>Date</b>	<b>Description of Changes</b>
Draft 1	11/29/2004	First BIRT release.

## Contents

<b>1. Introduction .....</b>	<b>3</b>
<b>2. About Lists and Tables.....</b>	<b>3</b>
2.1 Bands.....	3
2.1.1 <i>Groups</i> .....	3
2.2 Data Binding for Lists.....	4
2.2.1 <i>Input Parameter Binding</i> .....	4
2.2.2 <i>Data Sorting</i> .....	4
2.2.3 <i>Data Filters</i> .....	4
2.2.4 <i>Aggregates</i> .....	4
2.3 Conditional Content .....	4
2.4 Multiple Presentations of Data.....	5
2.5 Charts or Tables within a List .....	5
<b>3. List Item.....</b>	<b>5</b>
3.1 Drop Sections .....	5
<b>4. Table .....</b>	<b>7</b>
4.1.1 <i>Style Precedence</i> .....	7
4.2 Computing the Number of Columns .....	7
4.3 Drop Headings.....	8
<b>5. Combining a List and a Table with a Single Data Set.....</b>	<b>9</b>
5.1 Conditional Table Header and Footer.....	10
5.2 Charts and Matrices.....	10

## 1. Introduction

This specification is part of a set that describes the BIRT Report Object Model. This document explains the two elements for displaying lists of data: the List and Table element. Both elements have a data set and display some or all of the rows from that data set. The primary difference is how the content is presented. A table provides a simple, tabular presentation of the data. A list is more general, and allows the developer to include all types of report content. A table is often used for simple listings, while a list may be used for creating form letters or creating complex report layouts.

## 2. About Lists and Tables

### 2.1 Bands

A list defines a number of *bands* that print in response to specific events. Each band can contain one or more sections called *subsections*. List bands include:

The bands of a report include:

- The *header* prints once, at the beginning of the list. It can contain a matrix, a chart, or other controls. Matrices and charts display data for the entire report. Aggregates display totals for the entire report. Controls display values for the first row in the report. Data set fields display the first row from the data set.
- The *detail* prints for every row. It can contain database fields and running totals.
- The *footer* prints once, at the end of the list. It can contain the same items as the report header. Data set fields display the last row from the query.

#### 2.1.1 Groups

Lists with more than a few rows often divide data into groups identified by a group key. Each group may have its own header and footer. Some grouped reports omit the header and simply hide the group-level fields for all except the first row for that group.

Each group has two bands:

- The *group header* prints once for each group, before the first detail row and before the first group header of any nested groups. It can contain matrices<sup>1</sup>, charts, aggregates and data set fields. Matrices and charts display all rows within the group. Aggregates compute over the entire group. Database fields show values for the first row in the group.
- The *group footer* prints once for each group, after all detail frames for that group and any group footers for nested groups. It can contain the same items as the group header, except that database fields display values for the last row in the group.

---

<sup>1</sup> Matrices are not in the first release.

## 2.2 Data Binding for Lists

Every list is bound to a data set. A list also supports data sorting, data filtering, grouping, and input parameter binding.

### 2.2.1 Input Parameter Binding

Data sets define named input parameters. A list can bind values to the data set input parameters. This feature is often required when creating a nested report. Suppose an outer list iterates over list in one database. A nested report iterates over orders for that user from a second database. In this case, the inner list can use data set parameter bindings to select orders for the customer identified in the outer list.

### 2.2.2 Data Sorting

Most lists present data in a defined order called the sort order. The sort order is defined by the group keys, followed by the sorting criteria for detail rows. The list automatically influences the data set sort order to ensure that the sort criteria are met.

Groups can also be sorted. For example, customer groups can be sorted by the total sales to the customer for the quarter.<sup>2</sup>

### 2.2.3 Data Filters

A list can apply additional filters to its data set. For example, a report may want to display the list of recent transactions broken down in several ways. First, show all sales, then returns, then exchanges. Suppose that each list requires a different format. In this case, we can use three BIRT List elements. Each list has a filter that matches only the rows to appear within that one list.

Groups can also be filtered. For example, a report can show only the top 10 customers as defined by total sales within a quarter.<sup>3</sup>

### 2.2.4 Aggregates

Header and footer bands often display totals. Totals can be calculated over a set of rows: all the rows for the list, or the rows for a group. For example, sum the sales of all customers within a given sales region. Aggregates can also include a filter over the rows. For example, sum all the sales for customers with questionable credit.

## 2.3 Conditional Content

Many reports require differing output for an event depending on certain conditions. For example, a list of customers may want to display residential, business and government customers differently.

The developer creates conditional content by creating multiple subsections within a band. Each subsection contains the content needed for a particular condition. The developer defines a condition that hides the subsection except when it is needed.

Other reports require a combination of output. For example, when printing a statement, all customers may need the company logo and statement heading. The customer is then

---

<sup>2</sup> Group sorting is not in the first release.

<sup>3</sup> Group filters are not in the first release.

displayed differently as discussed above. All customers then need the same column headings.

This example also requires multiple subsections within a band. However, the developer would apply no conditions to the first and last subsections so that they always appear, but the subsections in the middle would have conditions as above.

## 2.4 Multiple Presentations of Data

Reports sometimes want to display the same data multiple ways. For example, a portfolio report may show holdings as a chart and as a list. A sales report may show sales charted by time, by product area, and by sales region, each in a separate chart. An employee report may provide a list of employees by name and again by department.

The report developer simply adds elements to the report to show the data. BIRT detects the multiple presentations and automatically makes multiple passes over the result set.

The multiple presentations can occur for the entire result set, or can occur for only the contents of a group. In the portfolio report above, the report may list a series of funds, and each fund displays both a list and a chart. The report would have a list to iterate over all the data. A group would represent each fund. The list and chart would display the data within the fund group.

## 2.5 Charts or Tables within a List

As explained in a later section, another way to implement the above is to use a grid in a group header, and that grid contains a row with two columns. The left cell displays the chart, the right cell displays a table.

# 3. List Item

A list is a mechanism for iterating over the results of a data set and printing sections in response to the first row, last row and every row. Groups provide optional levels for headings and totals.

Some lists may contain no content when run. Perhaps the filter conditions matched no rows, or the data source was empty. Normally, none of the report sections print when the report is empty. However, developers often wish to display a “no data available” message to let the user know that the report worked, but couldn’t find any data. The developer does this by setting a property on the header and/or footer section to display it even if (or only if) the list is empty.

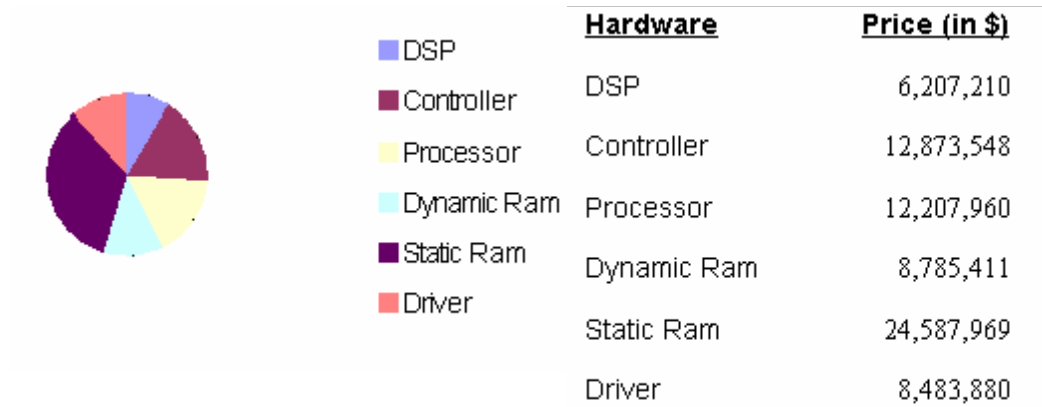
## 3.1 Drop Sections

Users sometimes want to achieve a drop effect. For example, list a chart alongside details, or list header information alongside details.

The bold parts below are drop headings:

	Order #	Value
<b>Customer: 123</b>	456	\$100
<b>Matrix Link</b>	678	\$200
<b>UK</b>	901	\$300
	902	\$400
<b>Customer: 124</b>	400	\$500
<b>Actuate Corp.</b>	500	\$300
<b>USA</b>		
<b>Customer: 125</b>	101	\$100
<b>Fred Fin</b>	102	\$200
<b>Australia</b>	103	\$300
	104	\$400

Example with a drop chart:



A drop section follows these rules:

- A group or list header contains a “drop” section that is to overlay detail sections. The drop section has the Drop property set to one of the valid choices.
- The drop section is placed into the output.
- BIRT resets the vertical output position to the top of the drop section. Detail sections then are placed next to the drop section.
- Detail sections are printed until the end of the group (or list) that defined the drop section.
- If the sum of the heights of the detail sections is less than that of the drop section, BIRT moves the output position to that following the drop section. (That is, a drop section overlaps only its own detail rows, but not detail rows of later groups or lists.)

Drop sections will not be available in the first release; they require the use of Dynamic HTML and cannot be done with standard HTML that is the target of the first release.

## 4. Table

A table is a list that is structured into a rows and columns. The columns are defined for the entire table. Rows are created in response to the same events as for a list.

Like a list, a table is defined by a series of bands. A table defines the same bands as a list. Bands within a table divide into a number of rows. Each row is further divided into a set of cells.

Tables differ from Lists in how they handle the header. A List displays the header once. A Table header can optionally appear at the top of each page.

### 4.1.1 Style Precedence

The following rules govern how BIRT ERD locates a style property for an item in a table cell:

1. Item style
2. Row style
3. Column style
4. Table Band default style (header, footer, etc.)

CSS provides for only a small subset of style properties on columns. Specifically:<sup>4</sup>

Table cells may belong to two contexts: rows and columns. However, in the source document cells are descendants of rows, never of columns. Nevertheless, some aspects of cells can be influenced by setting properties on columns.

The following properties apply to column and column-group elements:

#### 'border'

The various border properties apply to columns only if 'border-collapse' is set to 'collapse' on the table element. In that case, borders set on columns and column groups are input to the conflict resolution algorithm that selects the border styles at every cell edge.

#### 'background'

The background properties set the background for cells in the column, but only if both the cell and row have transparent backgrounds. See "Table layers and transparency."

#### 'width'

The 'width' property gives the minimum width for the column.

## 4.2 Computing the Number of Columns

BIRT ERD uses rules similar to HTML 4.0 for computing the number of columns. The following is adapted from *HTML 4.01 Specification, W3C Recommendation 24 December 1999* available at <http://www.w3.org/TR/html401/>.

---

<sup>4</sup> CSS 2.1, Section 17.3 Columns, available at <http://www.w3.org/TR/CSS21/tables.html>.

There are two ways to determine the number of columns in a table (in order of precedence):

1. If the Table element contains any column definition elements, BIRT ERD calculates the number of columns by summing the following:  
For each column definition element, take the value of its repeat attribute (default value 1).
2. Otherwise, if the Table element contains no column definition elements, BIRT ERD bases the number of columns on what is required by the rows. The number of columns is equal to the number of columns required by the row with the most columns, including cells that span multiple columns. For any row that has fewer than this number of columns, the end of that row should be padded with empty cells.

It is an error if a table contains column definition elements and the two calculations do not result in the same number of columns.

### 4.3 Drop Headings

BIRT tables support drop headings. These are very similar to the drop sections described in the List section above. The examples shown in that section can easily be implemented using a BIRT table.

A drop heading follows these rules:

- A drop heading can be defined only for a group header, but not for a group footer or column heading.
- A drop heading is defined as one or more cells that vertically span the group of table rows that represent the detail data rows for a group. That is, the drop property works much like the vertical span, except that the exact size of the span is computed at run time.
- A row with drop columns appears in a group header band.
- The vertical span starts with the group header. If, however, all other cells within the group header table row are empty, then the vertical span instead starts with the first detail row.
- A group header can contain two or more rows. In such a case, the drop can be defined in any of these rows. The drop will merge group header cells as well as detail cells as long as these header cells are 1) empty, and 2) appears after the row with the drop cell.
- At run time, the detail table rows are added to the table until the end of the group (or list) that defined the drop section.
- Once the total number of detail rows are known, this number is placed into the vertical span property of the drop cell, and this is the number of rows that the drop cell will span in the printed output for this one group.

Note that, by definition, each group will contain at least one detail row, and so a drop header will always appear. If the drop header is taller than the contents of the detail rows, then the CSS UA will insert extra space as needed to display the full header content. If, however, the sum of the detail rows is taller than the header content, then



the header content is placed within the drop cell at the top, center, or bottom of the cell depending on the header cell's formatting properties.

Some special conditions require consideration:

- The header row that contains a drop must be the last row within a group header drop for the drop to take effect. If it is not the last row in the band, then the drop is ignored.
- The above rule applies at runtime, not design time. For example, a header may have three different group headers for a group, each with a different condition. Only one is selected for any given report group. In this case, the drop column is in effect.
- The drop setting is honored only if the detail table row has an empty cell in the same position as the group header drop column. If the detail cell is not empty, then the drop setting in the group header is ignored for that column. This decision is made on a column-by-column basis. (Said another way, the Engine computes the span as the lesser of 1) the number of contiguous blank cells under the drop cell, or 2) the total number of detail rows within the group.)
- A blank detail cell is one that contains no report items, or all contained report items are data items that have a null or empty display value (after mapping and formatting.)
- If a drop is defined on the inner-most group header, that drop cell spans the (optional) group header row, and all detail rows for that group (with the above caveats.) The drop property specifies if the drop should also span into the group footer. If it does span into the group footer, the same caveats above apply to the footer cells: they must be blank.
- A drop can apply to any group level. If the group is not the inner-most, then the drop cell will span all the group headers for any nested groups, and will span all detail items for all nested groups.
- A drop cannot be applied to a group footer.

## 5. Combining a List and a Table with a Single Data Set

A report may wish to create a compound layout that includes both a list and a table. Such a structure is easy if both the list and table have their own data set. But, suppose that the report wants to print customer statements. A query returns the list of customers that need statements, along with the orders that are to appear on each statement. The report chooses to display a header portion of the statement that uses a free-form layout along with some text. The body of the statement is a table that shows each order. The structure is something like this:

XYZ Company  
Statement for the Month of September, 2004

Bill To:       ABC Corp.  
                  123 Somewhere St.  
                  Your Town, ST, 12345

<u>Order No.</u>	<u>Date</u>	<u>Amount</u>
123	8/1/2004	1234.56
145	8/9/2004	2345.67

The report design achieves this effect by using a List to iterate over the customers. The list has a group for each customer. That group has a header that displays the statement header. The header for the customer group contains a table.

```

Data Sets
  Statement Query
Body
  List (Data Set = "Statement Query")
    Groups
      Customer Group
        Header
          Order Table (Data Set = "")

```

The table leaves the data set property unset, which means to "inherit" the data set from the container. The list creates the table at the start of each inner-most group, gives the table the detailed rows, and finishes the table when a group level break occurs in the list.

Specifically, in the example above, the list handles the group levels for the customer, the table displays the details of the list of orders for that customer.

The following general rules apply for this kind of structure:

- A list can contain a table in the header or footer slot of the list. If that table has a blank data set name, the table receives rows from the list.
- A list can contain a table in the header or footer slot of a table within the list. If that table has a blank data set name, the table receives rows from the group.
- Any given header or footer can contain any number of tables.
- Tables can have their own filter and sort criteria that are applied to the rows displayed in the table.
- The appearance of a table in a header or footer does not change any other part of the list. A list can contain both a table in a header, as well as a report item in the detail band.

## 5.1 Conditional Table Header and Footer

The combination of a List and Table is also useful to create a header or footer for a table that displays data in a format different from the table layout. Define a list that specifies the data set and the header and/or footer. Put the list into the detail band of the list. The header and footer can display overall totals for the list, and will be hidden if the data set returns no rows.

## 5.2 Charts and Matrices

Charts and matrices follow the same rules. Both can also appear in a header or footer. A given header or footer can also display both a chart and a table, either one after the other as report bands, or side-by-side as items within a grid.

