

Business Intelligence and Reporting Tools

Release 1.0 Requirements
Document Version 1.0 – November 8, 2004

Contents

- Eclipse Business Intelligence and Reporting Tools Project Requirements 2**
- Project Overview 2**
- Release Objectives and Document Scope 2**
- Requirements Overview 3**
- Requirements 4**
- Report Types 4*
- Layout Elements 7*
- Simple Elements 9*
- Image Element 9*
- Viewing 9*
- Report Export (Output Formats) 9*
- Printing 10*
- Eclipse Report Designer 10*
- Scripting 12*
- Data Transformation 12*
- Data Sources 13*
- API and Extensibility 13*
- Internationalization 14*

Eclipse Business Intelligence and Reporting Tools Project Requirements

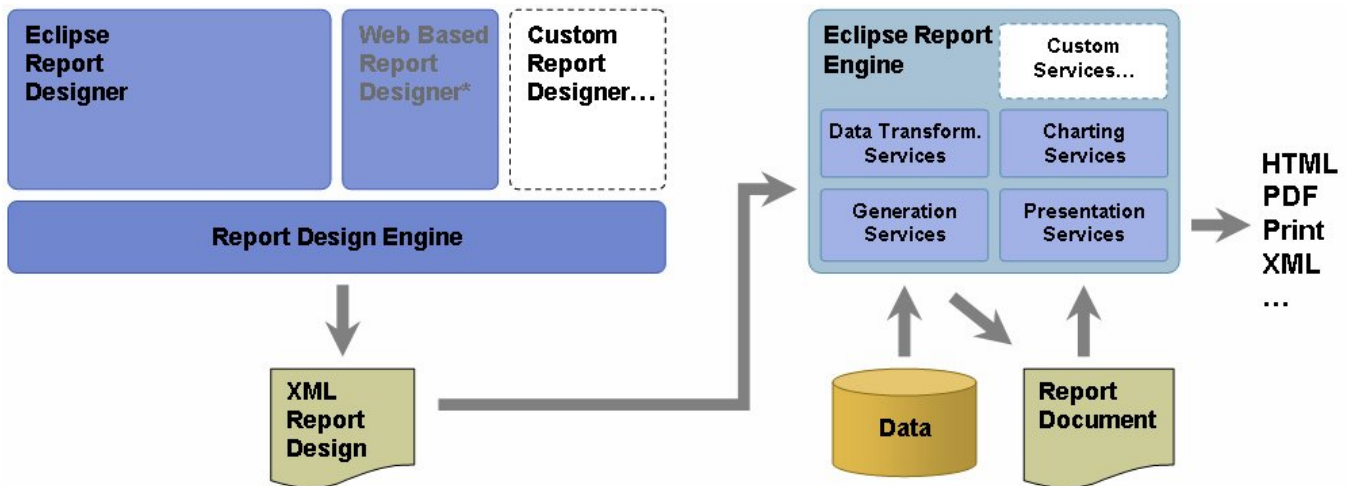
This document defines the requirements for the Eclipse Business Intelligence and Reporting Tools Project Release 1. You are invited to comment on these requirements. Please send all feedback to the Eclipse Business Intelligence and Reporting Tools newsgroup news://news.eclipse.org/eclipse.birt.

Project Overview

The Eclipse Business Intelligence and Reporting Tools (BIRT) Project (<http://www.eclipse.org/birt/index.html>) will address a broad range of needs in the business intelligence and reporting space using the Eclipse platform.

The initial goal of the Eclipse Business Intelligence and Reporting Tools Project is to provide a robust platform that can be used to quickly and effectively create and deploy reports with any degree of complexity without having the developer create the data access, processing and formatting logic using Java code or components. Reports vary dramatically in size, content and complexity. They include or combine various features (for example, listing, charts, crosstabs) and characteristics (for example, navigation, hyperlinks, interactive viewing) and can be provided in a number of formats (for example, web pages, PDF files, printed documents, Excel files).

The project will address both the design time needs of report creation as well as the run-time requirements of report generation and deployment. A high-level architecture diagram is provided below.



* Not in Release 1.

The Eclipse Business Intelligence and Reporting Project is divided into a number of sub-projects that reflect the initial focus on the reporting aspects of business intelligence and the high-level architecture of the project. The goal of BIRT Release 1 is to develop the initial release of the Eclipse Report Designer (ERD), Eclipse Report Engine (ERE), and Eclipse Charting Engine (ECE).

Release Objectives and Document Scope

The release objectives for Release 1 of the Eclipse Business Intelligence and Reporting Tools Project are:

1. Support creating of common report types, including:
 - Listings of information (Example: Transactions in an account)

- Sorting, grouping and aggregation of data with and without subtotals (Example: A listing of all product sales for each sales person, grouped by state)

Crosstab reports are not included in the first release.

2. Develop the following three subprojects (see <http://www.eclipse.org/birt/index.html> for definition of these subprojects):
 - Eclipse Report Designer, including the Report Design Engine
 - Report Engine (including Data Transformation Service, Generation Service, and Presentation Service)
 - Charting Services
3. Develop a business intelligence and reporting toolset that targets the following users:
 - Application developers: Java developer using Eclipse development tools who may or may not have any prior experience with report development.
 - Report developers: Report developers who are not Java developers but who are helping the application developers with creating reports for use in the Java based application.
4. The target users can be productive with developing the first report in 15 minutes using sample data and in 2-3 hours using the customer's data from starting downloading the product without having to read the manuals or get help or support from other users.

The purpose of this document is to define the requirements for these three subprojects to support creation of the supported report types listed above.

Requirements Overview

The table below provides an overview of the requirements for release 1. The following sections define the detailed requirements.

Each report type is created using a number of layout elements. For some report types the developer can choose between more than one report elements. For example, "Listing and Grouped Listing Reports" can be created by using either the "List" or the "Table" layout elements.

Requirement	Category
Listing and Grouped Listing Reports	Report Types
Multi-section Reports	Report Types
Subreports	Report Types
List (including groupings, subtotals)	Layout Elements
Table (including groupings, subtotals)	Layout Elements
Text	Layout Elements
Grid	Layout Elements
Chart	Layout Elements
Label Element	Simple Elements
Data Element	Simple Elements
Image Element	Graphic Items Elements
Web Style Output	Viewing
Web-based Report Viewer	Viewing
Adobe Acrobat Portable Document Format (PDF)	Report Export (Output Formats)
Printing	Printing

Requirement	Category
Report Wizard	User Interface
Schematic Design View	User Interface
Outline View	User Interface
Report Preview	User Interface
Style Builder	User Interface
Chart Wizard	User Interface
Palette View	User Interface
Page Setup User Interface	User Interface
Grouping User Interface	User Interface
Properties User Interface	User Interface
Query Editor	User Interface
Expression Builder	User Interface
Cut, Copy, and Paste	User Interface
Parameters	User Interface
Scripting	Scripting
Aggregations	Data Transformation
Computer Columns	Data Transformation
Filtering	Data Transformation
Sorting	Data Transformation
Grouping	Data Transformation
JDBC Queries	Data Access
Custom Data Access	Data Access
Multiple Data Sources and Queries	Data Access
Report Execution and Viewing API	API
Extensible Model	API
Report Design API and Format	API
Internationalization	Internationalization

Requirements

Report Types

This section defines the report types BIRT Release 1 will support.

Common for all Report Types

These items are commonly used in report designs of all types:

- Company name and logo
- Report heading
- Page header/footer

- Report date and time
- Confidentiality statement

Report designs can use any or none of these items.

Listing and Grouped Listing Reports

Listing reports and grouped listing reports are the most common reports, used for reports like invoices, phone bills, packing lists, transaction details, bank statements, etc. The data can be listed in one list or can be organized and divided into groups of related information (that is, grouped listing report). Listing reports may be laid out as a table with well-defined columns or may be more free-form where every element may not be aligned in columns. BIRT will support both simple and grouped listing reports (see below for schematic examples of both types).

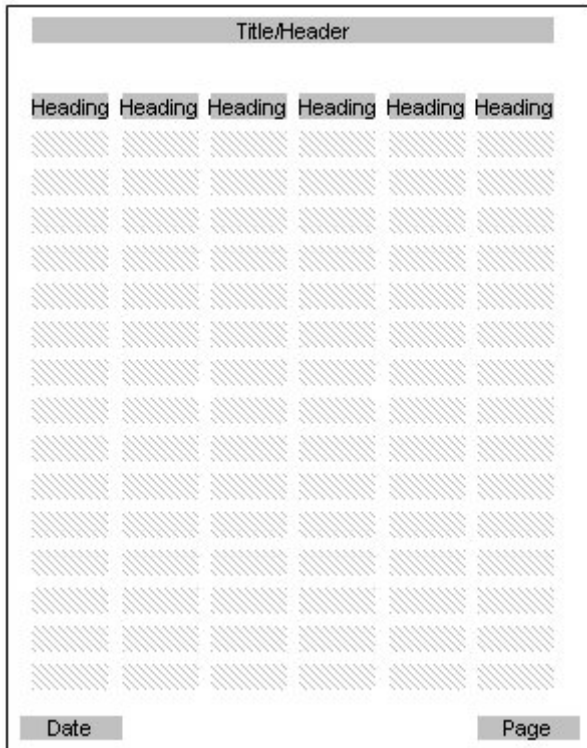


Figure 1: Simple Listing Report

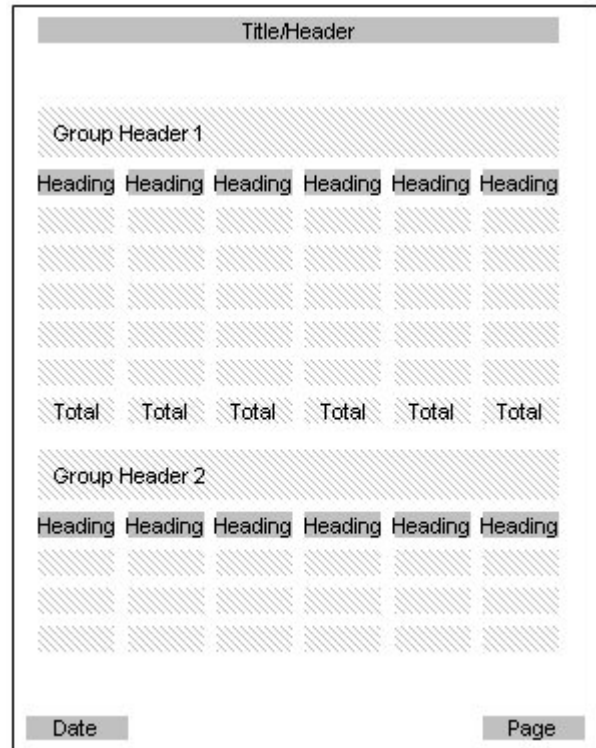


Figure 2: Grouped Listing Report

Multi-section Reports

A multi-section report contains two or more report sections positioned vertically (sequential), horizontally (side by side), or both. See below for two schematic examples. Each report section may include its own data set, or may use the same data set as one of the other report sections. Each report section can contain any type of supported report type.

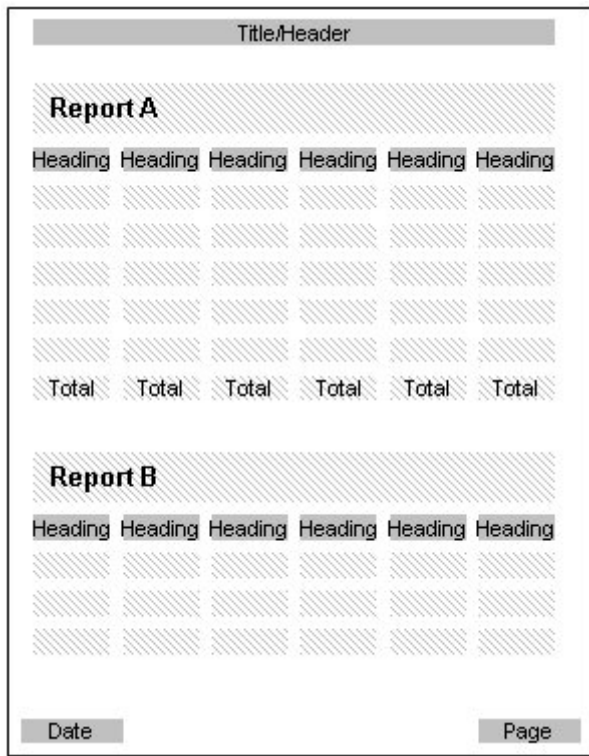


Figure 3: Sequential Report Sections

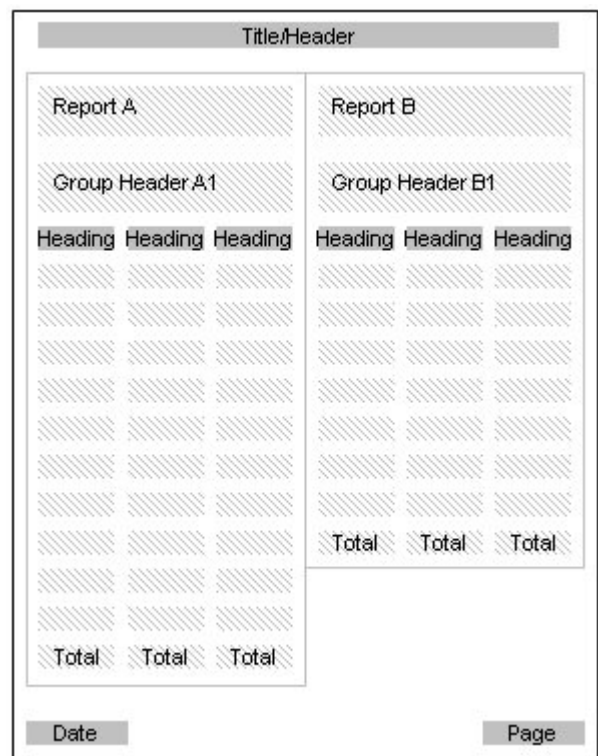


Figure 4: Side by Side Report Sections

The sequential and side by side report sections examples above are specific cases of "dashboard style" report layout. Dashboard style report layout refers to the concept of having multiple report sections positioned on the screen/page in a dashboard manner, as shown in Figure 5 below.

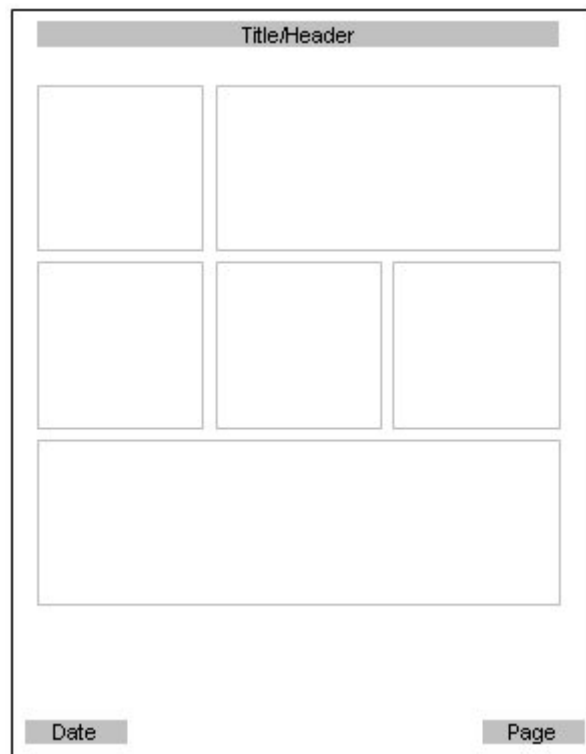


Figure 5: Dashboard Style Report Layout

Developers can use tables or grids to align the report sections vertically and horizontally.

Subreports

BIRT will support subreports as nested reports. A subreport is much like a grouped listing, except that the inner groups of the report come from a separate data set. Master/detail parameter bindings can tie together subreports. For example, the outer report may iterate over customers from one query while the subreport may list pending orders for this customer using a separate query.

The “main report” and subreports are all defined within the same report design file. Release 1 will not support referencing subreports defined in external report design files.

Layout Elements

List (including groupings, subtotals)

A list element displays a list of data from a single query.

Simple Listing

The most familiar type of report is the simple listing: a set of rows of data printed down the page. Most listings include a report header with introductory information, and report footer with totals. A list starts with a query and has three bands (printed blocks) report header, detail, and footer. The report header and footer will appear once in the report, while the body will be repeated once for each data row.

Grouped Listing

Listings with more than a few rows often divide data into groups identified by a group key (column or expression). 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 (similar to a spreadsheet with suppress repeated values).

The group header will be included once for each group, before the first detail row and before the first group header of any nested groups.

The group footer will be included once for each group, after all detail content for that group and any group footers for nested groups.

Blank Listing

Some listings may contain no content when the report is generated. Perhaps the filter conditions matched no rows, or the data set was empty. Normally, none of the report sections print when the report is empty. However, developers often wish to display a message (for example, “no data available”) to let the user know that the report worked, but couldn’t find any data. BIRT will allow the developer to specify such a message to be displayed if the listing is empty.

Table (including groupings, subtotals)

A table layout element is very similar to the list element, with one difference: a table element is structured into rows and columns (thus the name “table”). The columns are defined for the entire table; while in a list the “virtual columns” can vary between the various group and detail sections. Using a table element makes it easy to align the columns consistently across the entire report section.

A table can contain groups and subtotals/totals and will support the “blank listing” feature in the same way as a list element (see above).

Text

Many reports contain formatted text that is of variable length, such as memo fields (for example, description of a support case). To support this BIRT will provide a text element that presents a block of text. In Release 1 the text element will support variable length content and HTML formatted content.

The content of a text element can be of variable length and will automatically size according to the length of the content. When used within a grid element (see below), any elements in cells below the cell containing the variable sized text element will be moved down accordingly.

The content of a text element can be one data field or a concatenation of multiple data fields.

Grid

BIRT will provide a grid layout element to help align reports items for clean layout and to make it easy to design reports with complex layouts.

Grid is familiar to anyone who has used HTML tables, Word tables or Excel: data is divided into a series of rows and columns. Items can span both rows and columns. The grid layout is ideal for many simple reports and reports with a “dashboard style” layout (that is, multiple sections organized vertically, horizontally, or both).

The grid element will not process data rows unlike the list and table elements. The grid can be used to create the main structure of the report and to align report items within another element, but not iterate through all the data rows.

Many reports include content that can grow or shrink based on a number of factors. The grid element will automatically handle this issue by resizing rows and columns as needed.

A cell in a grid will be able to contain one or more items.

Chart

Charts are key element of many reports. Charts elements can be placed in various sections of a report, either individually or several charts together (for example four pie charts next to each other showing the sales breakdown between the various sales channels, one chart for each of the four global regions).

BIRT will include charting services that will support the following capabilities:

- Bar, line, pie, XY-scatter, stock charts (horizontal and vertical)
- 2D, 2.5D, and 3D charts
- Overlay Y-axes
- Solid and transparent colors
- Automatic and manual axis scaling
- Hyperlink from the various elements of a chart to a URI

Simple Elements

This section identifies the simple report elements required for BIRT release 1.

Label Element

A label element is a static piece of text displayed in the report.

Data Element

A data element displays data from a query, a formula, a parameter, etc. BIRT will offer one common data element that can contain any kind of data (string, integer, date, etc.). The data element will support all standard formatting capabilities.

The data element will support presenting data in a fixed locale (for example, always US locale) or in a specified locale the report is presented in (for example, in German locale for user A, UK locale for user B).

Image Element

Reports often display images in various formats: corporate logos, product images, icons, and so on. Popular image formats are JPEG for photographs, and GIF, PNG and SVG for computer-generated graphics such as logos.

Requirements for the image element:

- Reference the image via a URI
- Associate a hyperlink with the image

The image element will provide options for handling images that are larger or smaller than the size of the element itself.

Viewing

BIRT will support “web style output” where all the content is presented in one continuous web page. The web style output is ideal for online viewing of shorter reports.

Web Style Output

BIRT will be able to generate report output that is presented as one continuous web page and not broken into pages. This is useful for applications and reports that are primarily used for viewing in a browser, such as account transaction details.

Web-based Report Viewer

BIRT will offer a web-based report viewer for displaying user interface for collecting report parameters and to view reports.

Report Export (Output Formats)

Report export works with the formatted report output, not the underlying data. Export reproduces the report pages as best as possible in a variety of output formats. Report export is the right choice when the user wants to read the report, but is not usually a good format if the user wants access to the data.

Adobe Acrobat Portable Document Format (PDF) Output

BIRT will be able to export the report content to Adobe PDF format.

Formatting Objects (FO) Output

BIRT will support exporting report content to [Formatting Objects](http://xml.apache.org/fop/), from which tools such as FOP (<http://xml.apache.org/fop/>) can be used to convert the report to other formats.

Printing

Printing capabilities can be incorporated into the application by the developer using the output from the presentation services.

Eclipse Report Designer

BIRT will offer a desktop authoring environment for report development based on Eclipse. The Eclipse Report Designer enables application and report developers to create simple and complex reports for use within their organization. The tool recognizes and caters to the broad range of report development skills from the non-programmer report developer focused on report layout to the application developer looking for sophisticated control over report creation.

The Eclipse Report Designer will provide a report wizard that helps users create new reports and a schematic design view (design editor) that enables the users to both modify and fine-tune the design created by the wizard and to design reports from scratch without using the report wizard. It is expected that most new and novice users will start with the report wizard and then use the design view to finish up the report design. Experienced users may use the wizard to create the starting point for new reports or may go directly into the design view.

Report Wizard

The Eclipse Report Designer will provide an easy to use report wizard which will guide users through the process of creating a report. The wizard will support creating a number of common reports, initially listing and grouped listing reports. The wizard will allow the user to:

- Choose what type of report to create
- Specify the data to be included in the report (tables, columns, sorting, filtering)
- Specify how to group the data
- Choose from a set of layout (formatting) styles

The key objective of the wizard is to make it easy for novice users to create reports. The wizard will not expose most advanced features as it would make the wizard too complex and hard to use.

Schematic Design View

The schematic design view displays the report design without including the data, that is, the detail section is only displayed once, not one time per data row. The design view:

- Displays the visual report content such as lists, tables, data items, labels, images and more
- Shows the affect of visual properties such as fonts, colors, sizes and more
- Shows the visual structure of the report
- Allows the user to directly edit the report design

The schematic design view gives the report developer full control of the content, layout, formatting, and processing of the report. It provides the report developer with precise control of the report design.

Outline View

The outline view shows an abbreviated, abstract description of the report: it shows the abstract display of the report structure including queries, groupings, frames and so on. It helps the user visualize how the report is structured.

The outline view is an extension of a standard Eclipse feature that displays a tree view of the major portions of the report. The user clicks on an item in the outline view to make it visible in the editor.

Report Preview

The report designer will provide a HTML preview embedded within the designer. The preview will generate the report and display it in the same way as if it was deployed to the application.

Style Builder

BIRT will support style sheets to provide easy to use and productive mechanism to establish uniform style. Styles can be used for:

- Defining the format of multiple items will have the same non-default format.
- Changing the default format applied to all items.

The Style Builder provides the user interface to define and manage styles within the Eclipse Report Designer.

Chart Wizard

A user friendly SWT based chart definition wizard will be integrated into the BIRT design tool to facilitate chart definition.

Palette View

The palette is a simple, uniform way to access all to all elements that can be added to the report. There are many kinds of items that the user can add to the report: parameters, database fields, special values, library components, and so on. The palette must provide a tree view or other organizational metaphor to let the user browse the various categories of items.

When the user drags a palette entry into the report, the Eclipse Report Designer creates the proper report item in the report design.

Page Setup User Interface

The Eclipse Report Designer will provide user interface for the report developer to define the page setup. This includes page size, orientation, and margins. The page definition will be used for output to PDF and FO.

Grouping User Interface

The Eclipse Report Designer will provide user interface for the report developer to define the grouping levels for the report item (table and list). The user interface must support defining:

- Group key and interval: how to determine the group into which a particular data row will be placed.
- Group sort order: the order in which groups appear within the report.

The Eclipse Report Designer will provide visual indication of the grouping levels in the layout view.

Properties User Interface

The Eclipse Report Designer will include specific user interface for setting common properties such as fonts, paragraph formatting, etc. In addition, a generic property sheet is required for setting properties that are not included in specific property sheets and for users who prefer to see all properties at a glance. In the generic property sheet the properties will be organized in ways that make it easy for the report developer to navigate, find, and use the desired properties.

This user interface provides “builder buttons” where appropriate to help the user set the particular property (examples include color chooser, expression builders, etc.).

The property sheet will support selecting multiple report components and changing their common properties at the same time.

Query Editor

The Eclipse Report Designer will include a query editor where the report developer can type or paste in the SELECT statement. The query editor can handle arbitrarily complex SELECT statements.

Expression Builder

The Eclipse Report Designer will include an expression builder to help the report developer creating expressions. Expressions can be used for manipulating the data displayed in a data control. The expression builder will help the developer creating expressions with the correct syntax.

Cut, Copy, and Paste

The Eclipse Report Designer will support cut, copy, and paste of report elements, text, etc. throughout the product.

Cut, copy, and paste for report elements can be used both inside one report design and between individual report designs.

Parameters

BIRT will support parameters that can be passed into the report at report execution time. Parameters are most commonly used as query parameters to filter the data returned from the data set (for example, ClientName parameter used to filter the query such that only transactions for the specified client is returned). Parameters can also be used to pass other types of information into the report, such as user id and password or a threshold value used to highlight orders with order total above the specified value.

Parameters can be organized in groups to keep related parameters together in the parameter screen.

When the report developer is previewing a report in the Eclipse Report Designer, a dialog will be displayed asking the user to enter the desired parameter values.

Scripting

BIRT will support scripting to customize the reports. Scripting can be used to access custom data sources, perform custom data processing, programmatically control the report content and layout, etc. Various report components will have a set of methods where custom script can be added. The scripts can access the report object model to inspect and set various attributes of the report.

BIRT will use JavaScript as the scripting language.

Data Transformation

The report generation services which is part of the Report Engine will provide the data transformation capabilities described in the following sections. These capabilities will also be available for generating and previewing reports within the Eclipse Report Designer.

Aggregations

Aggregations and calculations are very common in reports. For example: in a sales forecast report display the sales forecast total per sales rep, region, and company wide. BIRT will support single-pass aggregations in Release 1.

Computed Columns

BIRT will support computed columns (for example, "Extension = Price * Quantity").

Filtering

Filtering is used to limit the report content to the desired subset of the data. BIRT will support filtering.

Sorting

The Report Engine will have the capability to sort the data from the underlying data source when required to generate the report.

Grouping

The Report Engine will have the capability to group the data from the underlying data source when required to generate the report.

Data Sources

JDBC Data Sources

BIRT will support JDBC data sources in Release 1. The report developer can build queries against JDBC data sources using the query editor within the Eclipse Report Designer (see above). Release 1 will not support stored procedures.

Custom Data Access

BIRT Release 1 will support custom data access through scripting. This can be used to access for example Java objects as the data source for a report.

Multiple Data Sources and Queries

The Eclipse Report Designer will support multiple data sources and queries per report, with each one providing data to separate report sections (for example, two tables listed sequentially in the report).

API and Extensibility

BIRT will provide the necessary APIs to enable Java application developers to embed and integrate the reporting capabilities into their application.

Report Execution and Viewing API

The Report Engine will provide a report execution and viewing API. The API can take parameter values that will be passed into the report at execution time. The API will support the various report output formats.

Extensible Model

BIRT and the Eclipse Report Designer will provide a number of commonly requested extension points to support application-specific and general purpose extensions to the designer. Specific examples:

- Extensible data access capabilities to support reporting against any target data source.
- Custom scripting to incorporate complex and/or application specific business logic into report designs (for example, custom functions).
- Extensible model that will support both extended components and extended properties. This can be used to create new visual components to be incorporated into reports (for example, new graphical representations of data) and new attributes for existing components.
- Custom charts may be plugged into the charting engine. This includes extension points at the chart model, the property sheets defining the user interface and custom renders for the new chart type being defined.

Report Design API and Format

The report design format for BIRT will be XML-based. BIRT will offer a report design API through the Report Design Engine providing a programmatic interface to create and modify report designs.

The open XML based report design format allows custom report design products (for example, web based report designer) to be built for various types of devices for different user applications.

The XML-based report design format will support version comparison (that is, diff) such that a developer can easily determine what has changed between two versions and support programmatic generation of report designs from non-Java applications. The report design format will be human readable to support version and change analysis and programmatic generation of designs.

Internationalization

BIRT will support any locale supported by Java and will be fully internationalized, with the exception of Right-to-Left languages in Release 1. BIRT will support Unicode.

The initial release of the Eclipse Report Designer will only be available in English, but since it will be fully internationalized it can easily be translated into any language.

Reports may be generated and viewed by users in many different locales. To support designing one report that can serve users in many locales, BIRT and the Eclipse Report Designer will support both locale-sensitive and fixed-locale format strings. The locale-sensitive format strings will format the content according to the user's locale. Fixed-locale format strings will always format the content in the locale specified by the report developer.

BIRT will use binary sorting for sorting data. Lexical sorting will not be included.

BIRT will support externalizing report content (such as report titles, labels, etc.) which will make it easy for developers to create multi-lingual reports, reports that display the static text in the user's language.